PROLAND: A NEW INFEASIBLE MANIFOLD OPTIMIZATION METHOD FOR LORA FINE-TUNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Low-rank adaptation (LoRA) enables parameter-efficient fine-tuning but can suffer from redundant basis directions. Constraining the basis factor to the Stiefel manifold addresses this redundancy through orthonormality. Landing methods allow temporarily infeasible iterates to explore a neighborhood of the manifold while progressively restoring orthogonality without per-step retractions. To improve optimization efficiency while retaining this flexibility, we introduce ProLand (Product-space Landing), a retraction-free optimizer that combines coordinated factor updates with Muon-style spectral directions. By designing the updates in the space of the represented weight increment, ProLand separates objective optimization from orthogonality restoration, supporting efficient feasibility recovery. This coordinated construction also enables Muon-style directions to guide objective optimization while preserving that separation. Experiments cover four GLUE validation tasks on DeBERTa-v3-base and seven language benchmarks on Qwen2.5-3B, Llama-3.2-3B, and Moonlight-16B-A3B across multiple adapter ranks. Across these 11 tasks, ProLand achieves the highest average score among the compared methods in every tested encoder and decoder model–rank configuration. Code is available at `https://anonymous.4open.science/r/proland-supplement-7A24/`.

1 INTRODUCTION

Low-rank adaptation (LoRA) makes fine-tuning large language models more accessible by learning a low-rank weight increment $\Delta W = BA$ while keeping pretrained weights fixed (Hu et al., 2022). Its parameter efficiency depends on making effective use of a small number of trainable directions. The columns of B define these directions in the output space, while A contains their coordinates (Zhu et al., 2024). However, Park et al. (2025) observe that AdamW fine-tuning can produce highly correlated columns of B, leading to basis redundancy and inefficient use of the available rank. They address this issue by imposing $B^\top B = I$, so that the basis is optimized on the Stiefel manifold. This constraint keeps the basis directions orthonormal while leaving their coordinates free to adapt. It preserves the class of representable rank-at-most-r weight increments, as discussed in Section 2.

Once the Stiefel constraint is imposed, a central algorithmic choice is how to maintain orthogonality during optimization. Stiefel-LoRA (Park et al., 2025) enforces orthonormality through a retraction at each step. Landing methods (Ablin & Peyré, 2022; Ablin et al., 2024) instead allow temporarily infeasible iterates and use an explicit correction to restore feasibility, avoiding per-step retractions. Manifold-LoRA (Zhang et al., 2026) applies this approach to LoRA fine-tuning. Beyond avoiding repeated retractions, allowing controlled departures from the manifold gives the optimizer additional freedom to explore the surrounding space. Figure 1 illustrates the potential benefit by comparing our proposed method (ProLand) with variants that replace its Landing correction with per-step retractions to enforce $B^\top B = I$. From the same initialization and under the same rule for computing product-space directions at their respective iterates, ProLand reaches a better stationary solution in this example. This observation motivates exploring infeasible paths while retaining control over constraint violation.

Our goal is to accelerate this form of infeasible optimization for LoRA: make effective progress on the fine-tuning objective while rapidly recovering feasibility. These two goals are coupled, since an

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.

update that reduces the loss can also perturb the orthogonality of B and interfere with the Landing correction. At the same time, the model depends on the joint weight increment BA, so the directions chosen for the two factors should be coordinated. We introduce **ProLand** (Product-space Landing), a retraction-free optimizer that addresses these issues through two complementary design choices.

First, coordinated updates in the product space support rapid feasibility recovery. We design the joint first-order change $D_B A + B D_A$ in the represented weight increment and realize it through factor directions satisfying $B^\top D_B = 0$, even when B is infeasible. This condition makes the optimization step leave $B^\top B$ unchanged to first order and eliminates its first-order interaction with the Landing correction. Consequently, the optimization step perturbs the constraint residual only at second order in its step size. Under the conditions of Section 4, the residual contracts geometrically toward a neighborhood whose radius is quadratic in that step size. This provides a mechanism for recovering feasibility while continuing to optimize the represented weight increment.

Second, Muon-style directions guide objective optimization. We incorporate the spectral update principle of Muon (Jordan et al., 2024) into this coordinated construction. ProLand forms a momentum input in the space of realizable first-order changes in BA, computes its polar factor, and projects the result back into that space. The spectral direction is therefore defined at the level of BA, rather than independently in the two factor spaces, and is realized by the factor updates above while retaining their separation from feasibility restoration. Thus, the spectral direction and the Landing correction can be combined without a retraction at every step.

For the resulting method, we establish joint guarantees for objective stationarity and feasibility under suitable regularity and step-size conditions, assuming exact arithmetic. We show that the deterministic ProLand method attains $O(T^{-1/2})$ product-space stationarity and $O(T^{-1})$ constraint violation at the randomized output. With stochastic gradients and horizon-dependent momentum and step size, the corresponding expected bounds are $O(T^{-1/4})$ and $O(T^{-3/2})$. The feasibility bounds also hold uniformly along each trajectory after an initial transient.

To make this construction practical for LoRA fine-tuning, we derive an exact low-rank implementation using momentum factors of width at most $2r$ and polar-factor cores of size at most $2r \times 2r$, avoiding dense weight-sized optimizer states. In experiments, ProLand achieves the highest task-averaged scores among the compared methods on four GLUE tasks with DeBERTa-v3-base and across all tested decoder model–rank configurations. Matched ablations show that the Landing correction substantially reduces constraint violation.

1.1 Related Work and Positioning

Several methods improve LoRA optimization by coordinating the updates of its two factors. LoRA-Pro (Wang et al., 2025) and LORO (Mo et al., 2025) account for the change in the represented matrix when designing factor updates for fine-tuning and low-rank pretraining, respectively. LoRA-Muon (Cesista et al., 2026) and sMuon (Anson et al., 2026) extend Muon-style spectral optimization to low-rank adapters. In particular, sMuon reconstructs a low-rank gradient from factor gradients and derives coordinated directions through linearization and least squares, with momentum represented in factor coordinates. ProLand additionally imposes the Stiefel constraint on B and selects a factor realization satisfying $B^\top D_B = 0$ at both feasible and infeasible iterates. This realization enables its spectral update to work with a separate feasibility correction.

Spectral updates have also been developed explicitly for manifold optimization. Riemannion (Bogachev et al., 2026) extends Muon to optimization of the represented matrix on the fixed-rank manifold. LoRA-TSD (Andriianov et al., 2026) develops tangent-space spectral descent on that manifold and maps its updates back to the factors through a factor-induced retraction. Skewon (Solonko et al., 2026) constructs spectral directions for a Stiefel-constrained matrix by solving a linear minimization problem over tangent directions subject to a spectral-norm constraint, followed by a retraction. These methods illustrate how spectral updates can be adapted to manifold geometry. ProLand focuses on the complementary challenge of combining such directions with infeasible iterates and controlled restoration of the basis constraint.

A complementary line of work avoids per-step retractions through Landing methods, which allow infeasible iterates and restore orthogonality through an explicit correction (Ablin & Peyré, 2022; Ablin et al., 2024). Manifold-LoRA (Zhang et al., 2026) applies this approach to Stiefel and oblique

constraints. Its update construction uses the gradient with respect to A and a projected-gradient expression for B, together with a feasibility correction; its analysis establishes convergence of the underlying retraction-free gradient method. ProLand retains the Landing mechanism and constructs its optimization direction in the space of joint changes in BA. The resulting factor directions satisfy $B^\top D_B = 0$ even away from the manifold, eliminating all terms linear in the optimization step size from the constraint-residual recurrence. For this construction, we establish product-space stationarity guarantees and pathwise bounds on constraint violation under the conditions in Section 4. Appendix F gives further comparisons.

2 Problem Setup and Preliminaries

Consider a single LoRA module with all other model parameters held fixed. We parameterize its trainable weight increment as $\Delta W = BA$, where $B \in \mathbb{R}^{m\times r}$, $A \in \mathbb{R}^{r\times n}$, and $1 \le r \le \min\{m, n\}$. Let $f(\Delta W)$ denote the corresponding fine-tuning objective. Imposing the orthonormality constraint $B^\top B = I_r$ on B while leaving A unconstrained (Park et al., 2025; Zhang et al., 2026) yields

$$\min_{B\in \mathrm{St}(m,r),\, A\in\mathbb{R}^{r\times n}} f(BA), \tag{1}$$

where $\mathrm{St}(m, r) := \{B \in \mathbb{R}^{m\times r} : B^\top B = I_r\}$ denotes the Stiefel manifold.

The orthonormality constraint preserves the set of representable weight increments:

$$\left\{BA : B \in \mathrm{St}(m, r),\ A \in \mathbb{R}^{r\times n}\right\} = \left\{\Delta W \in \mathbb{R}^{m\times n} : \mathrm{rank}(\Delta W) \le r\right\}.$$

Thus, the constraint restricts the factorization, not the class of rank-at-most-r weight increments. A proof is given in Appendix A.1.

At a feasible point $B \in \mathrm{St}(m, r)$, the tangent space $T_B\,\mathrm{St}(m, r)$ consists of matrices $D \in \mathbb{R}^{m\times r}$ satisfying $B^\top D + D^\top B = 0$ (Absil et al., 2008). Accordingly, for arbitrary B, let $C(B) := B^\top B - I_r$, and we measure feasibility by $\|C(B)\|_{\mathrm{op}}$, which is zero at feasible points.

First-order product geometry. The differential of the product map $\Phi(B, A) := BA$ is

$$\mathrm{D}\Phi(B, A)[\Delta B, \Delta A] = \Delta B\, A + B\, \Delta A.$$

The image of this differential, $\mathcal{S}_{B,A} := \mathrm{range}\,\mathrm{D}\Phi(B, A)$, is the corresponding first-order product space. Let $(\cdot)^\dagger$ denote the Moore–Penrose pseudoinverse, and define $P_B := BB^\dagger$ and $P_A := A^\dagger A$, the orthogonal projectors onto the column space of B and the row space of A, respectively. Under the Frobenius inner product $\langle X, Y\rangle_{\mathrm{F}} := \mathrm{tr}(X^\top Y)$, with induced norm $\|\cdot\|_{\mathrm{F}}$, the orthogonal projector onto $\mathcal{S}_{B,A}$ is

$$\Pi_{B,A} Z = P_B Z + (I - P_B) Z P_A. \tag{2}$$

The same orthogonal projection operator applies to arbitrary-rank factors, including $B \notin \mathrm{St}(m, r)$; see Appendix A.1.

For $F := \nabla f(BA)$, define the projected gradient $R := \Pi_{B,A} F$. Since $\Pi_{B,A}$ is an orthogonal projector, $R = 0$ if and only if $F \perp \mathcal{S}_{B,A}$. This is equivalent to $B^\top F = 0$ and $FA^\top = 0$; see the derivation in Appendix A.2. When $B \in \mathrm{St}(m, r)$, these conditions characterize first-order stationarity of the factorized constrained problem. We use the nuclear norm $\|R\|_*$ as the product-space stationarity measure and quantify constraint violation separately by $\|C(B)\|_{\mathrm{op}}$.

Polar factor and spectral–nuclear duality. For a nonzero matrix H, let $H = U\Sigma V^\top$ be its compact singular value decomposition, retaining only nonzero singular values. We define its canonical polar factor as $\mathrm{polar}(H) := UV^\top$, with $\mathrm{polar}(0) := 0$ (Jordan et al., 2024). This choice satisfies $\|\mathrm{polar}(H)\|_{\mathrm{op}} \le 1$.

The nuclear norm is dual to the operator norm, and the polar factor attains the corresponding dual maximum:

$$\max_{\|D\|_{\mathrm{op}}\le 1} \langle H, D\rangle_{\mathrm{F}} = \langle H, \mathrm{polar}(H)\rangle_{\mathrm{F}} = \|H\|_*.$$

ProLand constructs its product-space update direction from this polar factor.

3 THE PROLAND METHOD

ProLand combines a product-space spectral update with a Landing correction. Its construction addresses three geometric issues and one implementation issue. First, as the factors change, past momentum need not lie in the current product space. Second, even an input in this space may have a polar factor outside it. Third, the factor directions must realize the selected first-order product direction while allowing a separate correction to control constraint violation. Finally, the product-space operations should admit an exact low-rank implementation.

At iteration t, let G_t be a stochastic estimate of $\nabla f(B_t A_t)$. Algorithm 1 gives the complete iteration. Both updates are computed at (B_t, A_t) and applied simultaneously.

Algorithm 1 ProLand

Require: Initial factors B_0, A_0; initial momentum $\bar{H}_{-1}$; iteration budget T; momentum coefficient $\beta \in [0, 1)$; optimization step sizes $\{\gamma_t\}_{t=0}^{T-1}$; Landing step size $\rho > 0$

for $t = 0, \dots, T-1$ **do**
  Obtain G_t at $B_t A_t$; set $\Pi_t \leftarrow \Pi_{B_t, A_t}$.
  $\bar{H}_t \leftarrow \beta \Pi_t \bar{H}_{t-1} + (1-\beta)\Pi_t G_t$.
  $J_t \leftarrow \beta \bar{H}_t + (1-\beta)\Pi_t G_t$; $Y_t \leftarrow \mathrm{polar}(J_t)$.
  $D_{A,t} \leftarrow B_t^\dagger Y_t$.
  $D_{B,t} \leftarrow (I - P_{B_t}) Y_t A_t^\dagger$.
  $C_t \leftarrow B_t^\top B_t - I_r$.
  $A_{t+1} \leftarrow A_t - \gamma_t D_{A,t}$;
  $B_{t+1} \leftarrow B_t - \gamma_t D_{B,t} - \rho B_t C_t$ ▷ simultaneous update
end for

Reprojected momentum. Write $\mathcal{S}_t := \mathcal{S}_{B_t, A_t}$ and $R_t := \Pi_t \nabla f(B_t A_t)$. ProLand forms $\bar{H}_t$ from the reprojected momentum $\Pi_t \bar{H}_{t-1}$ and the projected gradient $\Pi_t G_t$, a stochastic estimate of R_t. Both quantities lie in $\mathcal{S}_t$, so $\bar{H}_t \in \mathcal{S}_t$. Similarly, $J_t \in \mathcal{S}_t$ because it combines $\bar{H}_t$ with the same projected gradient.

Product-space polar direction. Although $J_t \in \mathcal{S}_t$, its polar factor Y_t need not lie in this space. ProLand therefore selects the first-order product direction $\xi_t := \Pi_t Y_t$.

Factor realization and Landing. The factor directions in Algorithm 1 realize this projection without explicitly forming ξ_t:

$$B_t D_{A,t} + D_{B,t} A_t = \xi_t, \qquad B_t^\top D_{B,t} = 0. \tag{3}$$

The orthogonality condition holds even when B_t is infeasible and implies $D_{B,t} \in T_{B_t} \mathrm{St}(m, r)$ at feasible points. This condition is not restrictive: every direction in $\mathcal{S}_t$ admits a factor realization satisfying it; see Appendix A.3.

To control constraint violation without per-step retractions, ProLand adds the Landing correction $-\rho B_t C_t$ to the B-update. This is a negative gradient step for $\mathcal{N}(B) := \frac{1}{4}\|C(B)\|_{\mathrm{F}}^2$, since $\nabla \mathcal{N}(B_t) = B_t C_t$. This correction is also compatible with the optimization direction: the condition $B_t^\top D_{B,t} = 0$ gives $\langle D_{B,t}, B_t C_t \rangle_{\mathrm{F}} = 0$, so the optimization direction and the Landing correction are orthogonal, with their step sizes γ_t and ρ controlled separately.

Figure 1 compares Landing and per-step retractions from the same initialization, using the same product-space direction rule at their respective iterates. They approach distinct stationary solutions while all attain feasibility to numerical tolerance. This example shows that replacing Landing with per-step retractions can change the solution reached. Further comparison details are given in Appendix H.

Exact low-rank implementation. ProLand admits an exact low-rank implementation. Since every matrix in $\mathcal{S}_t$ has rank at most $2r$, both $\bar{H}_t$ and J_t admit thin factorizations. Given an initial momentum factorization of width at most $2r$, the recursion operates on these factors and evaluates

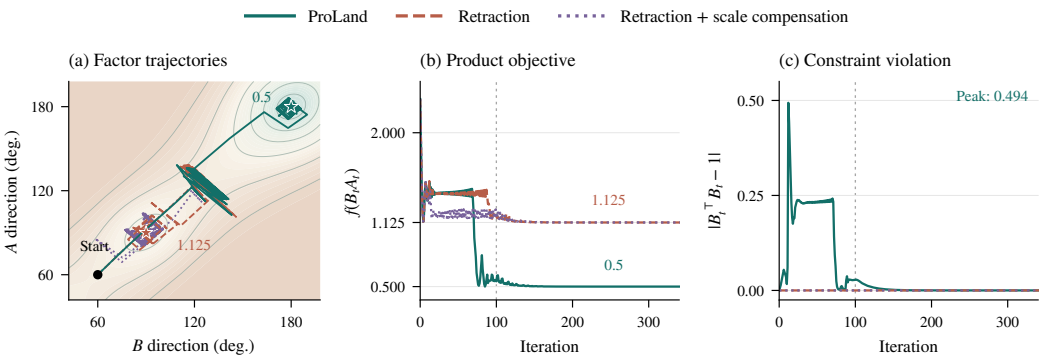

Figure 1: **Feasibility treatment changes the solution.** (a) Factor trajectories over loss contours minimized over product scale; (b) product objective values; (c) Constraint violation.

the polar factor through a core of dimensions at most $2r \times 2r$. The factor gradients $g_{A,t} = B_t^\top G_t$ and $g_{B,t} = G_t A_t^\top$ from the same sampled loss suffice for the recursion, without forming G_t. Excluding gradient evaluation, each iteration requires $O((m+n)r^2 + r^3)$ arithmetic operations and $O((m+n)r)$ additional storage. Appendix B gives the implementation and proves its equivalence to Algorithm 1 in exact arithmetic.

4 Theoretical Analysis

Section 3 shows that ProLand realizes the target first-order product direction through coordinated factor updates. We analyze Algorithm 1 in exact arithmetic, first establishing feasibility and factor conditioning and then deriving product-space stationarity guarantees.

Assumption 4.1 (Objective and stochastic regularity)**.** The objective f is bounded below and L-smooth: $f(W) \geq f_* > -\infty$ and $\|\nabla f(W) - \nabla f(W')\|_{\mathrm{F}} \leq L\|W - W'\|_{\mathrm{F}}$ for all W, W'. In the stochastic setting, let $\{\mathcal{F}_t\}$ be the filtration generated by the algorithmic history up to iteration t, before sampling G_t. For the stochastic gradient G_t defined in Section 3, assume $\mathbb{E}[G_t \mid \mathcal{F}_t] = \nabla f(B_t A_t)$ and $\mathbb{E}[\|G_t - \nabla f(B_t A_t)\|_{\mathrm{F}}^2 \mid \mathcal{F}_t] \leq \sigma^2$ for a finite constant $\sigma \geq 0$. These are standard assumptions in stochastic nonconvex optimization (Ghadimi & Lan, 2013; Bottou et al., 2018).

Since LoRA commonly uses a zero-initialized factor, we impose the following regularity condition only after a finite initialization phase and relabel the first iterate of this regular segment as $t = 0$ (Hayou et al., 2024a; Wang et al., 2025).

Assumption 4.2 (A-factor regularity)**.** There exist constants $a_- > 0$ and $A_{\max} < \infty$, independent of T, such that $a_- \leq \sigma_r(A_t) \leq \|A_t\|_{\mathrm{op}} \leq A_{\max}$ for $0 \leq t \leq T$. In the stochastic setting, these bounds hold almost surely with common deterministic constants.

Similar full-rank and uniform conditioning assumptions are used in analyses of low-rank factorization (Wang et al., 2025; Savostianova et al., 2023). Here, the uniform lower bound keeps $A_t^\dagger$ bounded, enabling uniform estimates for the factor realization and the variation of the product-space projector Π_t; see Appendix D.

We first establish pathwise feasibility and the resulting conditioning of B_t.

Proposition 4.1 (Feasibility and B-factor conditioning)**.** *Let Algorithm 1 satisfy Assumption 4.2. Fix $0 < \epsilon < 1$ and $0 < \rho \leq 1/2$, suppose $\|C_0\|_{\mathrm{op}} \leq \epsilon$, and define $q_C := 1 - (1-\epsilon)(2\rho + \rho^2\epsilon) \in (0,1)$. If the positive step sizes satisfy $\sup_{0 \leq t < T} \gamma_t \leq a_- \sqrt{\epsilon(1-q_C)}$, then, for all $0 \leq t \leq T$,*

$$\|C_t\|_{\mathrm{op}} \leq \epsilon, \qquad \sqrt{1-\epsilon} \leq \sigma_r(B_t) \leq \sigma_1(B_t) \leq \sqrt{1+\epsilon}, \qquad \|B_t^\dagger\|_{\mathrm{op}} \leq (1-\epsilon)^{-1/2}.$$

Moreover, for constant $\gamma_t \equiv \gamma$,

$$\|C_t\|_{\mathrm{op}} \leq q_C^t \|C_0\|_{\mathrm{op}} + \frac{\gamma^2(1-q_C^t)}{a_-^2(1-q_C)}. \tag{4}$$

The mechanism behind Proposition 4.1 follows from the exact Gram recursion $C_{t+1} = (1-2\rho)C_t + (\rho^2 - 2\rho)C_t^2 + \rho^2 C_t^3 + \gamma_t^2 D_{B,t}^\top D_{B,t}$. The identity $B_t^\top D_{B,t} = 0$ cancels the terms linear in γ_t, including those coupling the optimization step and the Landing correction. Within the safe region, the Landing correction contracts the feasibility error, while the optimization step contributes only a quadratic perturbation, yielding the geometric transient and $O(\gamma^2)$ term in the bound above. Detailed results and proofs are given in Appendix C.

Together with Assumption 4.2, these bounds provide the uniform factor conditioning needed to control the factor realization and the variation of Π_t.

We next turn to product-space stationarity. Spectral–nuclear duality and the self-adjointness of Π_t give $\langle \nabla f(B_t A_t), \xi_t \rangle_{\mathrm{F}} = \langle R_t, Y_t \rangle_{\mathrm{F}} \geq \|R_t\|_* - 2\|J_t - R_t\|_*$. This inequality supplies the leading descent term, with an error determined by the polar-factor input mismatch $J_t - R_t$. Combining it with smoothness and bounds on projector variation and momentum tracking yields the following convergence rates.

Theorem 4.2 (Deterministic and stochastic convergence of ProLand)**.** *Suppose Assumptions 4.1–4.2 and the conditions of Proposition 4.1 hold, with ϵ, ρ, all regularity constants, and all initial-state bounds independent of T. Define $\tau_T(b) := \lceil 2b \log T / |\log q_C| \rceil$ for $b > 0$. The following statements hold for all sufficiently large T.*

(i) Deterministic case. *Use exact gradients, $\beta = 0$, and $\gamma_t \equiv \gamma_T = \theta T^{-1/2}$ for fixed $\theta > 0$. Draw $\widehat{t} \sim \mathrm{Unif}\{\tau_T(1/2), \ldots, T-1\}$ and return $(B_{\widehat{t}}, A_{\widehat{t}})$. Then*

$$\mathbb{E}_{\widehat{t}} \|R_{\widehat{t}}\|_* = O(T^{-1/2}), \qquad \mathbb{E}_{\widehat{t}} \|C_{\widehat{t}}\|_{\mathrm{op}} = O(T^{-1}). \tag{5}$$

(ii) Stochastic case. *Assume $\mathbb{E}[\|\bar{H}_0 - R_0\|_{\mathrm{F}}^2 \mid \mathcal{F}_0] \leq U_0 < \infty$ almost surely, for a deterministic constant U_0 independent of T. Set $1 - \beta_T = c_\beta T^{-1/2}$ and $\gamma_t \equiv \gamma_T = c_\gamma T^{-3/4}$ for fixed $c_\beta, c_\gamma > 0$, keeping both parameters constant during the run. Draw $\widehat{t} \sim \mathrm{Unif}\{\tau_T(3/4), \ldots, T-1\}$, independently of the algorithmic randomness, and return $(B_{\widehat{t}}, A_{\widehat{t}})$. Then*

$$\mathbb{E}\|R_{\widehat{t}}\|_* = O(T^{-1/4}), \qquad \mathbb{E}\|C_{\widehat{t}}\|_{\mathrm{op}} = O(T^{-3/2}). \tag{6}$$

Thus, both guarantees apply to the same randomized output. In (i), expectation is over the output index; in (ii), it also includes the stochastic gradients. All constants hidden in $O(\cdot)$ are independent of T. Appendix E provides the full-horizon averaged-stationarity bounds and uniform pathwise constraint violation bounds on the tail, together with the corresponding proofs.

5 Experiments

5.1 Experimental setup

We begin with the encoder DeBERTa-v3-base (He et al., 2023), evaluated at rank 16 on four GLUE (Wang et al., 2018) validation tasks: RTE, MRPC, STS-B, and CoLA. Across both encoder and decoder evaluations, we compare ProLand with AdamW, LoRA-Muon, fixed-step Manifold-LoRA, per-factor Muon, Riemannion, and sMuon. For DeBERTa, we report means and sample standard deviations across three seeds.

We then evaluate Qwen2.5-3B (Qwen Team, 2024) and Llama-3.2-3B (Grattafiori et al., 2024; Meta, 2024) at adapter ranks $r \in \{4, 8, 16\}$, and Moonlight-16B-A3B (Liu et al., 2025) at rank 16, on seven language benchmarks: MMLU (Hendrycks et al., 2021), PIQA (Bisk et al., 2020), WinoGrande (Sakaguchi et al., 2020), OpenBookQA (Mihaylov et al., 2018), HellaSwag (Zellers et al., 2019), BoolQ (Clark et al., 2019), and ARC-Challenge (Clark et al., 2018). The model set spans the AdamW-pretrained Qwen and Llama backbones and Muon-pretrained Moonlight. Each task uses a separate adapter on a frozen backbone. The decoder tables report zero-shot accuracy and its unweighted mean across the seven tasks. All experimental settings are collected in Appendix L.

5.2 Encoder results on four GLUE tasks

Table 1 reports results for the encoder backbone and task-specific validation metrics. ProLand achieves the highest four-task mean, 84.46 ± 0.38, exceeding the strongest baseline, sMuon, by

0.57 points. It has the highest means on RTE and CoLA. The construction therefore transfers beyond decoder language models. The same update rule also attains the highest four-task mean here.

Table 1: **DeBERTa-v3-base on GLUE validation sets.** Task columns report the mean ± sample standard deviation across seeds; Avg. is the unweighted mean of the four task scores. Bold and underline mark the best and second-best means in each column.

Method	RTE	MRPC	STS-B	CoLA	Avg.
AdamW	79.78 ± 2.25	90.21 ± 0.47	90.89 ± 0.09	64.80 ± 1.13	81.42
LoRA-Muon	83.63 ± 1.82	91.71 ± 0.88	91.74 ± 0.25	67.77 ± 0.82	83.71
Per-factor Muon	83.27 ± 2.35	$\mathbf{92.67 \pm 0.31}$	$\mathbf{91.75 \pm 0.05}$	67.67 ± 1.89	83.84
sMuon	85.08 ± 0.91	91.07 ± 0.66	91.59 ± 0.09	67.83 ± 0.34	83.89
Riemannion	83.75 ± 1.44	91.33 ± 0.41	91.68 ± 0.14	66.97 ± 1.37	83.43
Manifold-LoRA	78.58 ± 1.99	89.52 ± 0.91	90.70 ± 0.12	64.08 ± 1.97	80.72
ProLand (ours)	$\mathbf{85.20 \pm 1.08}$	92.15 ± 1.16	91.64 ± 0.06	$\mathbf{68.85 \pm 1.02}$	**84.46**

5.3 Decoder language benchmark results

ProLand has the highest mean score in all seven reported model–rank settings (Tables 2, 3, and 4), with margins of 0.46 to 1.10 percentage points over the strongest baseline in each setting. Across the seven settings, ProLand is first or second on MMLU in every setting (first in six), and in the top two on WinoGrande and ARC-Challenge in six settings each.

Table 2: **Qwen2.5-3B: language benchmarks.** Zero-shot accuracy (%) at ranks 4, 8, and 16. Avg. is the unweighted mean; bold and underline mark the best and second-best scores within each rank.

Rank	Method	MMLU	PIQA	Wino	OBQA	Hella	BoolQ	ARC-C	Avg.
4	AdamW	65.18	78.73	69.69	42.80	74.01	86.15	49.66	66.60
	LoRA-Muon	63.64	**79.38**	71.03	42.20	72.95	**87.16**	49.32	66.53
	Per-factor Muon	64.76	78.67	70.40	**43.80**	74.37	86.67	48.46	66.73
	sMuon	65.30	78.67	69.93	41.40	**74.63**	86.45	49.06	66.49
	Riemannion	64.25	79.27	70.40	42.40	74.43	86.91	49.40	66.72
	Manifold-LoRA	65.23	79.33	69.06	42.60	74.33	86.42	48.55	66.50
	ProLand (ours)	**65.77**	79.22	**71.35**	43.20	74.46	87.06	**51.88**	**67.56**
8	AdamW	65.28	79.11	69.38	42.60	74.39	86.61	49.74	66.73
	LoRA-Muon	64.96	79.16	69.46	**43.40**	73.75	85.84	49.32	66.55
	Per-factor Muon	64.36	78.51	70.09	42.20	**74.65**	86.57	48.29	66.38
	sMuon	65.45	79.00	70.09	41.80	73.75	84.74	47.35	66.03
	Riemannion	65.33	78.84	70.64	43.00	74.30	86.33	49.06	66.78
	Manifold-LoRA	**65.90**	**79.60**	69.53	41.60	74.43	86.39	51.02	66.92
	ProLand (ours)	65.82	79.54	**71.43**	**43.40**	74.46	**88.17**	**53.33**	**68.02**
16	AdamW	65.15	78.73	70.64	42.00	**75.00**	87.37	49.91	66.97
	LoRA-Muon	63.27	76.01	69.53	**44.00**	73.15	87.09	49.06	66.02
	Per-factor Muon	65.33	78.45	69.14	43.00	74.70	86.79	49.91	66.76
	sMuon	65.31	79.00	68.98	42.80	74.68	87.00	48.81	66.65
	Riemannion	64.93	78.94	70.64	41.60	74.88	**87.89**	49.83	66.96
	Manifold-LoRA	61.63	79.33	69.38	43.40	72.98	86.79	50.60	66.30
	ProLand (ours)	**65.95**	**79.49**	**71.19**	43.40	74.68	87.83	**52.73**	**67.89**

Table 3: **Llama-3.2-3B: language benchmarks.** Zero-shot accuracy (%) at ranks 4, 8, and 16. Avg. is the unweighted mean; bold and underline mark the best and second-best scores within each rank.

Rank	Method	MMLU	PIQA	Wino	OBQA	Hella	BoolQ	ARC-C	Avg.
4	AdamW	54.70	77.80	70.09	42.80	73.43	85.17	47.87	64.55
	LoRA-Muon	52.76	76.01	72.06	42.20	73.02	86.97	42.32	63.62
	Per-factor Muon	54.92	78.18	70.88	**43.60**	73.70	85.75	47.87	64.99
	sMuon	54.68	78.07	70.96	42.80	73.53	85.84	46.84	64.67
	Riemannion	54.23	78.35	71.51	42.40	73.47	85.17	47.95	64.73
	Manifold-LoRA	54.66	79.11	**72.22**	42.20	73.61	85.41	**48.81**	65.14
	ProLand (ours)	**56.20**	**79.16**	71.59	42.80	**73.99**	**87.37**	48.72	**65.69**
8	AdamW	52.29	78.84	70.09	43.00	72.99	85.14	48.98	64.47
	LoRA-Muon	53.46	76.39	72.69	41.80	73.89	**87.77**	43.86	64.26
	Per-factor Muon	51.87	77.97	71.82	42.60	74.34	85.57	**49.23**	64.77
	sMuon	54.02	**79.33**	72.38	41.80	74.44	86.73	48.21	65.27
	Riemannion	55.62	**79.33**	71.43	43.00	74.95	85.50	48.98	65.54
	Manifold-LoRA	53.71	79.16	70.88	41.80	73.66	85.50	46.50	64.46
	ProLand (ours)	**56.69**	79.27	**72.93**	**43.60**	**75.08**	85.90	48.89	**66.05**
16	AdamW	51.57	78.51	71.11	**43.20**	72.14	**87.00**	46.50	64.29
	LoRA-Muon	55.68	78.94	72.14	42.00	73.88	85.57	48.72	65.27
	Per-factor Muon	54.12	78.35	71.27	41.80	74.02	86.64	49.83	65.15
	sMuon	55.01	**79.49**	73.48	42.60	74.55	85.78	49.06	65.71
	Riemannion	54.27	78.45	71.90	40.20	74.67	86.94	**50.85**	65.33
	Manifold-LoRA	52.24	77.69	69.46	39.20	73.70	85.81	46.84	63.56
	ProLand (ours)	**56.75**	79.27	**73.80**	43.00	**75.35**	85.93	49.91	**66.29**

Table 4: **Moonlight-16B-A3B: language benchmarks.** Zero-shot accuracy (%) at LoRA rank 16. Avg. is the unweighted mean; bold and underline mark the best and second-best scores.

Method	MMLU	PIQA	Wino	OBQA	Hella	BoolQ	ARC-C	Avg.
AdamW	55.36	81.07	69.77	44.60	75.00	88.01	54.44	66.89
LoRA-Muon	63.81	**82.48**	73.80	46.40	80.84	**89.42**	61.52	71.18
Per-factor Muon	65.56	82.05	**74.27**	47.60	80.57	87.83	63.91	71.68
sMuon	61.18	81.23	72.61	46.80	77.26	69.79	60.41	67.04
Riemannion	59.41	81.72	73.48	46.00	76.02	87.34	57.08	68.72
Manifold-LoRA	58.88	81.23	71.11	44.60	76.82	88.75	57.08	68.35
ProLand (ours)	**67.85**	81.99	73.88	**47.80**	**81.56**	87.37	**64.51**	**72.14**

Moonlight was pretrained with Muon, unlike the two AdamW-pretrained 3B backbones. Liu et al. (2025) report benefits from retaining Muon across pretraining and full-parameter SFT, and Anson et al. (2026) find supporting evidence for Muon-based LoRA on Moonlight. Here, ProLand exceeds Per-factor Muon, the strongest baseline, by 0.46 percentage points in mean score and LoRA-Muon by 0.96 points. The advantage over the AdamW baseline is more pronounced, reaching 5.25 percentage points in the seven-task mean. These baselines use Muon-style adapter updates on the same pretrained backbone, so the improvement cannot be attributed to the spectral direction alone. Appendix G.2 reviews studies on optimizer compatibility between pretraining and fine-tuning.

5.4 The Role of Landing

We study how Landing affects orthogonality during training and how the resulting constraint violations relate to task accuracy. Appendix L.4 gives the experimental settings.

Constraint-violation trajectories. We compare ProLand with and without Landing, alongside per-factor Muon. In Figure 2, the median constraint violation for ProLand rises initially and then falls. Both controls show increasing median constraint violations late in training. Adding Landing reduces the final median constraint violation, averaged across runs, by 98.1%. The trajectory shows gradual recovery of feasibility. Appendix I gives the final constraint violations and their variation across adapters.

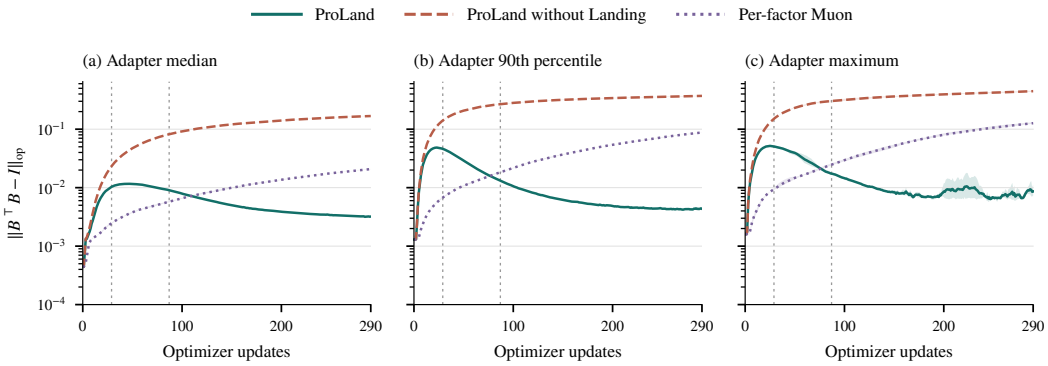

Figure 2: **Landing suppresses orthogonality drift.** Panels show the median, 90th percentile, and maximum of $\|B^\top B - I\|_{\mathrm{op}}$ across adapters. Curves average these statistics across runs; shading shows their range.

Constraint violation and downstream accuracy. Table 5 compares ProLand, the version without Landing, and the variant with retraction under the same conditions.

Adding Landing reduces the median constraint violation by 92.5%, the 90th percentile of constraint violation by 96.8%, and the maximum constraint violation by 96.1%. It controls drift even in the adapters with the largest violations. Retraction attains smaller constraint violations at the cost of a per-step retraction, while ProLand attains the highest mean accuracy. Both experiments show that Landing suppresses drift without a retraction at every update.

Table 5: **Landing ablation under shared conditions.** Mean ± sample SD; bold marks the highest accuracy or lowest constraint violation.

		Constraint violation (10^{-3}) ↓		
Method	Acc. (%) ↑	Median	P90	Maximum
ProLand	$\mathbf{40.73 \pm 0.69}$	7.615 ± 0.540	9.389 ± 0.732	13.556 ± 0.492
Without Landing	38.93 ± 0.61	101.410 ± 0.523	297.497 ± 5.759	349.397 ± 12.500
With retraction	39.67 ± 0.76	$\mathbf{0.506 \pm 0.010}$	$\mathbf{1.282 \pm 0.025}$	$\mathbf{1.552 \pm 0.023}$

6 Conclusion

We introduce ProLand, a retraction-free optimizer for LoRA with B constrained to the Stiefel manifold that combines product-level spectral updates with coordinated factor updates. The basis-factor direction has no first-order effect on the column orthonormality of B, even at infeasible iterates, while a separate Landing correction progressively restores feasibility. An exact low-rank implementation avoids dense product-space optimizer states. Under the stated assumptions and step-size conditions, we establish deterministic and stochastic bounds on product-space stationarity and constraint violation at a common randomized output, assuming exact polar-factor computation. In the main language evaluations, ProLand achieves the highest task-averaged scores among the compared methods in the encoder setting and all seven decoder model–rank configurations. Matched ablations further show that Landing substantially reduces constraint violation.

AI USE STATEMENT

We used large language models (LLMs) to assist with language editing, translation, and writing parts of the code (e.g., plotting scripts and Bash scripts). All AI-assisted text and code were reviewed and checked by the authors, who take full responsibility for the final content of this work.

REPRODUCIBILITY STATEMENT

The assumptions and complete proofs for all theoretical results are provided in Appendices A–E. Appendix B gives the exact low-rank implementation, while Appendices G–L document the experimental setup, hyperparameters, evaluation protocol, and additional diagnostics needed to reproduce the empirical results.

REFERENCES

Pierre Ablin and Gabriel Peyré. Fast and accurate optimization on the orthogonal manifold without retraction. In *International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pp. 5636–5657. PMLR, 2022. URL https://proceedings.mlr.press/v151/ablin22a.html.

Pierre Ablin, Simon Vary, Bin Gao, and Pierre-Antoine Absil. Infeasible Deterministic, Stochastic, and Variance-Reduction Algorithms for Optimization under Orthogonality Constraints. *Journal of Machine Learning Research*, 25(389):1–38, 2024. URL https://jmlr.org/papers/v25/23-0451.html.

P.-A. Absil, Robert Mahony, and Rodolphe Sepulchre. *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, 2008. doi: 10.1515/9781400830244. URL https://press.princeton.edu/books/hardcover/9780691132983/optimization-algorithms-on-matrix-manifolds.

Dmitrii Andriianov, Andrey Veprikov, and Aleksandr Beznosikov. LoRA-TSD: Tangent-Space Spectral Descent for LoRA via Muon-Style Updates. arXiv preprint arXiv:2609.02734v1, 2026. URL https://arxiv.org/abs/2609.02734v1.

Ben Anson, Conor Houghton, and Edward Milsom. Approximate Muon with low-rank adapters. arXiv preprint arXiv:2608.14492v1, 2026. URL https://arxiv.org/abs/2608.14492v1.

Jacob Austin et al. Program synthesis with large language models. arXiv preprint arXiv:2108.07732, 2021. URL https://arxiv.org/abs/2108.07732.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. PIQA: Reasoning about Physical Commonsense in Natural Language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 7432–7439, 2020. doi: 10.1609/aaai.v34i05.6239. URL https://ojs.aaai.org/index.php/AAAI/article/view/6239.

Vladimir Bogachev, Vladimir Aletov, Alexander Molozhavenko, Denis Bobkov, Vera Soboleva, Aibek Alanov, and Maxim Rakhuba. LoRA meets Riemannion: Muon Optimizer for Parametrization-independent Low-Rank Adapters. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/8505a7e1da40320bf5692d7edec5aa45-Abstract-Conference.html.

Léon Bottou, Frank E. Curtis, and Jorge Nocedal. Optimization methods for large-scale machine learning. *SIAM Review*, 60(2):223–311, 2018. doi: 10.1137/16M1080173.

Franz Louis Cesista, Katherine Crowson, Cédric Simal, and Stella Biderman. LoRA-Muon: Spectral Steepest Descent on the Low-Rank Manifold. arXiv preprint arXiv:2606.12921, 2026. URL https://arxiv.org/abs/2606.12921.

Mark Chen et al. Evaluating large language models trained on code. arXiv preprint arXiv:2107.03374, 2021. URL https://arxiv.org/abs/2107.03374.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. BoolQ: Exploring the Surprising Difficulty of Natural Yes/No Questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 2924–2936. Association for Computational Linguistics, 2019. doi: 10.18653/v1/N19-1300. URL https://aclanthology.org/N19-1300/.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have Solved Question Answering? Try ARC, the AI2 Reasoning Challenge. arXiv preprint arXiv:1803.05457, 2018. URL https://arxiv.org/abs/1803.05457.

Saeed Ghadimi and Guanghui Lan. Stochastic first- and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2013. doi: 10.1137/120880811.

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

Soufiane Hayou, Nikhil Ghosh, and Bin Yu. The impact of initialization on LoRA finetuning dynamics. In *Advances in Neural Information Processing Systems*, volume 37, 2024a. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/d4387c37b3b06e55f86eccdb8cd1f829-Abstract-Conference.html.

Soufiane Hayou, Nikhil Ghosh, and Bin Yu. LoRA+: Efficient Low Rank Adaptation of Large Models. arXiv preprint arXiv:2402.12354v2, 2024b. URL https://arxiv.org/abs/2402.12354v2.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. DeBERTaV3: Improving DeBERTa using ELECTRA-Style Pre-Training with Gradient-Disentangled Embedding Sharing. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2111.09543.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring Massive Multitask Language Understanding. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Keller Jordan, Yuchen Jin, Vlado Boza, Jiacheng You, Franz Cesista, Laker Newhouse, and Jeremy Bernstein. Muon: An optimizer for hidden layers in neural networks. Keller Jordan blog, 2024. URL https://kellerjordan.github.io/posts/muon/. December 8, 2024.

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, 2025. doi: 10.52202/085713-2521. Source text: arXiv:2510.01938v2, April 2, 2026.

Kai Lion, Liang Zhang, Bingcong Li, and Niao He. PoLAR: Polar-Decomposed Low-Rank Adapter Representation. arXiv preprint arXiv:2506.03133v2, 2025. URL https://arxiv.org/abs/2506.03133v2.

Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. Is your code generated by ChatGPT really correct? rigorous evaluation of large language models for code generation. arXiv preprint arXiv:2305.01210, 2023. URL https://arxiv.org/abs/2305.01210.

Jingyuan Liu, Jianlin Su, Xingcheng Yao, Zhejun Jiang, Guokun Lai, Yulun Du, Yidao Qin, Weixin Xu, Enzhe Lu, Junjie Yan, Yanru Chen, Huabin Zheng, Yibo Liu, Shaowei Liu, Bohong Yin, Weiran He, Han Zhu, Yuzhi Wang, Jianzhou Wang, Mengnan Dong, Zheng Zhang, Yongsheng Kang, Hao Zhang, Xinran Xu, Yutao Zhang, Yuxin Wu, Xinyu Zhou, and Zhilin Yang. Muon is

Scalable for LLM Training. arXiv preprint arXiv:2502.16982, 2025. URL https://arxiv.org/abs/2502.16982.

Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. DoRA: Weight-Decomposed Low-Rank Adaptation. arXiv preprint arXiv:2402.09353v6, 2024. URL https://arxiv.org/abs/2402.09353v6.

Meta. Llama 3.2 Model Card. Official model card, 2024. URL https://github.com/meta-llama/llama-models/blob/main/models/llama3_2/MODEL_CARD.md.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a Suit of Armor Conduct Electricity? A New Dataset for Open Book Question Answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2381–2391. Association for Computational Linguistics, 2018. doi: 10.18653/v1/D18-1260. URL https://aclanthology.org/D18-1260/.

Zhanfeng Mo, Long-Kai Huang, and Sinno Jialin Pan. Parameter and memory efficient pretraining via low-rank Riemannian optimization. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/16fafdc5e8ab22780873d607ce4387a8-Abstract-Conference.html.

Juneyoung Park, Minjae Kang, Seongbae Lee, Haegang Lee, Seongwan Kim, and Jaeho Lee. Riemannian Optimization for LoRA on the Stiefel Manifold. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 20971–20985. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-emnlp.1143. URL https://aclanthology.org/2025.findings-emnlp.1143/.

Xingyu Qu, Peigeng Huang, and Samuel Horvath. Can Muon fine-tune Adam-pretrained models? arXiv preprint arXiv:2605.10468, 2026. URL https://arxiv.org/abs/2605.10468.

Qwen Team. Qwen2.5 Technical Report. arXiv preprint arXiv:2412.15115, 2024. URL https://arxiv.org/abs/2412.15115.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. WinoGrande: An Adversarial Winograd Schema Challenge at Scale. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 8732–8740, 2020. doi: 10.1609/aaai.v34i05.6399. URL https://ojs.aaai.org/index.php/AAAI/article/view/6399.

Dayana Savostianova, Emanuele Zangrando, Gianluca Ceruti, and Francesco Tudisco. Robust low-rank training via approximate orthonormal constraints. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/d073692637b4fb8c4eb4b81f0fa2df7b-Abstract-Conference.html.

Mikhail Solonko, Alexander Molozhavenko, and Maxim Rakhuba. Muon on the Stiefel Manifold Admits an Exact Closed-Form Update. arXiv preprint arXiv:2608.06218v1, 2026. URL https://arxiv.org/abs/2608.06218v1.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, 2018. URL https://aclanthology.org/W18-5446/.

Zhengbo Wang, Jian Liang, Ran He, Zilei Wang, and Tieniu Tan. LoRA-Pro: Are low-rank adapters properly optimized? In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/ea184f920a0f0f8d8030aa1bd7ac9fd4-Abstract-Conference.html.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a Machine Really Finish Your Sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4791–4800. Association for Computational Linguistics, 2019. doi: 10.18653/v1/P19-1472. URL https://aclanthology.org/P19-1472/.

Qingru Zhang, Minshuo Chen, Alexander Bukharin, Nikos Karampatziakis, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. AdaLoRA: Adaptive Budget Allocation for Parameter-Efficient Fine-Tuning. arXiv preprint arXiv:2303.10512v2, 2023. URL https://arxiv.org/abs/2303.10512v2.

Yuan Zhang, Jiang Hu, Zhijian Lai, Lin Lin, and Zaiwen Wen. Retraction-Free Optimization over the Stiefel Manifold for the LoRA Fine-Tuning. arXiv preprint arXiv:2607.25299v1, 2026. URL https://arxiv.org/abs/2607.25299v1.

Jiacheng Zhu, Kristjan Greenewald, Kimia Nadjahi, Haitz Sáez de Ocáriz Borde, Rickard Brüel Gabrielsson, Leshem Choshen, Marzyeh Ghassemi, Mikhail Yurochkin, and Justin Solomon. Asymmetry in low-rank adapters of foundation models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 62369–62385. PMLR, 2024.

A PRODUCT GEOMETRY AND FACTOR REALIZATIONS

This appendix collects the geometric results underlying the product-space formulation of ProLand. Section A.1 shows that the parameterization with B constrained to the Stiefel manifold preserves the full rank-at-most-r LoRA class and characterizes the first-order product space and its orthogonal projector. Section A.2 relates the projected gradient and product-space stationarity measure used in the convergence analysis to the first-order conditions of the factorized constrained problem. Section A.3 shows that every realizable product direction admits a factor realization satisfying the orthogonality condition required by the feasibility analysis, and establishes the reconstruction identity used by the exact low-rank implementation.

A.1 ONE-SIDED STIEFEL PARAMETERIZATION AND PRODUCT GEOMETRY

We first prove the expressivity identity stated in Section 2. For any $W \in \mathbb{R}^{m\times n}$ with $1 \le k := \operatorname{rank}(W) \le r$, let $W = U_k\Sigma_k V_k^\top$ be its compact singular value decomposition. Complete U_k to an $m \times r$ matrix with orthonormal columns and set

$$B = [U_k \;\; U_c] \in \operatorname{St}(m, r), \qquad A = \begin{bmatrix} \Sigma_k V_k^\top \\ 0 \end{bmatrix}.$$

Then $BA = W$. If $W = 0$, take any $B \in \operatorname{St}(m, r)$ and $A = 0$. Conversely, every admissible factor pair satisfies $\operatorname{rank}(BA) \le r$. Hence

$$\{BA : B \in \operatorname{St}(m, r),\ A \in \mathbb{R}^{r\times n}\} = \{W \in \mathbb{R}^{m\times n} : \operatorname{rank}(W) \le r\}.$$

Moreover, for feasible B, $(BA)^\top(BA) = A^\top B^\top BA = A^\top A$. Thus A and BA have the same nonzero singular values: the orthogonality constraint normalizes the basis factor B without fixing the scale or singular values of the represented product.

Lemma A.1 (Product differential image and projector)**.** *Let $B \in \mathbb{R}^{m\times r}$ and $A \in \mathbb{R}^{r\times n}$ have arbitrary ranks. Then*

$$\mathcal{S}_{B,A} = \operatorname{range} \mathrm{D}\Phi(B, A) = \{Z \in \mathbb{R}^{m\times n} : (I - P_B)Z(I - P_A) = 0\}.$$

The map

$$\Pi_{B,A}Z = P_B Z + (I - P_B)ZP_A$$

is the Frobenius-orthogonal projector onto $\mathcal{S}_{B,A}$. Moreover,

$$Z \in \mathcal{S}_{B,A} \implies \operatorname{rank}(Z) \le 2r, \qquad \|\Pi_{B,A}Z\|_{\mathrm{F}} \le \|Z\|_{\mathrm{F}} \quad \textit{for all } Z \in \mathbb{R}^{m\times n}.$$

Proof. If $Z \in \mathcal{S}_{B,A}$, then $Z = \Delta BA + B\Delta A$ for some factor perturbations. Since $A(I - P_A) = 0$ and $(I - P_B)B = 0$, $(I - P_B)Z(I - P_A) = 0$. Thus

$$\mathcal{S}_{B,A} \subseteq \{Z : (I - P_B)Z(I - P_A) = 0\}.$$

Conversely, if $(I - P_B)Z(I - P_A) = 0$, the four-block decomposition gives

$$\begin{aligned} Z &= P_B ZP_A + P_B Z(I - P_A) + (I - P_B)ZP_A \\ &\quad + (I - P_B)Z(I - P_A) \\ &= P_B Z + (I - P_B)ZP_A \\ &= B(B^\dagger Z) + \big((I - P_B)ZA^\dagger\big)A. \end{aligned}$$

This has the form $B\Delta A + \Delta BA$, so $Z \in \mathcal{S}_{B,A}$, proving the characterization.

For arbitrary Z, $(I - P_B)(\Pi_{B,A}Z)(I - P_A) = 0$. By the characterization above, $\Pi_{B,A}Z \in \mathcal{S}_{B,A}$, hence $\operatorname{range}\Pi_{B,A} \subseteq \mathcal{S}_{B,A}$. For $Y \in \mathcal{S}_{B,A}$,

$$Y - \Pi_{B,A}Y = (I - P_B)Y(I - P_A) = 0.$$

Therefore $Y = \Pi_{B,A}Y \in \operatorname{range}\Pi_{B,A}$, and hence

$$\operatorname{range}\Pi_{B,A} = \mathcal{S}_{B,A}.$$

Since $\Pi_{B,A}$ maps into $\mathcal{S}_{B,A}$ and acts as the identity on this space, $\Pi_{B,A}^2 = \Pi_{B,A}$; hence it is a projection onto $\mathcal{S}_{B,A}$.

For arbitrary Z, $Z - \Pi_{B,A}Z = (I - P_B)Z(I - P_A)$. For every $Y \in \mathcal{S}_{B,A}$,

$$\begin{aligned}\langle Z - \Pi_{B,A}Z, Y\rangle_{\mathrm{F}} &= \mathrm{tr}\big(Z^\top (I - P_B)Y(I - P_A)\big)\\ &= 0,\end{aligned}$$

where the last equality uses $(I - P_B)Y(I - P_A) = 0$. Hence $Z - \Pi_{B,A}Z \in \mathcal{S}_{B,A}^{\perp}$, proving that $\Pi_{B,A}$ is the Frobenius-orthogonal projector onto $\mathcal{S}_{B,A}$.

For $Z \in \mathcal{S}_{B,A}$,

$$\mathrm{rank}(Z) \le \mathrm{rank}(P_B Z) + \mathrm{rank}((I - P_B)ZP_A) \le 2r.$$

Finally, $\|Z\|_{\mathrm{F}}^2 = \|\Pi_{B,A}Z\|_{\mathrm{F}}^2 + \|Z - \Pi_{B,A}Z\|_{\mathrm{F}}^2$, hence $\|\Pi_{B,A}Z\|_{\mathrm{F}} \le \|Z\|_{\mathrm{F}}$. □

A.2 Product-Space and Factor-Space Stationarity

Proposition A.2 (Equivalent first-order conditions)**.** *Let $B \in \mathrm{St}(m, r)$ and $A \in \mathbb{R}^{r\times n}$, and suppose f is differentiable at BA. For $F := \nabla f(BA)$ and $R := \Pi_{B,A}F$,*

$$R = 0 \quad \Longleftrightarrow \quad B^\top F = 0 \text{ and } FA^\top = 0.$$

Under the induced Frobenius metric,

$$R = 0 \quad \Longleftrightarrow \quad \nabla_A f(BA) = 0 \text{ and } \mathrm{grad}_B\, f(BA) = 0.$$

Consequently, at feasible factors, the stationarity measure $\|R\|_*$ used in Section 4 vanishes exactly at first-order stationary points of the factor problem with B constrained to the Stiefel manifold.

Proof. By Lemma A.1, $R = 0$ if and only if $F \perp \mathcal{S}_{B,A}$. The following argument applies to arbitrary-rank factors, without requiring B to be feasible. For arbitrary perturbations,

$$\begin{aligned}\langle F, \mathrm{D}\Phi(B, A)[\Delta B, \Delta A]\rangle_{\mathrm{F}} &= \langle F, \Delta B A + B\Delta A\rangle_{\mathrm{F}}\\ &= \langle FA^\top, \Delta B\rangle_{\mathrm{F}} + \langle B^\top F, \Delta A\rangle_{\mathrm{F}}.\end{aligned}$$

Since ΔB and ΔA can be chosen independently, $F \perp \mathcal{S}_{B,A}$ if and only if $FA^\top = 0$ and $B^\top F = 0$.

The factor gradients are

$$\nabla_A f(BA) = B^\top F, \qquad \mathrm{grad}_B\, f(BA) = FA^\top - B\,\mathrm{sym}(B^\top FA^\top).$$

If $B^\top F = 0$, then $B^\top FA^\top = (B^\top F)A^\top = 0$, so $\mathrm{grad}_B\, f(BA) = FA^\top$. This proves the second equivalence. Finally, the nuclear norm vanishes if and only if its argument is zero, so $\|R\|_* = 0$ has the same zero set. □

A.3 Factor Realization and Reconstruction

The feasibility analysis in Section 4 requires a factor realization whose B-component satisfies $B^\top \Delta B = 0$. The following result shows that imposing this condition does not restrict the first-order product directions in $\mathcal{S}_{B,A}$.

Lemma A.3 (Compatible factor realization and reconstruction)**.** *Let $B \in \mathbb{R}^{m\times r}$ and $A \in \mathbb{R}^{r\times n}$ have arbitrary ranks. Then*

$$\mathcal{S}_{B,A} = \left\{\Delta BA + B\Delta A : B^\top \Delta B = 0\right\}.$$

Define

$$\mathcal{R}_{B,A}(M_A, M_B) = (B^\dagger)^\top M_A + (I - P_B)M_B(A^\dagger)^\top. \tag{7}$$

If $M_A = B^\top Z$ and $M_B = ZA^\top$ are the factor coordinates of a common matrix Z, then

$$\mathcal{R}_{B,A}(B^\top Z, ZA^\top) = \Pi_{B,A}Z.$$

Proof. The restricted perturbations yield elements of $\mathcal{S}_{B,A}$ by the definition of $\mathrm{D}\Phi$. Conversely, given any factor perturbation $(\Delta B, \Delta A)$, set

$$\Delta B' = (I - P_B)\Delta B, \qquad \Delta A' = \Delta A + B^\dagger \Delta BA.$$

Since $B^\top(I - P_B) = 0$,

$$\begin{aligned} B^\top \Delta B' &= 0, \\ \Delta B' A + B \Delta A' &= (I - P_B)\Delta B A + B \Delta A + P_B \Delta B A \\ &= \Delta B A + B \Delta A. \end{aligned}$$

Thus every element of $\mathcal{S}_{B,A}$ admits a restricted realization, proving the set equality.

For the reconstruction identity, using $(B^\dagger)^\top B^\top = P_B$ and $A^\top (A^\dagger)^\top = P_A$ gives

$$\begin{aligned} \mathcal{R}_{B,A}(B^\top Z, Z A^\top) &= P_B Z + (I - P_B) Z P_A \\ &= \Pi_{B,A} Z. \end{aligned}$$

□

Least-squares characterization and the selected factor directions. For the current factors and polar factor Y_t in Algorithm 1, the product-space projection has the equivalent least-squares characterization

$$(D^\star_{A,t}, D^\star_{B,t}) \in \operatorname*{arg\,min}_{D_A, D_B} \|Y_t - (B_t D_A + D_B A_t)\|_{\mathrm{F}}^2 .$$

Here $D_A \in \mathbb{R}^{r \times n}$ and $D_B \in \mathbb{R}^{m \times r}$. Since $B_t D_A + D_B A_t$ ranges over $\mathcal{S}_t$, Lemma A.1 gives the unique optimal product direction $\xi_t = \Pi_t Y_t$. Indeed, for every factor pair,

$$\|Y_t - (B_t D_A + D_B A_t)\|_{\mathrm{F}}^2 = \|Y_t - \xi_t\|_{\mathrm{F}}^2 + \|\xi_t - (B_t D_A + D_B A_t)\|_{\mathrm{F}}^2 .$$

Thus the product direction is unique, but the minimizing factor pair need not be unique. Algorithm 1 selects

$$D_{A,t} = B_t^\dagger Y_t, \qquad D_{B,t} = (I - P_{B_t}) Y_t A_t^\dagger.$$

Using $P_{B_t} = B_t B_t^\dagger$, $P_{A_t} = A_t^\dagger A_t$, and $B_t^\top (I - P_{B_t}) = 0$ yields

$$\begin{aligned} B_t D_{A,t} + D_{B,t} A_t = P_{B_t} Y_t + (I - P_{B_t}) Y_t P_{A_t} = \Pi_t Y_t = \xi_t, \\ B_t^\top D_{B,t} = 0. \end{aligned}$$

These identities hold for arbitrary-rank factors, whether or not B_t is feasible. The selected pair therefore also solves the least-squares problem with the additional condition $B_t^\top D_B = 0$, with the same optimal value and product direction. This is a characterization of the existing factor directions, not an additional numerical solve in Algorithm 1.

B Exact low-rank implementation

Using the reconstruction map $\mathcal{R}_{B,A}$ in (7), Algorithm 1 admits an exact factor-coordinate realization without forming dense product-space states. Algorithm 2 stores the EMA state in thin-factor form, reconstructs both $\bar{H}_t$ and the polar-factor input J_t from factor coordinates, and evaluates the polar factor through a core whose row and column dimensions are each at most $2r$.

Algorithm 2 Exact factor-coordinate realization of ProLand

Require: Initial factors B_0, A_0; thin EMA factors $L_{H,-1}, R_{H,-1}$ with $L_{H,-1}R_{H,-1}^\top = \bar{H}_{-1}$ and width at most $2r$; horizon T; $\beta \in [0,1)$, $\rho > 0$, steps $\gamma_t > 0$

1: **for** $t = 0, \dots, T-1$ **do**

Coordinate Recursions

2: Obtain factor gradients $g_{A,t}, g_{B,t}$ at (B_t, A_t).

3: $\widetilde{M}_{A,t} \leftarrow (B_t^\top L_{H,t-1})R_{H,t-1}^\top$;
$\widetilde{M}_{B,t} \leftarrow L_{H,t-1}(R_{H,t-1}^\top A_t^\top)$

4: $M_{F,t} \leftarrow \beta\widetilde{M}_{F,t} + (1-\beta)g_{F,t}, \quad F \in \{A, B\}$;
$\widehat{M}_{F,t} \leftarrow \beta M_{F,t} + (1-\beta)g_{F,t}, \quad F \in \{A, B\}$

Thin Reconstructions

5: Compute exact pseudoinverses $B_t^\dagger, A_t^\dagger$.

6: $L_{H,t} \leftarrow [(B_t^\dagger)^\top,\ M_{B,t} - B_t(B_t^\dagger M_{B,t})]$;
$R_{H,t} \leftarrow [M_{A,t}^\top,\ A_t^\dagger]$

7: $L_{J,t} \leftarrow [(B_t^\dagger)^\top,\ \widehat{M}_{B,t} - B_t(B_t^\dagger \widehat{M}_{B,t})]$;
$R_{J,t} \leftarrow [\widehat{M}_{A,t}^\top,\ A_t^\dagger]$

Small-Core Polar Factor

8: Compute thin QR factorizations $L_{J,t} = Q_L L_c$ and $R_{J,t} = Q_R R_c$, with orthonormal columns in Q_L, Q_R.

9: $K_c \leftarrow L_c R_c^\top$

10: Compute the compact SVD $K_c = U_c \Sigma_c V_c^\top$.

11: $U_{Y,t} \leftarrow Q_L U_c$; $V_{Y,t} \leftarrow Q_R V_c$.
If $K_c = 0$, use zero-rank factors so that $U_{Y,t}V_{Y,t}^\top = 0$.

Factor Directions and Landing

12: $D_{A,t} \leftarrow (B_t^\dagger U_{Y,t})V_{Y,t}^\top$.

13: $D_{B,t} \leftarrow [U_{Y,t} - B_t(B_t^\dagger U_{Y,t})](V_{Y,t}^\top A_t^\dagger)$

14: $A_{t+1} \leftarrow A_t - \gamma_t D_{A,t}$;
$B_{t+1} \leftarrow B_t - \gamma_t D_{B,t} - \rho B_t(B_t^\top B_t - I_r)$

15: Carry $L_{H,t}, R_{H,t}$ to the next iteration.

16: **end for**

Proposition B.1 (Exact factor-coordinate realization at arbitrary rank)**.** *Consider Algorithms 1 and 2 with the same initial factors B_0, A_0, horizon T, and parameters $\beta \in [0,1)$, $\rho > 0$, $\gamma_t > 0$. Suppose $L_{H,-1}R_{H,-1}^\top = \bar{H}_{-1}$ with width at most $2r$, and $g_{A,t} = B_t^\top G_t$, $g_{B,t} = G_t A_t^\top$ for the same product-space input G_t at every $0 \le t < T$. In exact arithmetic, Algorithm 2 generates the same factor iterates as Algorithm 1 and satisfies*

$$L_{H,t}R_{H,t}^\top = \bar{H}_t, \qquad U_{Y,t}V_{Y,t}^\top = \mathrm{polar}(J_t)$$

at every iteration. The reduced core has dimensions at most $2r \times 2r$, with possibly unequal row and column dimensions; rank-deficient factors and the zero-core case are included.

Proof. Assume inductively that $L_{H,t-1}R_{H,t-1}^\top = \bar{H}_{t-1}$, and define

$$Z_t := \beta\bar{H}_{t-1} + (1-\beta)G_t.$$

The supplied initial thin representation and common initial factors give the induction base. At the common factor pair (B_t, A_t), the coordinate recursions give

$$M_{A,t} = B_t^\top Z_t, \qquad M_{B,t} = Z_t A_t^\top.$$

By the reconstruction identity in Appendix A.3, using the map in (7),

$$\mathcal{R}_{B_t,A_t}(M_{A,t}, M_{B,t}) = \Pi_t Z_t = \bar{H}_t.$$

The second coordinate recursion gives

$$\widehat{M}_{A,t} = B_t^\top[\beta Z_t + (1-\beta)G_t], \qquad \widehat{M}_{B,t} = [\beta Z_t + (1-\beta)G_t]A_t^\top.$$

Applying the same reconstruction identity,

$$\begin{aligned}\mathcal{R}_{B_t,A_t}(\widehat{M}_{A,t}, \widehat{M}_{B,t}) &= \Pi_t[\beta Z_t + (1-\beta)G_t] \\ &= \beta\bar{H}_t + (1-\beta)\Pi_t G_t \\ &= J_t.\end{aligned}$$

The thin-factor definitions yield

$$\begin{aligned}L_{H,t}R_{H,t}^\top &= (B_t^\dagger)^\top M_{A,t} + (I - P_{B_t})M_{B,t}(A_t^\dagger)^\top \\ &= \mathcal{R}_{B_t,A_t}(M_{A,t}, M_{B,t}) \\ &= \bar{H}_t.\end{aligned}$$

The identical calculation with the hatted coordinates gives $L_{J,t}R_{J,t}^\top = J_t$.

Let $L_{J,t} = Q_L L_c$, $R_{J,t} = Q_R R_c$, and $K_c = L_c R_c^\top$. The thin QR factorizations give

$$J_t = Q_L K_c Q_R^\top.$$

The row and column dimensions of K_c are each at most $2r$ and may be unequal. If $K_c = U_c\Sigma_c V_c^\top$ is its compact SVD, then

$$J_t = (Q_L U_c)\Sigma_c(Q_R V_c)^\top.$$

Since Q_L, Q_R have orthonormal columns and U_c, V_c are the compact singular-vector factors of the core, this is a compact SVD of J_t. Therefore,

$$U_{Y,t}V_{Y,t}^\top = Q_L U_c V_c^\top Q_R^\top = Q_L\,\mathrm{polar}(K_c)Q_R^\top = \mathrm{polar}(J_t).$$

If $K_c = 0$, then $J_t = 0$ and the zero-rank factors agree with $\mathrm{polar}(0) = 0$.

Define $Y_t := U_{Y,t}V_{Y,t}^\top = \mathrm{polar}(J_t)$. Then

$$D_{A,t} = (B_t^\dagger U_{Y,t})V_{Y,t}^\top = B_t^\dagger Y_t,$$

and

$$\begin{aligned}D_{B,t} &= [U_{Y,t} - B_t(B_t^\dagger U_{Y,t})](V_{Y,t}^\top A_t^\dagger) \\ &= (I - P_{B_t})Y_t A_t^\dagger.\end{aligned}$$

These are exactly the factor directions in Algorithm 1. Therefore the simultaneous A- and B-updates coincide, closing the induction. □

The dense matrices $\bar{H}_t$, J_t, and Y_t are represented only through thin factors. Thin matrix products, pseudoinverses, and QR factorizations require $O((m+n)r^2)$ arithmetic up to $r \times r$ core factorizations, while all reduced core factorizations, including the polar-factor computation, require $O(r^3)$. Hence, excluding gradient evaluation, the exact realization requires $O((m+n)r^2 + r^3)$ arithmetic operations per iteration and $O((m+n)r)$ additional storage. Only the thin EMA factors $L_{H,t}, R_{H,t}$ are carried across iterations.

C Pathwise Feasibility and Safe-Region Analysis

This appendix proves Proposition 4.1. We first derive the exact Gram recursion and safe-region invariance, then collect the conditioning consequences used later and establish a product-preserving feasible reparameterization. We conclude by showing why pointwise full row rank of A_t is insufficient for a uniform safe-step bound.

C.1 Exact Gram Recursion and Gram-Polynomial Contraction

Suppressing the iteration index, consider the basis-factor update

$$B^+ = B - \gamma D - \rho BC = B(I_r - \rho C) - \gamma D, \qquad C := B^\top B - I_r, \qquad K := B^\top D,$$

where D denotes the basis-factor direction D_B realizing the optimization update. Since $B^\top B = I_r + C$ and $C = C^\top$, we obtain

$$\begin{aligned} C^+ &:= (B^+)^\top B^+ - I_r \\ &= (I_r - \rho C)(I_r + C)(I_r - \rho C) - I_r - \gamma\big[(I_r - \rho C)K + K^\top(I_r - \rho C)\big] + \gamma^2 D^\top D. \end{aligned}$$

The first term in the recursion is generated solely by the Landing correction. Expanding it gives

$$\begin{aligned} p_\rho(C) &:= (I_r - \rho C)(I_r + C)(I_r - \rho C) - I_r \\ &= (1 - 2\rho)C + (\rho^2 - 2\rho)C^2 + \rho^2 C^3. \end{aligned}$$

Hence, in the absence of the optimization step ($\gamma = 0$), the feasibility error evolves exactly as $C^+ = p_\rho(C)$. The linear coefficient $1 - 2\rho$ captures the local contraction induced by Landing, while the quadratic and cubic terms account for the off-manifold correction. Lemma C.1 turns this local picture into a uniform contraction bound over the safe region $\|C\|_{\mathrm{op}} \le \epsilon$.

The first-order interaction with the optimization direction reveals why the stronger identity $B^\top D_B = 0$ is useful. With $K = B^\top D$ as defined above,

$$-\gamma\big[(I_r - \rho C)K + K^\top(I_r - \rho C)\big] = -\gamma(K + K^\top) + \gamma\rho(CK + K^\top C).$$

Suppose only the skew-symmetry condition $B^\top D + D^\top B = 0$ is imposed, which is the usual Stiefel tangency relation when B lies on the manifold. Then $K^\top = -K$, so the pure $O(\gamma)$ term vanishes, but the mixed term becomes $\gamma\rho(CK - KC)$. This commutator vanishes when $C = 0$, but need not vanish away from the manifold. By contrast, the ProLand factor realization satisfies $B_t^\top D_{B,t} = 0$, and hence $K = 0$ without requiring $C_t = 0$. Therefore, all terms that are first order in γ_t disappear exactly, and the Gram recursion reduces to

$$C_{t+1} = p_\rho(C_t) + \gamma_t^2 D_{B,t}^\top D_{B,t}.$$

Thus, the optimization step along $D_{B,t}$ contributes to the feasibility error only through the second-order term $\gamma_t^2 D_{B,t}^\top D_{B,t}$.

Lemma C.1 (Contraction of the Gram polynomial)**.** *Let $0 < \epsilon < 1$ and $0 < \rho \le 1/2$. For any symmetric $C \in \mathbb{R}^{r\times r}$ satisfying $\|C\|_{\mathrm{op}} \le \epsilon$, the Gram polynomial p_ρ defined above satisfies*

$$\|p_\rho(C)\|_{\mathrm{op}} \le q_C \|C\|_{\mathrm{op}}, \qquad \|p_\rho(C)\|_{\mathrm{F}} \le q_C \|C\|_{\mathrm{F}},$$

where

$$q_C := 1 - 2\rho + (2\rho - \rho^2)\epsilon + \rho^2\epsilon^2 = 1 - (1 - \epsilon)(2\rho + \rho^2\epsilon) \in (0, 1).$$

Proof. Let $g_\rho(c) := 1 - 2\rho + (\rho^2 - 2\rho)c + \rho^2 c^2$, so that $p_\rho(c) = c\, g_\rho(c)$. For every $c \in [-\epsilon, \epsilon]$, the condition $0 < \rho \le 1/2$ gives

$$\begin{aligned} |g_\rho(c)| &\le 1 - 2\rho + (2\rho - \rho^2)|c| + \rho^2 c^2 \\ &\le 1 - 2\rho + (2\rho - \rho^2)\epsilon + \rho^2\epsilon^2 = q_C. \end{aligned}$$

The bound is attained at $c = -\epsilon$. Moreover, $q_C = 1 - (1 - \epsilon)(2\rho + \rho^2\epsilon) < 1$, while its first expression gives $q_C > 0$.

Let $C = U \operatorname{diag}(c_1, \dots, c_r) U^\top$ be an eigendecomposition. Since each $c_i \in [-\epsilon, \epsilon]$ and p_ρ is a polynomial,

$$p_\rho(C) = U \operatorname{diag}\big(p_\rho(c_1), \dots, p_\rho(c_r)\big) U^\top.$$

Therefore,

$$\|p_\rho(C)\|_{\mathrm{op}} = \max_i |p_\rho(c_i)| \le q_C \max_i |c_i| = q_C \|C\|_{\mathrm{op}},$$

and similarly

$$\|p_\rho(C)\|_{\mathrm{F}}^2 = \sum_{i=1}^r |p_\rho(c_i)|^2 \le q_C^2 \sum_{i=1}^r |c_i|^2 = q_C^2 \|C\|_{\mathrm{F}}^2.$$

Taking square roots proves the Frobenius-norm bound. □

C.2 Safe-Region Invariance and Induced Conditioning

Proof of Proposition 4.1. Under Assumption 4.2, $\sigma_r(A_t) \geq a_- > 0$ for every $0 \leq t \leq T$, and hence $\|A_t^\dagger\|_{\mathrm{op}} \leq a_-^{-1}$. Since $Y_t = \mathrm{polar}(J_t)$ satisfies $\|Y_t\|_{\mathrm{op}} \leq 1$ and $I - P_{B_t}$ is an orthogonal projector, the definition $D_{B,t} = (I - P_{B_t})Y_t A_t^\dagger$ gives

$$\|D_{B,t}\|_{\mathrm{op}} \leq a_-^{-1}, \qquad \|D_{B,t}^\top D_{B,t}\|_{\mathrm{op}} \leq a_-^{-2}, \qquad \|D_{B,t}^\top D_{B,t}\|_{\mathrm{F}} \leq \sqrt{r}\, a_-^{-2}.$$

Whenever $\|C_t\|_{\mathrm{op}} \leq \epsilon$, the exact Gram recursion and Lemma C.1 give

$$\|C_{t+1}\|_{\mathrm{op}} \leq q_C \|C_t\|_{\mathrm{op}} + a_-^{-2}\gamma_t^2,$$
$$\|C_{t+1}\|_{\mathrm{F}} \leq q_C \|C_t\|_{\mathrm{F}} + \sqrt{r}\, a_-^{-2}\gamma_t^2.$$

Under the step-size condition in Proposition 4.1, $\gamma_t \leq a_- \sqrt{\epsilon(1-q_C)}$, so $a_-^{-2}\gamma_t^2 \leq (1-q_C)\epsilon$. Thus $\|C_t\|_{\mathrm{op}} \leq \epsilon$ implies $\|C_{t+1}\|_{\mathrm{op}} \leq q_C\epsilon + (1-q_C)\epsilon = \epsilon$. Together with $\|C_0\|_{\mathrm{op}} \leq \epsilon$, induction yields

$$\|C_t\|_{\mathrm{op}} \leq \epsilon, \qquad 0 \leq t \leq T.$$

Since $C_t = B_t^\top B_t - I_r$, it follows that

$$(1-\epsilon)I_r \preceq B_t^\top B_t \preceq (1+\epsilon)I_r,$$

and therefore

$$\sqrt{1-\epsilon} \leq \sigma_r(B_t) \leq \sigma_1(B_t) \leq \sqrt{1+\epsilon}, \qquad \|B_t^\dagger\|_{\mathrm{op}} \leq (1-\epsilon)^{-1/2}.$$

For a constant step size $\gamma_t \equiv \gamma$, iterating the operator-norm recursion gives

$$\|C_t\|_{\mathrm{op}} \leq q_C^t \|C_0\|_{\mathrm{op}} + \frac{\gamma^2(1-q_C^t)}{a_-^2(1-q_C)}, \tag{8}$$

which is the bound stated in Proposition 4.1. For later use, the same argument gives the Frobenius-norm counterpart

$$\|C_t\|_{\mathrm{F}} \leq q_C^t \|C_0\|_{\mathrm{F}} + \frac{\sqrt{r}\,\gamma^2(1-q_C^t)}{a_-^2(1-q_C)}. \tag{9}$$

□

Empirical control of the B spectrum. We examine how Landing controls the spectrum of B during Qwen2.5-3B fine-tuning on CodeAlpaca. We compare Landing strengths under linear decay and constant learning rates, including the same update with Landing removed. Training settings are collected in Appendix L.4.

Figure 3 shows that Landing suppresses constraint violation under both schedules. With a constant learning rate, the final adapter-median constraint violation, averaged across runs, falls from 1.29×10^{-1} without Landing to 1.69×10^{-3} at $\rho = 0.05$, a reduction of approximately $76\times$. The improvement extends to the worst adapter. Through the singular-value bounds in Proposition 4.1, this smaller constraint violation keeps the singular values of B closer to one.

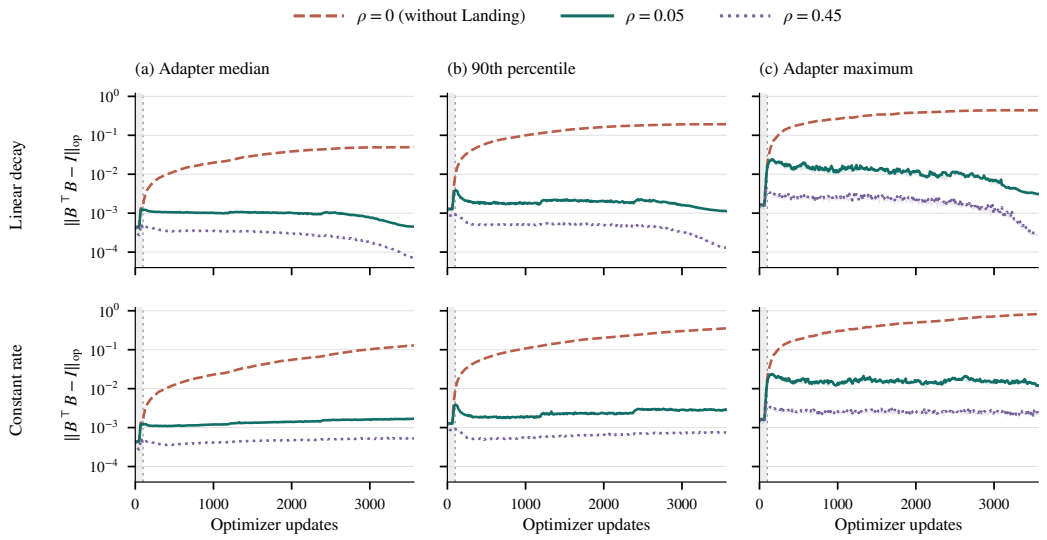

Figure 3: **Landing controls constraint violation of B.** Columns show the adapter median, 90th percentile, and maximum; rows compare linear decay and constant learning rates. Median and percentile curves show means across runs; maximum curves show the worst run and adapter. Shading spans the run-to-run range, and gray regions mark warmup.

C.3 Additional Conditioning Consequences

Proposition 4.1 already provides uniform conditioning of B_t. We record several additional consequences used in the subsequent analysis.

Proposition C.2 (Product conditioning and Stiefel distance)**.** *Under the conditions of Proposition 4.1, for every $0 \le t \le T$,*

$$\left\|(B_t^\top B_t)^{-1} - I_r\right\|_{\mathrm{op}} \le \frac{\|C_t\|_{\mathrm{op}}}{1 - \|C_t\|_{\mathrm{op}}}.$$

Moreover,

$$\frac{\|C_t\|_{\mathrm{F}}}{1+\sqrt{1+\epsilon}} \le \mathrm{dist}_{\mathrm{F}}(B_t, \mathrm{St}(m,r)) \le \frac{\|C_t\|_{\mathrm{F}}}{1+\sqrt{1-\epsilon}}. \tag{10}$$

For $i = 1, \dots, r$, the singular values of $W_t = B_t A_t$ satisfy

$$\sqrt{1-\epsilon}\,\sigma_i(A_t) \le \sigma_i(W_t) \le \sqrt{1+\epsilon}\,\sigma_i(A_t).$$

Consequently, Assumption 4.2 yields

$$\sigma_r(W_t) \ge \sqrt{1-\epsilon}\,a_-, \qquad \|W_t\|_{\mathrm{op}} \le \sqrt{1+\epsilon}\,A_{\max}, \qquad A_t = B_t^\dagger W_t.$$

Proof. Let c_i denote an eigenvalue of $C_t = B_t^\top B_t - I_r$. Proposition 4.1 gives $|c_i| \le \|C_t\|_{\mathrm{op}} < 1$. The corresponding eigenvalue of $(B_t^\top B_t)^{-1} - I_r$ is $-c_i/(1+c_i)$, and hence

$$\left|\frac{c_i}{1+c_i}\right| \le \frac{\|C_t\|_{\mathrm{op}}}{1 - \|C_t\|_{\mathrm{op}}}.$$

Let s_i denote the singular values of B_t. The nearest Stiefel point is the polar factor of B_t, so $\mathrm{dist}_{\mathrm{F}}^2(B_t, \mathrm{St}(m,r)) = \sum_{i=1}^r (s_i - 1)^2$. Since $|s_i - 1| = |s_i^2 - 1|/(s_i + 1)$ and Proposition 4.1 gives $\sqrt{1-\epsilon} \le s_i \le \sqrt{1+\epsilon}$,

$$\frac{|s_i^2 - 1|}{1+\sqrt{1+\epsilon}} \le |s_i - 1| \le \frac{|s_i^2 - 1|}{1+\sqrt{1-\epsilon}}.$$

Squaring and summing over i gives (10).

Finally, $W_t^\top W_t = A_t^\top B_t^\top B_t A_t$, and therefore

$$(1-\epsilon)A_t^\top A_t \preceq W_t^\top W_t \preceq (1+\epsilon)A_t^\top A_t.$$

Eigenvalue monotonicity yields the stated singular-value bounds. Combining them with Assumption 4.2 gives the uniform bounds on $\sigma_r(W_t)$ and $\|W_t\|_{\mathrm{op}}$. Proposition 4.1 also ensures that B_t has full column rank, so $B_t^\dagger W_t = B_t^\dagger B_t A_t = A_t$. □

C.4 Product-Preserving Feasible Reparameterization

This subsection establishes a product-preserving feasible reparameterization within the safe region. The result uses only the upper bound $\|A\|_{\mathrm{op}} \le A_{\max}$ and remains valid even if the lower-regularity condition $\sigma_r(A) \ge a_-$ in Assumption 4.2 fails.

Proposition C.3 (Product-preserving feasible reparameterization)**.** *Let $B \in \mathbb{R}^{m\times r}$ and $A \in \mathbb{R}^{r\times n}$ satisfy $\|B^\top B - I_r\|_{\mathrm{op}} \le \epsilon < 1$ and $\|A\|_{\mathrm{op}} \le A_{\max}$, and suppose f is differentiable at BA. Write $C := B^\top B - I_r$ and $R := \Pi_{B,A}\nabla f(BA)$, and define*

$$\widehat{B} = B(B^\top B)^{-1/2}, \qquad \widehat{A} = (B^\top B)^{1/2}A. \tag{11}$$

Then

$$\widehat{B}^\top \widehat{B} = I_r, \qquad \widehat{B}\widehat{A} = BA, \qquad \Pi_{\widehat{B},\widehat{A}} = \Pi_{B,A}.$$

Defining $\widehat{R} := \Pi_{\widehat{B},\widehat{A}}\nabla f(\widehat{B}\widehat{A})$, we further have

$$\widehat{R} = R, \qquad f(\widehat{B}\widehat{A}) = f(BA).$$

Moreover,

$$\|B-\widehat{B}\|_{\mathrm{F}} \le \frac{\|C\|_{\mathrm{F}}}{1+\sqrt{1-\epsilon}}, \qquad \|A-\widehat{A}\|_{\mathrm{F}} \le \frac{A_{\max}\|C\|_{\mathrm{F}}}{1+\sqrt{1-\epsilon}}, \tag{12}$$

and, at the feasible pair $(\widehat{B},\widehat{A})$,

$$\|\nabla_{\widehat{A}} f(\widehat{B}\widehat{A})\|_{\mathrm{F}} \le \|R\|_*, \qquad \|\operatorname{grad}_{\widehat{B}} f(\widehat{B}\widehat{A})\|_{\mathrm{F}} \le \sqrt{1+\epsilon}\, A_{\max}\|R\|_*, \tag{13}$$

where $\operatorname{grad}_{\widehat{B}}$ is defined as in Proposition A.2.

Proof. Set $G := B^\top B = I_r + C$. Since $\|C\|_{\mathrm{op}} \le \epsilon < 1$, $(1-\epsilon)I_r \preceq G \preceq (1+\epsilon)I_r$, so G is positive definite. Direct multiplication gives

$$\widehat{B}^\top \widehat{B} = G^{-1/2}GG^{-1/2} = I_r, \qquad \widehat{B}\widehat{A} = BG^{-1/2}G^{1/2}A = BA.$$

Because $G^{1/2}$ and $G^{-1/2}$ are invertible, $\widehat{B}$ and B have the same column space, while $\widehat{A}$ and A have the same row space. Hence $P_{\widehat{B}} = P_B$ and $P_{\widehat{A}} = P_A$, and the product-projector formula in Lemma A.1 gives $\Pi_{\widehat{B},\widehat{A}} = \Pi_{B,A}$. Together with product preservation,

$$\widehat{R} = \Pi_{\widehat{B},\widehat{A}}\nabla f(\widehat{B}\widehat{A}) = \Pi_{B,A}\nabla f(BA) = R.$$

For every eigenvalue $c \in [-\epsilon,\epsilon]$ of C,

$$|\sqrt{1+c}-1| = \frac{|c|}{\sqrt{1+c}+1} \le \frac{|c|}{1+\sqrt{1-\epsilon}}.$$

Applying this inequality spectrally gives $\|G^{1/2}-I_r\|_{\mathrm{F}} \le \|C\|_{\mathrm{F}}/(1+\sqrt{1-\epsilon})$. Since $B = \widehat{B}G^{1/2}$ and $\widehat{B}^\top\widehat{B} = I_r$, $\|B-\widehat{B}\|_{\mathrm{F}} = \|G^{1/2}-I_r\|_{\mathrm{F}}$. Also, $\widehat{A}-A = (G^{1/2}-I_r)A$, so

$$\|\widehat{A}-A\|_{\mathrm{F}} \le \|G^{1/2}-I_r\|_{\mathrm{F}}\|A\|_{\mathrm{op}} \le \frac{A_{\max}\|C\|_{\mathrm{F}}}{1+\sqrt{1-\epsilon}}.$$

Finally, let $F := \nabla f(BA)$. Since $R = \Pi_{B,A}F$, the projector identity gives $F-R = (I-P_B)F(I-P_A)$. Preservation of the factor spaces therefore implies $\widehat{B}^\top(F-R) = 0$ and $(F-R)\widehat{A}^\top = 0$. By the chain rule, $\nabla_{\widehat{A}} f(\widehat{B}\widehat{A}) = \widehat{B}^\top F = \widehat{B}^\top R$, and hence $\|\nabla_{\widehat{A}} f(\widehat{B}\widehat{A})\|_{\mathrm{F}} \le \|R\|_{\mathrm{F}} \le \|R\|_*$. The $\widehat{B}$-component of the Riemannian gradient is the Frobenius-orthogonal projection of $F\widehat{A}^\top = R\widehat{A}^\top$ onto $T_{\widehat{B}}\operatorname{St}(m,r)$. Consequently,

$$\|\operatorname{grad}_{\widehat{B}} f(\widehat{B}\widehat{A})\|_{\mathrm{F}} \le \|R\widehat{A}^\top\|_{\mathrm{F}} \le \|R\|_{\mathrm{F}}\|\widehat{A}\|_{\mathrm{op}} \le \sqrt{1+\epsilon}\, A_{\max}\|R\|_*,$$

where $\|\widehat{A}\|_{\mathrm{op}} = \|G^{1/2}A\|_{\mathrm{op}} \le \sqrt{1+\epsilon}\, A_{\max}$. □

Thus, constraint violation controls the distance to an exactly feasible factorization representing the same product, while the projected gradient R controls first-order stationarity at that feasible representation. This reparameterization is used only as an analytical certificate and is not an additional step of Algorithm 1.

C.5 Why Pointwise Full Row Rank of A_t Is Insufficient

The quantitative lower bound is stronger than pointwise full row rank because both the factor realization and the variation of $P_{A_t} = A_t^\dagger A_t$ depend on $A_t^\dagger$. This mirrors the inverse-smallest-singular-value sensitivity of tangent-space projections in fixed-rank optimization identified by Savostianova et al. (2023). The following counterexample shows that requiring each A_t to have full row rank is insufficient for a uniform step-size bound that preserves a fixed safe region.

Proposition C.4 (Counterexample under pointwise full row rank)**.** *Fix $0 < \epsilon < 1$ and any $\gamma > 0$. There exist factors with $m = n = 2$, $r = 1$, $B \in \mathrm{St}(2,1)$, and full-row-rank A such that one ProLand step with exact gradients and $\beta = 0$ satisfies*

$$\|(B^+)^\top B^+ - I_r\|_{\mathrm{op}} > \epsilon.$$

Hence, without a positive uniform lower bound on $\sigma_r(A)$, pointwise full row rank alone cannot provide a positive safe step size that is uniform over all admissible A.

Proof. Take $B = e_1$, $A = ae_1^\top$ with $a > 0$, and $f(W) = \frac{1}{2}(W_{21}+1)^2$. Then $W = BA = ae_1e_1^\top$, so $F := \nabla f(W) = e_2e_1^\top$. Since $P_B = P_A = e_1e_1^\top$, the product-projector formula gives $\Pi_{B,A}F = F$. With $\beta = 0$, we therefore have $J = F$ and $Y = \mathrm{polar}(J) = F$. Consequently,

$$D_A = B^\dagger Y = 0, \qquad D_B = (I - P_B)YA^\dagger = \frac{1}{a}e_2, \qquad C = B^\top B - I_r = 0.$$

The Landing correction therefore vanishes at this point, and one step gives

$$B^+ = B - \gamma D_B - \rho BC = e_1 - \frac{\gamma}{a}e_2, \qquad C^+ = (B^+)^\top B^+ - 1 = \frac{\gamma^2}{a^2}.$$

For any fixed $\gamma > 0$, choosing $0 < a < \gamma/\sqrt{\epsilon}$ gives $C^+ > \epsilon$. Nevertheless, $A = ae_1^\top$ has full row rank for every $a > 0$, while $\sigma_r(A) = a$ and $\|A^\dagger\|_{\mathrm{op}} = a^{-1}$. Thus the pseudoinverse can become arbitrarily large while full row rank is preserved. Pointwise full row rank therefore does not yield a uniform safe-step bound, which motivates the uniform lower bound $\sigma_r(A_t) \geq a_- > 0$ in Assumption 4.2. □

Observed singular-value bounds for A. Proposition C.4 explains why a quantitative lower bound on $\sigma_r(A)$ enters the safe-step condition. We examine this assumption in the same training runs as Figure 3. Across all recorded post-warmup states, adapters, and runs, we observe $\sigma_r(A) \geq 3.65 \times 10^{-4}$ and $\sigma_1(A) \leq 8.06 \times 10^{-2}$, with a maximum condition number below 4.45. These observations support the A-factor regularity assumption over this training window. Figure 4 shows the full trajectories: the extrema level off with linear decay and continue to grow under the constant rate while remaining within the observed bounds.

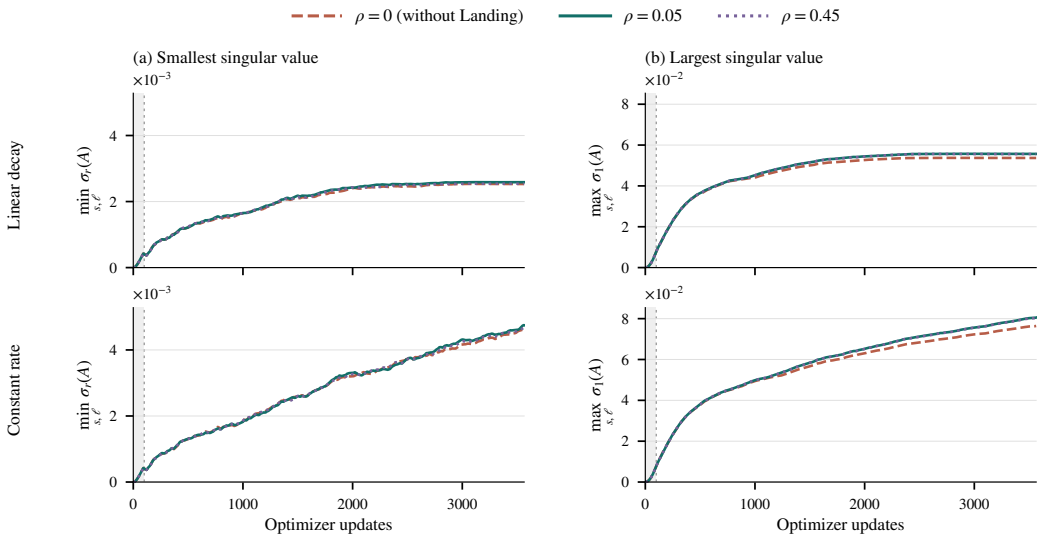

Figure 4: **Extremal singular values of A during fine-tuning.** Curves take extrema over all adapters and runs. The gray region marks warmup; the reported positive lower bound applies to recorded states after warmup. Colors and line styles match Figure 3.

D Product Descent and Polar-Factor Input Tracking

This appendix develops the finite-step descent and input-tracking estimates used in Theorem 4.2. Section D.1 derives the exact product increment and one-step descent bound; Section D.2 controls the variation of the product-space projector and projected gradient; Sections D.3–D.4 analyze the momentum-state tracking error and its effect on the polar-factor input. Throughout, we write $W_t := B_t A_t$, $F_t := \nabla f(W_t)$, $\Delta_t := J_t - R_t$, and $\bar{\gamma} := a_- \sqrt{\epsilon(1-q_C)}$.

D.1 Exact Product Increment and One-Step Descent

Let Algorithm 1 run under Assumptions 4.1–4.2 and the conditions of Proposition 4.1, with a constant optimization step $\gamma_t \equiv \gamma \in (0, \bar{\gamma}]$.

We first record the exact product increment generated by the simultaneous factor updates. From

$$A_{t+1} = A_t - \gamma D_{A,t}, \qquad B_{t+1} = B_t - \gamma D_{B,t} - \rho B_t C_t,$$

together with (3),

$$W_{t+1} - W_t = -\gamma \xi_t + Q_t, \qquad Q_t = \gamma^2 D_{B,t} D_{A,t} - \rho B_t C_t (A_t - \gamma D_{A,t}). \tag{14}$$

Proposition 4.1 and Assumption 4.2, together with the rank bounds of the factor directions, give the bounds on $D_{A,t}$ and $D_{B,t}$ below. Lemma A.1 gives $\mathrm{rank}(J_t) \le 2r$; the rank identity for the polar factor and Frobenius nonexpansiveness of Π_t then give the bound on ξ_t:

$$\|D_{A,t}\|_{\mathrm{op}} \le (1-\epsilon)^{-1/2}, \qquad \|D_{A,t}\|_{\mathrm{F}} \le \sqrt{\frac{r}{1-\epsilon}}, \qquad \|D_{B,t}\|_{\mathrm{F}} \le \frac{\sqrt{r}}{a_-}, \qquad \|\xi_t\|_{\mathrm{F}} \le \sqrt{2r}.$$

Consequently,

$$\|Q_t\|_{\mathrm{F}} \le \frac{\sqrt{r}}{a_- \sqrt{1-\epsilon}} \gamma^2 + \rho \sqrt{1+\epsilon}\, \|C_t\|_{\mathrm{F}} \left(A_{\max} + \frac{\gamma}{\sqrt{1-\epsilon}} \right).$$

Using the Frobenius feasibility bound (9) from Appendix C, we obtain

$$\|Q_t\|_{\mathrm{F}} \le C_Q \gamma^2 + D_Q q_C^t, \tag{15}$$

where

$$C_Q := \frac{\sqrt{r}}{a_-\sqrt{1-\epsilon}} + \frac{\rho\sqrt{r(1+\epsilon)}}{a_-^2(1-q_C)}\left(A_{\max} + \frac{\bar{\gamma}}{\sqrt{1-\epsilon}}\right),$$

$$D_Q := \rho\sqrt{1+\epsilon}\,\|C_0\|_{\mathrm{F}}\left(A_{\max} + \frac{\bar{\gamma}}{\sqrt{1-\epsilon}}\right).$$

Let $S_Q := \sqrt{2r} + \bar{\gamma}C_Q$. Then (14) and (15) imply

$$\|W_{t+1} - W_t\|_{\mathrm{F}} \le S_Q\gamma + D_Q q_C^t. \tag{16}$$

By Proposition C.2 and $\mathrm{rank}(W_t) \le r$,

$$\|W_t\|_{\mathrm{F}} \le \sqrt{r}\,\|W_t\|_{\mathrm{op}} \le \sqrt{r(1+\epsilon)}\,A_{\max}.$$

Assumption 4.1 therefore gives the uniform gradient bound

$$\|F_t\|_{\mathrm{F}} \le G_{\max} := \|F_0\|_{\mathrm{F}} + 2L\sqrt{r(1+\epsilon)}\,A_{\max}. \tag{17}$$

Lemma D.1 (One-step product descent)**.** *For every $0 \le t < T$,*

$$f(W_{t+1}) \le f(W_t) - \gamma\|R_t\|_* + 2\sqrt{2r}\,\gamma\|\Delta_t\|_{\mathrm{F}} + C_D\gamma^2 + D_D q_C^t, \tag{18}$$

where $C_D := G_{\max}C_Q + LS_Q^2$ and $D_D := G_{\max}D_Q + LD_Q^2$.

Proof. Starting from the alignment inequality in Section 4 and using $\Delta_t \in \mathcal{S}_t$, hence $\mathrm{rank}(\Delta_t) \le 2r$,

$$\langle F_t, \xi_t\rangle_{\mathrm{F}} \ge \|R_t\|_* - 2\|\Delta_t\|_* \ge \|R_t\|_* - 2\sqrt{2r}\,\|\Delta_t\|_{\mathrm{F}}.$$

This argument uses only spectral–nuclear duality and does not require any Lipschitz property of the polar-factor map.

Frobenius smoothness along the actual product-update segment gives

$$f(W_{t+1}) \le f(W_t) + \langle F_t, W_{t+1} - W_t\rangle_{\mathrm{F}} + \frac{L}{2}\|W_{t+1} - W_t\|_{\mathrm{F}}^2.$$

Substituting (14) and the alignment bound above yields

$$f(W_{t+1}) \le f(W_t) - \gamma\|R_t\|_* + 2\sqrt{2r}\,\gamma\|\Delta_t\|_{\mathrm{F}} + G_{\max}\|Q_t\|_{\mathrm{F}} + \frac{L}{2}\|W_{t+1} - W_t\|_{\mathrm{F}}^2.$$

Using (15), (16), and

$$(S_Q\gamma + D_Q q_C^t)^2 \le 2S_Q^2\gamma^2 + 2D_Q^2 q_C^t$$

gives (18). □

D.2 Projector and Projected-Gradient Variation

We next control the temporal variation of the reachable product space and the projected gradient.

Lemma D.2 (Orthogonal-projector perturbation)**.** *For matrices X, Y of the same shape, let $P_X = XX^\dagger$ and $P_Y = YY^\dagger$. Then*

$$\|P_X - P_Y\|_{\mathrm{op}} \le \min\{1,\, \max\{\|X^\dagger\|_{\mathrm{op}}, \|Y^\dagger\|_{\mathrm{op}}\}\|X - Y\|_{\mathrm{op}}\}.$$

No equal-rank assumption on X and Y is required.

Proof. Write $P = P_X$ and $Q = P_Y$. Since

$$(P - Q)^2 = P(I - Q)P + (I - P)Q(I - P),$$

with the two terms acting on orthogonal subspaces,

$$\|P - Q\|_{\mathrm{op}} = \max\{\|(I - Q)P\|_{\mathrm{op}}, \|(I - P)Q\|_{\mathrm{op}}\}.$$

Using $(I - Q)Y = 0$ and $(I - P)X = 0$,

$$(I - Q)P = (I - Q)(X - Y)X^\dagger, \qquad (I - P)Q = (I - P)(Y - X)Y^\dagger.$$

This gives the second bound. Since P and Q are orthogonal projectors, $-I \preceq P - Q \preceq I$, and hence $\|P - Q\|_{\mathrm{op}} \le 1$. □

Lemma D.3 (Projector and projected-gradient variation along ProLand)**.** *Let Algorithm 1 run under Assumptions 4.1–4.2 and the conditions of Proposition 4.1, with $\gamma_t \equiv \gamma \in (0, \bar{\gamma}]$. Then, for every $0 \le t < T$,*

$$\|\Pi_{t+1} - \Pi_t\|_{\mathrm{F}\to\mathrm{F}} \le C_\Pi \gamma + D_\Pi q_C^t, \qquad \|R_{t+1} - R_t\|_{\mathrm{F}} \le C_R \gamma + D_R q_C^t,$$

where

$$C_\Pi := \frac{2}{a_-\sqrt{1-\epsilon}} + \frac{\rho\sqrt{1+\epsilon}\,\bar{\gamma}}{\sqrt{1-\epsilon}\,a_-^2(1-q_C)}, \quad D_\Pi := \rho\sqrt{\frac{1+\epsilon}{1-\epsilon}}\,\|C_0\|_{\mathrm{op}},$$
$$C_R := LS_Q + G_{\max} C_\Pi, \qquad D_R := LD_Q + G_{\max} D_\Pi.$$

Proof. For any matrix Z, direct subtraction of the product-space projectors gives

$$(\Pi_{t+1} - \Pi_t)Z = (P_{B_{t+1}} - P_{B_t})Z(I - P_{A_t}) + (I - P_{B_{t+1}})Z(P_{A_{t+1}} - P_{A_t}).$$

Lemma D.2, Proposition 4.1, and Assumption 4.2 therefore imply

$$\|\Pi_{t+1} - \Pi_t\|_{\mathrm{F}\to\mathrm{F}} \le \frac{\|B_{t+1} - B_t\|_{\mathrm{op}}}{\sqrt{1-\epsilon}} + \frac{\|A_{t+1} - A_t\|_{\mathrm{op}}}{a_-}.$$

For the A-factor, Proposition 4.1 gives

$$\|A_{t+1} - A_t\|_{\mathrm{op}} = \gamma\|D_{A,t}\|_{\mathrm{op}} \le \frac{\gamma}{\sqrt{1-\epsilon}}.$$

For the B-factor, $B_t^\top D_{B,t} = 0$ eliminates the cross terms exactly:

$$(B_{t+1} - B_t)^\top (B_{t+1} - B_t) = \gamma^2 D_{B,t}^\top D_{B,t} + \rho^2 C_t(I_r + C_t)C_t.$$

Using Proposition 4.1 and $\|D_{B,t}\|_{\mathrm{op}} \le a_-^{-1}$,

$$\|B_{t+1} - B_t\|_{\mathrm{op}} \le \frac{\gamma}{a_-} + \rho\sqrt{1+\epsilon}\,\|C_t\|_{\mathrm{op}}.$$

Hence

$$\|\Pi_{t+1} - \Pi_t\|_{\mathrm{F}\to\mathrm{F}} \le \frac{2\gamma}{a_-\sqrt{1-\epsilon}} + \rho\sqrt{\frac{1+\epsilon}{1-\epsilon}}\,\|C_t\|_{\mathrm{op}}.$$

The constant-step feasibility bound (4) and $\gamma^2 \le \bar{\gamma}\gamma$ yield

$$\|\Pi_{t+1} - \Pi_t\|_{\mathrm{F}\to\mathrm{F}} \le C_\Pi \gamma + D_\Pi q_C^t.$$

Finally,

$$R_{t+1} - R_t = \Pi_{t+1}(F_{t+1} - F_t) + (\Pi_{t+1} - \Pi_t)F_t.$$

Using Frobenius nonexpansiveness of Π_{t+1}, Assumption 4.1, and the D.1 bounds (16) and (17),

$$\begin{aligned}\|R_{t+1} - R_t\|_{\mathrm{F}} &\le L\|W_{t+1} - W_t\|_{\mathrm{F}} + G_{\max}\|\Pi_{t+1} - \Pi_t\|_{\mathrm{F}\to\mathrm{F}} \\ &\le (LS_Q + G_{\max} C_\Pi)\gamma + (LD_Q + G_{\max} D_\Pi) q_C^t = C_R\gamma + D_R q_C^t.\end{aligned}$$

□

D.3 Momentum-State Tracking and Noise Control

For the stochastic analysis, use the filtration $\{\mathcal{F}_t\}$ in Assumption 4.1, generated by the algorithmic history up to iteration t, before sampling G_t, and define $\zeta_t := G_t - F_t$. Then $B_t, A_t, W_t, \Pi_t, H_{t-1}, F_t, R_t$ are $\mathcal{F}_t$-measurable, while $\mathbb{E}[\zeta_t \mid \mathcal{F}_t] = 0$ and $\mathbb{E}[\|\zeta_t\|_{\mathrm{F}}^2 \mid \mathcal{F}_t] \le \sigma^2$. Throughout Sections D.3–D.4, assume $\mathbb{E}[\|H_0 - R_0\|_{\mathrm{F}}^2 \mid \mathcal{F}_0] \le U_0 < \infty$ almost surely, for a deterministic constant U_0 independent of T. Write $\mathbb{E}_0[\cdot] := \mathbb{E}[\cdot \mid \mathcal{F}_0]$, and define the retained-state tracking error $E_t := H_t - R_t$. Since $\Pi_t G_t = R_t + \Pi_t \zeta_t$, the polar-factor input mismatch satisfies

$$\Delta_t = \beta E_t + (1-\beta)\Pi_t \zeta_t. \tag{19}$$

Lemma D.4 (Tracking-error and input-noise decomposition)**.** *Define*

$$V_0 := \beta(\Pi_0 \bar{H}_{-1} - R_0), \qquad N_{-1} := 0, \qquad N_0 := (1-\beta)\Pi_0 \zeta_0,$$

and, for $t \geq 1$ and $t \geq 0$, respectively,

$$V_t = \beta \Pi_t (V_{t-1} + R_{t-1} - R_t), \qquad N_t = \beta \Pi_t N_{t-1} + (1-\beta)\Pi_t \zeta_t.$$

Then, for every $t \geq 0$,

$$E_t = V_t + N_t, \qquad \|V_0\|_{\mathrm{F}}^2 \leq U_0.$$

Moreover, for every $k \geq 0$,

$$\mathbb{E}_0 \|N_k\|_{\mathrm{F}}^2 \leq \frac{1-\beta}{1+\beta}(1-\beta^{2k+2})\sigma^2.$$

Define further $L_t := \beta^2 \Pi_t N_{t-1} + (1-\beta^2)\Pi_t \zeta_t$. Then

$$\Delta_t = \beta V_t + L_t, \qquad \mathbb{E}_0 \|L_t\|_{\mathrm{F}} \leq \sigma \chi_N(\beta),$$

where

$$\chi_N(\beta)^2 := \beta^4 \frac{1-\beta}{1+\beta} + (1-\beta^2)^2 = \frac{1-\beta}{1+\beta}(1 + 2\beta - 2\beta^3). \tag{20}$$

Proof. The retained-state update gives, for $t \geq 1$,

$$E_t = \beta \Pi_t (E_{t-1} + R_{t-1} - R_t) + (1-\beta)\Pi_t \zeta_t.$$

At $t = 0$,

$$E_0 = \beta(\Pi_0 \bar{H}_{-1} - R_0) + (1-\beta)\Pi_0 \zeta_0 = V_0 + N_0.$$

The definitions of V_t and N_t therefore imply $E_t = V_t + N_t$ by induction. Since V_0 and Π_0 are $\mathcal{F}_0$-measurable and $\mathbb{E}[\zeta_0 \mid \mathcal{F}_0] = 0$,

$$\mathbb{E}_0 \|E_0\|_{\mathrm{F}}^2 = \|V_0\|_{\mathrm{F}}^2 + \mathbb{E}_0 \|N_0\|_{\mathrm{F}}^2.$$

The initial-state bound thus gives $\|V_0\|_{\mathrm{F}}^2 \leq U_0$.

For the noise component, the $\mathcal{F}_t$-measurability of N_{t-1} and Π_t gives

$$\mathbb{E}[\langle \Pi_t N_{t-1}, \Pi_t \zeta_t \rangle_{\mathrm{F}} | \mathcal{F}_t] = 0.$$

Hence

$$\mathbb{E}[\|N_t\|_{\mathrm{F}}^2 \mid \mathcal{F}_t] \leq \beta^2 \|N_{t-1}\|_{\mathrm{F}}^2 + (1-\beta)^2 \sigma^2.$$

Taking $\mathbb{E}_0$ and iterating yields

$$\mathbb{E}_0 \|N_k\|_{\mathrm{F}}^2 \leq (1-\beta)^2 \sigma^2 \sum_{j=0}^{k} \beta^{2j} = \frac{1-\beta}{1+\beta}(1-\beta^{2k+2})\sigma^2.$$

Finally, using (19), $E_t = V_t + N_t$, and the recurrence for N_t,

$$\Delta_t = \beta V_t + \beta N_t + (1-\beta)\Pi_t \zeta_t = \beta V_t + L_t.$$

The same conditional-centering argument gives

$$\mathbb{E}_0 \|L_t\|_{\mathrm{F}}^2 \leq \beta^4 \mathbb{E}_0 \|N_{t-1}\|_{\mathrm{F}}^2 + (1-\beta^2)^2 \sigma^2 \leq \sigma^2 \chi_N(\beta)^2.$$

Jensen's inequality then gives $\mathbb{E}_0 \|L_t\|_{\mathrm{F}} \leq \sigma \chi_N(\beta)$. □

D.4 Averaged Tracking of the Polar-Factor Input

Lemma D.5 (Averaged tracking of the polar-factor input)**.** *Let Algorithm 1 run for T iterations under Assumptions 4.1–4.2, the initial-state bound in Section D.3, and the conditions of Proposition 4.1, with fixed $\beta \in [0,1)$ and $\gamma_t \equiv \gamma \in (0, \bar{\gamma}]$. Let C_R, D_R be the constants from Lemma D.3, and define*

$$M_0 := \sqrt{U_0} + \frac{D_R}{1-q_C}.$$

Then, almost surely,

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}_0\|\Delta_t\|_{\mathrm{F}} \le \frac{\beta M_0}{T(1-\beta)} + \frac{\beta^2 C_R\gamma}{1-\beta} + \sigma\chi_N(\beta), \tag{21}$$

where $\chi_N(\beta)$ is defined in Lemma D.4. The same bound holds for total expectations after replacing the initial-state-dependent quantities by common deterministic upper bounds.

Proof. Lemma D.3 gives, for $t \ge 1$,

$$\|R_t - R_{t-1}\|_{\mathrm{F}} \le C_R\gamma + D_R q_C^{t-1}.$$

The recurrence for V_t in Lemma D.4 and Frobenius nonexpansiveness of Π_t imply

$$\|V_t\|_{\mathrm{F}} \le \beta\|V_{t-1}\|_{\mathrm{F}} + \beta(C_R\gamma + D_R q_C^{t-1}).$$

Iterating gives

$$\|V_t\|_{\mathrm{F}} \le \beta^t\|V_0\|_{\mathrm{F}} + \beta\sum_{j=1}^{t}\beta^{t-j}(C_R\gamma + D_R q_C^{j-1}).$$

The finite geometric sums satisfy

$$\sum_{t=0}^{T-1}\sum_{j=1}^{t}\beta^{t-j} \le \frac{T-1}{1-\beta}, \qquad \sum_{t=0}^{T-1}\sum_{j=1}^{t}\beta^{t-j}q_C^{j-1} \le \frac{1}{(1-\beta)(1-q_C)}.$$

Hence

$$\frac{1}{T}\sum_{t<T}\beta\|V_t\|_{\mathrm{F}} \le \frac{\beta(1-\beta^T)\|V_0\|_{\mathrm{F}}}{T(1-\beta)} + \frac{\beta^2 C_R\gamma}{1-\beta} + \frac{\beta^2 D_R}{T(1-\beta)(1-q_C)}.$$

Using $\|V_0\|_{\mathrm{F}} \le \sqrt{U_0}$ from Lemma D.4, together with $1-\beta^T \le 1$ and $\beta^2 \le \beta$, yields

$$\frac{1}{T}\sum_{t<T}\beta\mathbb{E}_0\|V_t\|_{\mathrm{F}} \le \frac{\beta M_0}{T(1-\beta)} + \frac{\beta^2 C_R\gamma}{1-\beta}.$$

Finally, Lemma D.4 gives $\Delta_t = \beta V_t + L_t$ and $\mathbb{E}_0\|L_t\|_{\mathrm{F}} \le \sigma\chi_N(\beta)$. Therefore,

$$\begin{aligned}\frac{1}{T}\sum_{t<T}\mathbb{E}_0\|\Delta_t\|_{\mathrm{F}} &\le \frac{1}{T}\sum_{t<T}\beta\mathbb{E}_0\|V_t\|_{\mathrm{F}} + \frac{1}{T}\sum_{t<T}\mathbb{E}_0\|L_t\|_{\mathrm{F}} \\ &\le \frac{\beta M_0}{T(1-\beta)} + \frac{\beta^2 C_R\gamma}{1-\beta} + \sigma\chi_N(\beta).\end{aligned}$$

□

E Proofs of the Convergence Rates

This appendix proves the deterministic and stochastic convergence rates in Theorem 4.2. We first combine the one-step product descent bound in Lemma D.1 with the averaged polar-factor input tracking bound in Lemma D.5 to control the averaged product-space stationarity measure $\|R_t\|_*$, and use Proposition 4.1 together with (9) to control the constraint violation. We then derive the guarantees for the same randomized output from these averaged-stationarity and tail-feasibility bounds.

All iteration indices refer to the regular post-initialization segment described in Section 4. The initial state includes the inherited momentum $\bar{H}_{-1}$; reindexing does not restart the algorithm. The analysis assumes regularity throughout this segment and does not establish entry into it from zero initialization. The initialization phase and the feasibility cutoff $\tau_T(b)$ are distinct. For unconditional rates, initial-state-dependent quantities in the bounds below are controlled by common deterministic constants independent of T.

E.1 FINITE-HORIZON BOUNDS

Let $D_0 := f(W_0) - f_* + D_D/(1-q_C)$.

Lemma E.1 (Conditional finite-horizon stationarity bound)**.** *Let $\{(B_t, A_t)\}_{t=0}^{T}$ be generated by Algorithm 1 under Assumptions 4.1–4.2, the initial-state bound in Section D.3, and the conditions of Proposition 4.1. Use fixed $\beta \in [0,1)$ and $\gamma_t \equiv \gamma \in (0, \bar{\gamma}]$. With $\mathbb{E}_0[\cdot] = \mathbb{E}[\cdot \mid \mathcal{F}_0]$, almost surely,*

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}_0\|R_t\|_* \le \frac{D_0}{T\gamma} + C_D\gamma + 2\sqrt{2r}\left[\frac{\beta M_0}{T(1-\beta)} + \frac{\beta^2 C_R \gamma}{1-\beta} + \sigma\chi_N(\beta)\right].$$

Proof. Lemma D.1 gives

$$f(W_{t+1}) \le f(W_t) - \gamma\|R_t\|_* + 2\sqrt{2r}\,\gamma\|\Delta_t\|_{\mathrm{F}} + C_D\gamma^2 + D_D q_C^t.$$

Summing over $t = 0, \dots, T-1$ and using $f(W_T) \ge f_*$ and $\sum_{t=0}^{T-1} q_C^t \le (1-q_C)^{-1}$ gives

$$\gamma\sum_{t=0}^{T-1}\|R_t\|_* \le D_0 + TC_D\gamma^2 + 2\sqrt{2r}\,\gamma\sum_{t=0}^{T-1}\|\Delta_t\|_{\mathrm{F}}.$$

Dividing by $T\gamma$, taking $\mathbb{E}[\cdot \mid \mathcal{F}_0]$, and invoking Lemma D.5 yield the claim. Taking total expectations gives the corresponding unconditional estimate. □

Lemma E.2 (Uniform tail-feasibility bound)**.** *Suppose the conditions of Proposition 4.1 hold for Algorithm 1 with*

$$\gamma_t \equiv \gamma_T = c_\gamma T^{-b}, \qquad c_\gamma > 0, \quad b > 0,$$

where c_γ and b are fixed. Define

$$\tau_T(b) = \left\lceil \frac{2b\log T}{|\log q_C|} \right\rceil.$$

For all sufficiently large T,

$$\sup_{\tau_T(b)\le t\le T}\|C_t\|_{\mathrm{op}} \le \left(\|C_0\|_{\mathrm{op}} + \frac{c_\gamma^2}{a_-^2(1-q_C)}\right)T^{-2b},$$

$$\sup_{\tau_T(b)\le t\le T}\|C_t\|_{\mathrm{F}} \le \left(\|C_0\|_{\mathrm{F}} + \frac{\sqrt{r}\,c_\gamma^2}{a_-^2(1-q_C)}\right)T^{-2b}.$$

If $C_0 = 0$, then

$$\sup_{0\le t\le T}\|C_t\|_{\mathrm{op}} \le \frac{c_\gamma^2}{a_-^2(1-q_C)}T^{-2b},$$

$$\sup_{0\le t\le T}\|C_t\|_{\mathrm{F}} \le \frac{\sqrt{r}\,c_\gamma^2}{a_-^2(1-q_C)}T^{-2b}.$$

These bounds hold pathwise and do not require the objective-regularity or stochastic-oracle assumptions.

Proof. The operator-norm bound (4) and its Frobenius counterpart (9) give

$$\|C_t\|_{\mathrm{op}} \le q_C^t\|C_0\|_{\mathrm{op}} + \frac{c_\gamma^2 T^{-2b}(1-q_C^t)}{a_-^2(1-q_C)},$$

$$\|C_t\|_{\mathrm{F}} \le q_C^t\|C_0\|_{\mathrm{F}} + \frac{\sqrt{r}\,c_\gamma^2 T^{-2b}(1-q_C^t)}{a_-^2(1-q_C)}.$$

For $t \ge \tau_T(b)$,

$$q_C^t \le q_C^{\tau_T(b)} \le T^{-2b}.$$

Substitution and taking the supremum prove the tail bounds. If $C_0 = 0$, the initial-violation terms vanish, giving the full-run bounds. □

E.2 Deterministic and Stochastic Rates

Proof of Theorem 4.2(i). For exact gradients and $\beta = 0$,

$$J_t = \bar{H}_t = R_t, \qquad \Delta_t = E_t = 0.$$

Since $\gamma_T = \theta T^{-1/2} \to 0$, we have $\gamma_T \le \bar{\gamma}$ for all sufficiently large T, so Lemma D.1 applies. Telescoping the bound in Lemma D.1 gives

$$\frac{1}{T}\sum_{t=0}^{T-1} \|R_t\|_* \le \frac{D_0}{T\gamma_T} + C_D\gamma_T = \left(\frac{D_0}{\theta} + C_D\theta\right) T^{-1/2}.$$

Apply Lemma E.2 with $b = 1/2$ and $c_\gamma = \theta$:

$$\sup_{\tau_T(1/2)\le t\le T} \|C_t\|_{\mathrm{op}} \le \left(\|C_0\|_{\mathrm{op}} + \frac{\theta^2}{a_-^2(1-q_C)}\right) T^{-1},$$

where

$$\tau_T(1/2) = \left\lceil \frac{\log T}{|\log q_C|} \right\rceil = O(\log T).$$

If $C_0 = 0$, Lemma E.2 gives the same $O(T^{-1})$ bound over the full run $0 \le t \le T$. The rates for the same randomized output follow from the argument at the end of this subsection. □

We next consider stochastic gradients.

Power-law balance. To explain the parameter choices in Theorem 4.2(ii), let

$$1 - \beta_T = c_\beta T^{-a}, \qquad \gamma_T = c_\gamma T^{-b}, \qquad 0 < a < b < 1,$$

with fixed $c_\beta, c_\gamma > 0$. For $0 \le \beta < 1$, (20) gives

$$\chi_N(\beta)^2 = \frac{1-\beta}{1+\beta}(1 + 2\beta - 2\beta^3) \le 2(1-\beta).$$

Thus

$$\chi_N(\beta_T) \le \sqrt{2c_\beta}\, T^{-a/2}.$$

The finite-horizon term, product/Landing remainder, tracking transient, momentum drift, and stochastic noise in Lemma E.1 scale as

$$\frac{D_0}{T\gamma_T} = O(T^{b-1}), \qquad C_D\gamma_T = O(T^{-b}), \qquad \frac{\beta_T M_0}{T(1-\beta_T)} = O(T^{a-1}),$$
$$\frac{\beta_T^2 C_R \gamma_T}{1-\beta_T} = O(T^{a-b}), \qquad \sigma\chi_N(\beta_T) = O(T^{-a/2}).$$

Requiring all five terms to be $O(T^{-p})$ gives

$$1 - b \ge p, \qquad b \ge p, \qquad 1 - a \ge p, \qquad b - a \ge p, \qquad \frac{a}{2} \ge p.$$

From $a/2 \ge p$ we obtain $a \ge 2p$, and $b - a \ge p$ gives $b \ge a + p \ge 3p$. Meanwhile, $1 - b \ge p$ gives $b \le 1 - p$. Therefore,

$$3p \le 1 - p, \qquad p \le \frac{1}{4}.$$

When $p = 1/4$,

$$a \ge \frac{1}{2}, \qquad b \ge a + \frac{1}{4} \ge \frac{3}{4}, \qquad b \le \frac{3}{4},$$

so uniquely

$$(a, b) = \left(\frac{1}{2}, \frac{3}{4}\right).$$

When all five terms are present, the largest decay exponent certified by this bound is $1/4$, uniquely attained at $a = 1/2$ and $b = 3/4$.

Proof of Theorem 4.2(ii). Set $a = 1/2$ and $b = 3/4$, that is,

$$1 - \beta_T = c_\beta T^{-1/2}, \qquad \gamma_T = c_\gamma T^{-3/4}.$$

For all sufficiently large T, $\beta_T \in [0, 1)$ and $\gamma_T \le \bar{\gamma}$, so Lemmas E.1 and E.2 apply. The five terms have orders

$$T^{-1/4}, \qquad T^{-3/4}, \qquad T^{-1/2}, \qquad T^{-1/4}, \qquad T^{-1/4}.$$

Taking total expectations in Lemma E.1 gives

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\|R_t\|_* = O(T^{-1/4}).$$

Lemma E.2 with $b = 3/4$ gives, pathwise,

$$\sup_{\tau_T(3/4)\le t\le T} \|C_t\|_{\mathrm{op}} \le \left(\|C_0\|_{\mathrm{op}} + \frac{c_\gamma^2}{a_-^2(1 - q_C)} \right) T^{-3/2},$$

where

$$\tau_T(3/4) = \left\lceil \frac{3 \log T}{2|\log q_C|} \right\rceil = O(\log T).$$

If $C_0 = 0$, Lemma E.2 gives the same $O(T^{-3/2})$ bound over the full run $0 \le t \le T$. The rates for the same randomized output follow from the argument at the end of this subsection. □

Common randomized outputs. For either regime, let $\widehat{t}$ have the distribution specified in Theorem 4.2, with $b = 1/2$ or $b = 3/4$, respectively. Since $\tau_T(b) = O(\log T)$, the sampling interval is nonempty for all sufficiently large T, and $T/(T - \tau_T(b)) = 1 + o(1)$. By independence of the output index and nonnegativity,

$$\mathbb{E}\|R_{\widehat{t}}\|_* = \frac{1}{T - \tau_T(b)} \sum_{t=\tau_T(b)}^{T-1} \mathbb{E}\|R_t\|_* \le \frac{T}{T - \tau_T(b)} \left(\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\|R_t\|_* \right).$$

Thus restriction to this interval preserves the stationarity rate. Moreover,

$$\mathbb{E}\|C_{\widehat{t}}\|_{\mathrm{op}} \le \mathbb{E}\left[\sup_{\tau_T(b)\le t\le T} \|C_t\|_{\mathrm{op}} \right].$$

Substituting the corresponding averaged-stationarity and tail-feasibility bounds proves both parts of Theorem 4.2. In the deterministic case, the expectations on the left are only over $\widehat{t}$, while those over individual iterates on the right are unnecessary.

F Additional Related Work

LoRA learns a weight increment $\Delta W = BA$ while keeping pretrained weights fixed (Hu et al., 2022). Representative extensions modify rank allocation through AdaLoRA (Zhang et al., 2023), use different learning rates for the two factors through LoRA+ (Hayou et al., 2024b), or separate weight magnitude and direction through DoRA (Liu et al., 2024). These developments address different aspects of low-rank adaptation. Expanding the discussion in Section 1.1, we focus on methods that explicitly design the matrix update induced by the factors, impose geometric constraints on the adapter, or change how feasibility is maintained.

The product-space viewpoint concerns the represented matrix BA, whose first-order variation is $D_B A + B D_A$. LoRA-Pro selects factor gradients by matching this induced matrix direction to the full fine-tuning gradient in the Frobenius norm (Wang et al., 2025). LORO derives joint factor updates from Riemannian descent of the represented matrix for low-rank pretraining, combining periodic exact Riemannian updates with cheaper approximate factor updates (Mo et al., 2025). Spectral methods extend this viewpoint using Muon-style polar-factor directions (Jordan et al., 2024). In particular, sMuon reconstructs a low-rank product-space gradient from factor gradients and derives factor directions by linearizing a relaxed Muon objective and solving the resulting least-squares

problem (Anson et al., 2026). LoRA-TSD instead develops tangent-space spectral descent with a factor-induced realization (Andriianov et al., 2026). ProLand additionally constrains B to the Stiefel manifold and realizes the projected product direction through coordinated factor updates. For the input J_t in Algorithm 1, its selected product direction is $\xi_t = \Pi_t \operatorname{polar}(J_t)$, the Frobenius orthogonal projection of the polar factor onto the current space $\mathcal{S}_t$ of realizable first-order product variations. The coordinated factor directions realize this projected direction, rather than the unprojected polar factor; see Section 3 and Appendix A.3.

The constraint may be imposed on the represented matrix or on its factors. Riemannion and LoRA-TSD optimize the represented update on the fixed-rank manifold $\mathcal{M}_r := \{W \in \mathbb{R}^{m\times n} : \operatorname{rank}(W) = r\}$ (Bogachev et al., 2026; Andriianov et al., 2026). Skewon addresses a different object: it solves a spectral-norm linear minimization problem over $T_X \operatorname{St}$ for a supplied matrix direction, rather than constructing coordinated updates of a LoRA product (Solonko et al., 2026). Among factor-constrained methods, Stiefel-LoRA uses $B \in \operatorname{St}(m, r)$ with A unconstrained (Park et al., 2025), and Manifold-LoRA additionally considers an oblique constraint, which normalizes the columns of B without requiring mutual orthogonality (Zhang et al., 2026). Two-sided formulations use three factors: StelLA writes $USV^\top$ and PoLAR writes $X\Theta Y^\top$, with two Stiefel basis factors and an unconstrained $r \times r$ core (Li et al., 2025; Lion et al., 2025). ProLand retains the two-factor representation BA and constrains only B. As shown in Section 2, this formulation represents the full rank-at-most-r class. This representational statement is separate from the convergence assumptions: under the regularity and safe-region conditions of Section 4, the analyzed post-initialization products have rank r. The distinction from direct fixed-rank optimization is therefore not simply an admissible-rank comparison, but the use of coordinated factor directions satisfying $B^\top D_B = 0$ while realizing a selected first-order product variation.

Feasibility treatment further distinguishes these approaches. Stiefel-LoRA uses a QR retraction on B, whereas StelLA uses polar retractions on its two basis factors (Park et al., 2025; Li et al., 2025). Riemannion uses a truncated-SVD retraction of the represented matrix, and LoRA-TSD uses a factor-induced retraction on $\mathcal{M}_r$ (Bogachev et al., 2026; Andriianov et al., 2026); these are rank constraints on the product, not orthogonality constraints on either LoRA factor. Skewon also uses a retraction to maintain feasibility of its Stiefel variable (Solonko et al., 2026). In contrast, Landing methods allow intermediate infeasible iterates and apply an explicit correction to recover feasibility (Ablin & Peyré, 2022; Ablin et al., 2024). Manifold-LoRA applies this mechanism to projected-gradient factor updates and establishes convergence for its SGD-type variant (Zhang et al., 2026); PoLAR uses Landing fields on both basis factors in its practical fine-tuning algorithm (Lion et al., 2025). ProLand retains a separate Landing correction on B, while exploiting the identity $B_t^\top D_{B,t} = 0$ even at infeasible iterates. This identity eliminates the optimization-induced term linear in γ_t and the mixed optimization–Landing term in the exact Gram recursion in Section 4. The optimization step therefore contributes only a quadratic perturbation to that recurrence. Appendix C.1 gives the expansion and explains why the usual skew-symmetry condition on $B^\top D_B$ does not by itself eliminate the mixed term away from feasibility.

Momentum handling and low-rank computation are separate from the choice of feasibility mechanism. Riemannion uses projection transport to move tangent-vector momentum between successive tangent spaces (Bogachev et al., 2026). Skewon's algorithms accept a supplied direction M_t, which may be a gradient or momentum, but do not prescribe a recursion for projecting or transporting a stored momentum state between those spaces (Solonko et al., 2026). ProLand explicitly reprojects its previous state before combining it with the current projected gradient, so both the updated momentum $\bar{H}_t$ and the polar-factor input J_t lie in $\mathcal{S}_t$. This gives rank bounds of at most $2r$ for both matrices. Reduced spectral computations already appear in sMuon and Skewon (Anson et al., 2026; Solonko et al., 2026); ProLand's implementation contribution is the exact low-rank realization of its complete projected-momentum and Landing iteration. Given an initial momentum factorization of width at most $2r$, Appendix B proves equivalence to Algorithm 1 in exact arithmetic, using factor gradients from the same sampled loss and a polar-factor core with at most $2r$ rows and columns, without forming dense product-sized gradients or momentum states. Thus, exactness concerns the implementation of the stated projected iteration. Table 6 summarizes the direction, constraint, and feasibility distinctions discussed above.

Table 6: Structural comparison of related low-rank and Stiefel optimization methods. A checkmark indicates an explicit component of the reviewed method, not a performance advantage.

Method	Manifold constraint	Product direction	Muon-type	Feasibility handling
LoRA-Pro (Wang et al., 2025)	$\times$	✓	$\times$	—
LORO (Mo et al., 2025)	✓ $W \in \mathcal{M}_r$	✓	$\times$	Periodic rank-r retraction
sMuon (Anson et al., 2026)	$\times$	✓	✓	—
LoRA-TSD (Andriianov et al., 2026)	✓ $W \in \mathcal{M}_r$	✓	✓	Factor-induced rank-r retraction
Riemannion (Bogachev et al., 2026)	✓ $W \in \mathcal{M}_r$	✓	✓	Truncated-SVD retraction
Skewon (Solonko et al., 2026)	✓ $X \in \mathrm{St}$	$\times$	✓	Stiefel retraction
Stiefel-LoRA (Park et al., 2025)	✓ $B \in \mathrm{St}$	$\times$	$\times$	QR retraction on B
StelLA (Li et al., 2025)	✓ $U, V \in \mathrm{St}$	$\times$	$\times$	Polar retractions on U, V
PoLAR (Lion et al., 2025)	✓ $X, Y \in \mathrm{St}$	$\times$	$\times$	Landing on X, Y
Manifold-LoRA (Zhang et al., 2026)	✓ $B \in \mathrm{St}$ or $B \in \mathrm{Ob}$	$\times$	$\times$	Landing on B
ProLand (this work)	✓ $B \in \mathrm{St}$	✓	✓	Landing on B

✓/$\times$: present/absent in the reviewed update. "Product direction" means explicit selection or matching of the represented matrix update, not merely dependence of the objective on the factors through their product. "Muon-type" refers to spectral update design, not a polar parameterization or polar retraction. $\mathcal{M}_r$ denotes rank-r matrices; Ob denotes matrices with unit-norm columns. The StelLA entry describes its gradient-based implementation, and the PoLAR entry describes its practical fine-tuning algorithm. A dash means no separate manifold-restoration step; the low-rank parameterization still imposes a rank-at-most-r bound. Landing permits intermediate infeasibility.

G SUPPLEMENTARY LANGUAGE EXPERIMENTS

G.1 RANK-8 LOSS TRAJECTORIES

Figure 5 compares training and validation loss across methods and language tasks on Qwen2.5-3B. The comparison protocol is specified in Appendix L.2.

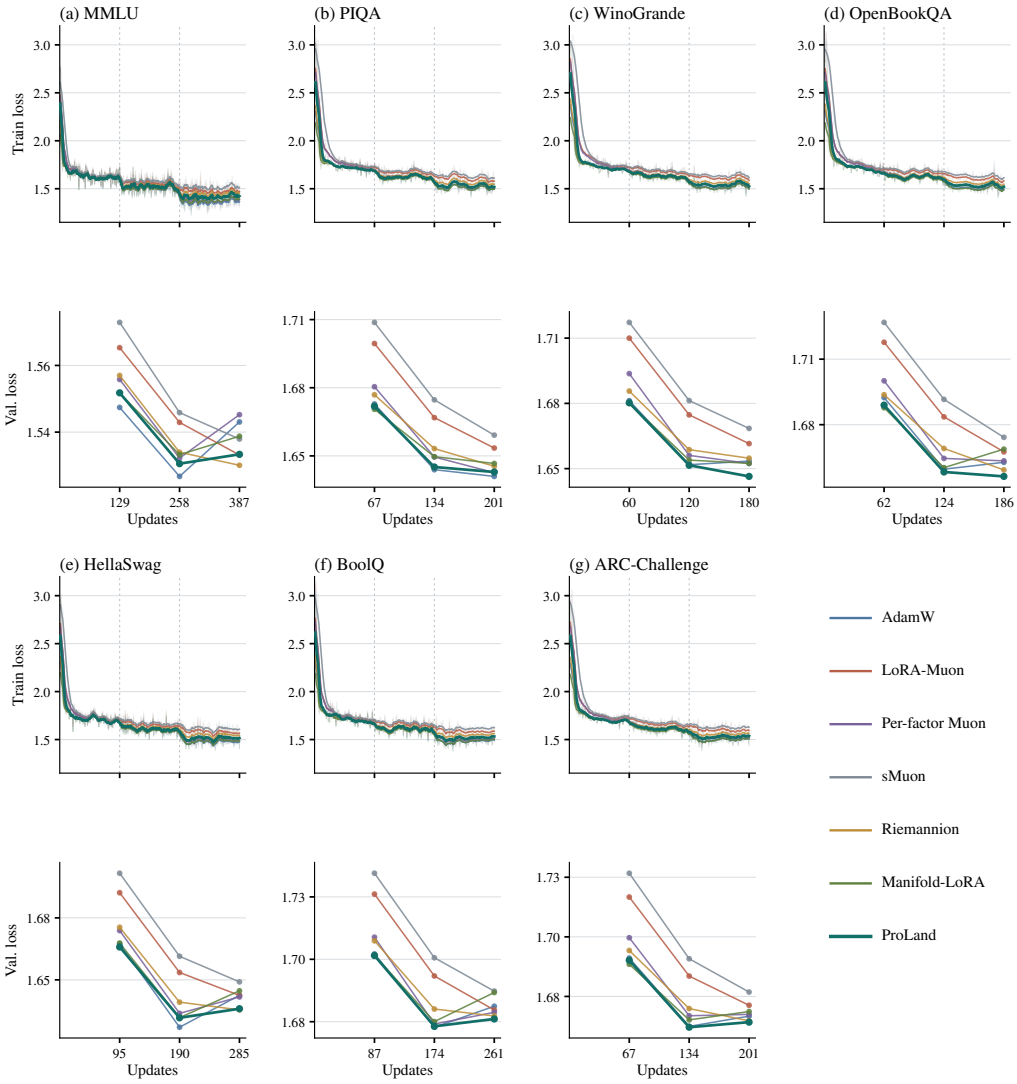

Figure 5: **Qwen2.5-3B rank-8 loss traces.** Each task pairs training loss (upper) with validation loss (lower). Training curves show smoothed and faint raw traces; lines between validation points serve as visual guides.

G.2 PRETRAINING AND FINE-TUNING OPTIMIZER COMPATIBILITY

Liu et al. (2025, Section 3.5, Table 6) cross Muon and AdamW between pretraining and full-parameter SFT using Moonlight checkpoints trained for 1.2T tokens. On their four reported benchmarks, Muon-pretrained checkpoints fine-tuned with Muon outperform the same checkpoints fine-tuned with AdamW.

For low-rank adaptation, Anson et al. (2026, Section 4) explicitly hypothesize that Muon-based LoRA may perform better on Muon-pretrained Moonlight and report partial support across eleven tasks. They also identify Qwen2.5-3B and Llama-3.2-3B as AdamW-pretrained, which is the source of our backbone classification. Qu et al. (2026) study optimizer mismatch with controlled pretraining and fine-tuning experiments and find that LoRA can mitigate it by constraining updates.

These findings motivate our choice of models pretrained with different optimizers. The Moonlight comparison in Table 4 uses the same pretrained checkpoint for all adapter methods. On this shared checkpoint, ProLand also has a higher mean score than Per-factor Muon and LoRA-Muon, both of which use Muon-style adapter updates.

H A TWO-FACTOR MOTIVATION EXAMPLE

Two minima of the same rank-one problem. Consider fitting $\mathrm{diag}(1.5, 1)$ with a rank-one product $W = BA$, assigning weight four to off-diagonal errors:

$$f(BA) = \tfrac{1}{2}\big[(w_{11} - 1.5)^2 + (w_{22} - 1)^2 + 4w_{12}^2 + 4w_{21}^2\big], \qquad W = BA. \tag{22}$$

To verify its global minimum, write $x = w_{11}$ and $y = w_{22}$. Rank one implies $w_{12}w_{21} = xy$, hence

$$f(W) \geq q(x, y) := \tfrac{1}{2}\big[(x - 1.5)^2 + (y - 1)^2\big] + 4|xy|.$$

Replacing x, y by their absolute values cannot increase q. In the positive quadrant its Hessian is $\left(\begin{smallmatrix} 1 & 4 \\ 4 & 1 \end{smallmatrix}\right)$, with a negative eigenvalue; no interior point is a local minimum. Since q is coercive, its minimum lies on an axis. The two axis minima are $(1.5, 0)$ and $(0, 1)$, with values 0.5 and 1.125. The matrix $W_{\mathrm{good}} = \mathrm{diag}(1.5, 0)$ attains the lower bound and is globally optimal. The other matrix, $W_{\mathrm{bad}} = \mathrm{diag}(0, 1)$, is a strict local minimum under the rank-one constraint. In local coordinates $B = (\delta, \sqrt{1 - \delta^2})^\top$ and $A = (a, 1 + b)$, expansion around $(\delta, a, b) = (0, 0, 0)$ gives

$$f(BA) = 1.125 + 2\delta^2 + 2a^2 - 1.5\delta a + \tfrac{1}{2}b^2 + O(\|(\delta, a, b)\|^3).$$

The quadratic term is positive definite, establishing the claim.

Compared methods. We compare Landing with retraction and retraction with scale compensation. Initialization and parameter settings are given in Appendix L.5. For the controls, let $\widetilde{B} = B_t - \gamma_t D_{B,t}$, $\widetilde{A} = A_t - \gamma_t D_{A,t}$, and $s = \|\widetilde{B}\|_2$. Retraction sets $(B_{t+1}, A_{t+1}) = (\widetilde{B}/s, \widetilde{A})$; retraction with scale compensation sets $(B_{t+1}, A_{t+1}) = (\widetilde{B}/s, s\widetilde{A})$. The latter leaves the product $\widetilde{B}\widetilde{A}$ unchanged by retraction.

Observed trajectories and interpretation. At the end of the run, ProLand reaches the global minimum with loss 0.5, while both controls reach the local minimum with loss 1.125; all three are feasible and stationary to numerical tolerance. ProLand temporarily leaves the manifold. In panel (a), only the background loss contours minimize over product scale; panel (b) evaluates the actual iterates. The construction and settings are collected in Appendix L.5; the example characterizes when the two trajectories separate.

I ORTHOGONALITY ABLATION

This appendix examines variation in constraint violation in the CodeAlpaca Landing ablation of Section 5.4. The separate OpenBookQA control is reported in Table 5. Training and aggregation settings are collected in Appendix L.4.

Constraint-violation statistics. The largest observed ProLand constraint violation across adapters and updates is 0.053. At the final update, the mean adapter medians are 0.00320, 0.16789, and 0.02079 for ProLand, ProLand without Landing, and per-factor Muon, respectively. ProLand ends lower.

Late-training behavior. Late in training, the ProLand median constraint violation continues to decline across runs, while both controls' medians rise.

J FACTOR-DIRECTION ORTHOGONALITY IN FINITE PRECISION

The implemented optimization directions satisfy the orthogonality condition $B^\top D_B = 0$ to high numerical accuracy at the sampled training states. Sampling and measurement definitions are given in Appendix L.4.

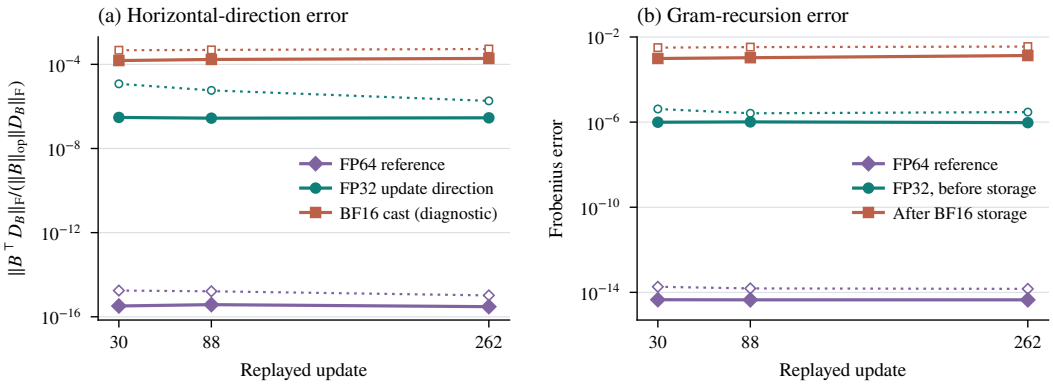

Figure 6: **Factor-direction orthogonality and the effect of parameter storage.** Solid and dotted lines show the median and maximum across runs and adapters. (a) Normalized compatibility error of the FP32 update direction and its BF16 cast. (b) Gram-recursion error before and after BF16 parameter storage.

Orthogonality of the factor directions. Although the sampled bases have constraint violations up to 0.052, the FP32 directions have median compatibility error $e_H = 2.88 \times 10^{-7}$ and maximum 1.19×10^{-5}. The implemented directions therefore satisfy the orthogonality condition to high numerical accuracy.

K CODE-GENERATION EXPERIMENTS

We also evaluate the same adapter methods on HumanEval, MBPP, and their augmented tests. Training and evaluation settings are collected in Appendix L.2.

Table 7: **Code-generation results.** Zero-shot greedy pass@1 (%). Avg. is their arithmetic mean; bold and underline mark the best and second-best scores within each backbone.

Method	HumanEval	HumanEval+	MBPP	MBPP+	Avg.
		Qwen2.5-3B			
AdamW	54.88	51.83	66.67	57.67	57.76
LoRA-Muon	57.32	53.05	67.99	57.67	59.01
Per-factor Muon	56.71	53.05	<u>68.25</u>	56.88	58.72
sMuon	56.71	53.05	67.46	**58.99**	59.05
Riemannion	<u>59.76</u>	<u>54.88</u>	67.99	57.14	<u>59.94</u>
Manifold-LoRA	57.93	53.05	**69.58**	**58.99**	59.89
ProLand (ours)	**60.98**	**56.71**	67.20	<u>57.94</u>	**60.70**
		Llama-3.2-3B			
AdamW	30.49	28.05	**50.53**	<u>39.15</u>	37.05
LoRA-Muon	31.71	26.22	48.41	37.83	36.04
Per-factor Muon	31.71	25.61	49.47	38.62	36.35
sMuon	**33.54**	**29.88**	49.21	<u>39.15</u>	**37.94**
Riemannion	31.10	27.44	<u>50.26</u>	**39.68**	37.12
Manifold-LoRA	<u>32.93</u>	**29.88**	47.88	38.62	37.33
ProLand (ours)	31.71	<u>28.66</u>	49.74	**39.68**	<u>37.45</u>

Results. At rank 8, ProLand achieves the highest average score on Qwen (60.70), outperforming the strongest baseline, Riemannion, by 0.76 percentage points (Table 7). On Llama, it ties Riemannion for the highest MBPP+ score (39.68).

Moonlight code results. On Moonlight-16B-A3B at rank 16, ProLand has the highest four-metric average (59.89), followed by sMuon (59.37) and Per-factor Muon (57.59; Table 8). LoRA-Muon leads on HumanEval and HumanEval+, while ProLand leads on MBPP and MBPP+.

Table 8: **Moonlight-16B-A3B: code benchmarks (%).** Avg. is the arithmetic mean of the four unrounded pass@1 scores; bold and underline mark the best and second-best scores.

Method	HumanEval	HumanEval+	MBPP	MBPP+	Avg.
AdamW	56.10	49.39	61.11	50.79	54.35
LoRA-Muon	**60.98**	**52.44**	62.17	52.12	56.93
Per-factor Muon	54.88	48.78	68.78	57.94	57.59
sMuon	57.93	<u>51.22</u>	<u>69.58</u>	<u>58.73</u>	<u>59.37</u>
Riemannion	47.56	38.41	60.85	49.74	49.14
Manifold-LoRA	53.66	43.90	60.58	49.74	51.97
ProLand	<u>59.15</u>	50.00	**71.16**	**59.26**	**59.89**

L EXPERIMENTAL SETTINGS

L.1 SHARED SETTINGS

We use training seed 2026 for the decoder language and code benchmarks and the loss-trajectory comparisons. The DeBERTa GLUE comparison, CodeAlpaca and OpenBookQA Landing ablations, and factor diagnostics additionally use seeds 1337 and 3407. For the CodeAlpaca orthogonality ablation and factor diagnostics, initialization and data order are paired across settings.

We use the base checkpoints Qwen/Qwen2.5-3B (Qwen Team, 2024), meta-llama/Llama-3.2-3B (Meta, 2024), and moonshotai/Moonlight-16B-A3B (Liu et al., 2025). Table 9 collects the shared training parameters for the main decoder benchmarks. Qwen and Moonlight use their tokenizers' chat templates; Llama uses a fixed Alpaca-style template.

Table 9: Fine-tuning and decoding settings for Qwen, Llama, and Moonlight. Experiment-specific choices are listed in the following tables.

Parameter	Value
Base weights / trainable parameters	Frozen / LoRA factors A, B
Adapter targets	All linear modules; exclude the Moonlight MoE routers
LoRA scaling / dropout / bias	$\alpha = 2r$ / 0 / none
Maximum sequence length / packing	512 / disabled
Objective	Response-token-weighted cross-entropy
Training precision	BF16; FP32 optimizer arithmetic for ProLand and language-benchmark sMuon
Gradient clipping	Global norm 1.0
Benchmark schedule / warmup	Constant / none
Benchmark epochs	3
Adapter ranks	3B language: 4, 8, 16; 3B code: 8; Moonlight and diagnostics: 16
Evaluation checkpoint	Final adapter of the selected run
Code evaluator	EvalPlus 0.3.1; HumanEval+ v0.1.10; MBPP+ v0.2.0
Code decoding	Zero-shot greedy; at most 768 new tokens per problem

L.2 DATA AND EVALUATION

For the decoder benchmarks, we follow sMuon's task and training-data design and its zero-shot evaluation setup (Anson et al., 2026). Each language task has its own adapter. MMLU mixes FLAN and MMLU auxiliary training data in equal proportion (31,129 training and 1,639 holdout rows), and each other language task mixes FLAN, CommonsenseQA, and its task-specific training split in equal proportion (46,694 and 2,458 rows). Source versions and orders are fixed, and smaller sources are cycled deterministically. Code generation and the factor diagnostics use CodeAlpaca-20k (19,014 and 1,002 rows). Effective batches range from 240 to 900 sequences per optimizer update across tasks and models. Language evaluation uses frozen zero-shot likelihood prompts in lm-eval, with no evaluation chat template or subsampling. MMLU, WinoGrande, and BoolQ report accuracy; PIQA, OpenBookQA, HellaSwag, and ARC-Challenge report length-normalized accuracy. We use the test splits for MMLU, OpenBookQA, and ARC-Challenge and the validation splits for the other tasks. MMLU aggregates by question count across subjects; the table average weights the seven tasks equally.

Four-task GLUE validation. We evaluate DeBERTa-v3-base at LoRA rank 16 on the validation splits of RTE, MRPC, STS-B, and CoLA. The task scores are RTE accuracy, the arithmetic mean of MRPC accuracy and F1, the arithmetic mean of STS-B Pearson and Spearman correlations, and CoLA Matthews correlation coefficient (MCC). For each seed, the four-task score is the unweighted mean of these task scores. Table 1 reports the mean and sample standard deviation across the three seeds in Appendix L.1 for each task, using the $n - 1$ denominator with $n = 3$; the average column reports the mean of the four task scores. This average is specific to these four validation tasks and is not a full GLUE score.

Training uses LoRA on the attention query and value projections only, at rank 16 with scaling $\alpha = 32$. The pretrained parameters remain frozen, and the adapters, pooler, and classification head are trained. All methods use the full training set for three epochs with batch size 32 and maximum sequence length 128, a learning rate of 3×10^{-4} for the adapters and the head, and a linear decay after 6% warmup. Parameters and optimizer states are kept in FP32, and the forward pass uses BF16 autocast. Each method retains its native initialization and update rule; ProLand uses $\rho = 0.05$ and weight decay $(0.001, 0)$ on A and B, while the other methods use weight decay 0.01, and the classification head is trained with AdamW throughout. Every method uses a single preset configuration with no additional hyperparameter search and the final checkpoint.

Code evaluation. We evaluate HumanEval (Chen et al., 2021) and MBPP (Austin et al., 2021), together with their augmented tests in EvalPlus (Liu et al., 2023). Each problem receives one zero-shot greedy completion, processed with the EvalPlus sanitizer. The four pass@1 scores in each row use the same final adapter and generated answers; their arithmetic mean gives the average score.

L.3 Parameter selection and final configurations

3B baselines. For each model, rank, and method, we select a language learning rate by accuracy on a deduplicated PIQA training holdout, then use it for all seven task-specific adapters. This selection set contains 794 questions drawn from the training holdout, excluding questions or normalized prompts present in training. For code generation, we select by final response-token-weighted cross-entropy on the CodeAlpaca holdout. Ties prefer the base rate and then the lower rate; code losses within 10^{-12} are treated as tied. Table 10 gives the search grids and training budgets, and Table 11 gives the selected rates. The baseline base rates are taken from sMuon (Anson et al., 2026).

3B ProLand language selection. ProLand follows the same protocol as the baselines: for each model and rank, we select a language learning rate from the corresponding three-point grid by accuracy on the deduplicated PIQA training holdout, and use it for all seven task-specific adapters, with the same tie-breaking rule. The Landing step is fixed at $\rho = 0.05$, and factor A receives no weight decay ($\lambda_A = 0$), in the main ProLand decoder language benchmarks.

Table 10: Candidate grids and selection stages. All rates are base learning rates before any optimizer-specific scaling.

Experiment	Candidates / training budget
3B baselines	$\eta \in \{0.3\eta_0, \eta_0, 3\eta_0\}$; 3 epochs per candidate; η_0 from Table 11
ProLand	Same multipliers and budget; language and code $\eta_0 = 6 \times 10^{-6}$
Factor-spectrum pilot	$\eta \in \{1.75 \times 10^{-7}, 7 \times 10^{-7}, 2 \times 10^{-6}, 7 \times 10^{-6}\}$; linear decay, $\rho = 0.05$

Table 11: Final learning rates for the 3B language and code benchmarks and for Moonlight. 3B language runs use ranks 4, 8, and 16, code runs rank 8, and Moonlight runs rank 16.

		Qwen2.5-3B			Llama-3.2-3B			3B code		Moonlight	
Method	η_0	$r=4$	$r=8$	$r=16$	$r=4$	$r=8$	$r=16$	Qwen	Llama	Lang.	Code
AdamW	0.001	0.0003	0.0003	0.0003	0.001	0.001	0.001	0.0003	0.0003	0.001	0.001
LoRA-Muon	1e-5	1e-5	3e-6	3e-5	3e-5	3e-5	1e-5	3e-6	3e-6	6e-5	6e-5
Per-factor Muon	6e-5	1.8e-5	1.8e-5	1.8e-5	1.8e-5	6e-5	6e-5	1.8e-5	1.8e-5	6e-5	1.8e-5
sMuon	7e-6	7e-6	2.1e-6	7e-6	2.1e-5	2.1e-5	2.1e-5	2.1e-6	7e-6	6e-5	7e-6
Riemannion	1e-5	1e-5	3e-6	1e-5	3e-5	1e-5	3e-5	3e-6	3e-6	6e-5	6e-5
Manifold-LoRA	0.001	0.0003	0.0003	0.001	0.001	0.001	0.001	0.0003	0.0003	0.001	0.001
ProLand	6e-6	6e-6	1.8e-6	1.8e-6	1.8e-5	1.8e-5	1.8e-5	1.8e-6	6e-6	6e-5	7e-6

Code and Moonlight. The 3B ProLand code results use the CodeAlpaca-holdout selection. Moonlight uses the method-specific rates from sMuon (Anson et al., 2026); these runs do not use

the 3B search grids. Table 11 records these rates. Table 12 gives the fixed 3B optimizer parameters; Moonlight ProLand uses the same momentum and code settings, including $\rho = 0.05$ and zero factor weight decay. Manifold-LoRA uses its fixed-step implementation in both Moonlight comparisons.

Table 12: Fixed optimizer settings for the 3B benchmarks. Baseline weight decay is 0.01 with each implementation's decay rule. AdamW and Manifold-LoRA use Adam coefficients $(0.9, 0.999)$ and $\epsilon = 10^{-8}$; the other methods use momentum 0.9. These internal settings are held fixed during the learning-rate searches.

Method	Numerical update settings
AdamW	Standard AdamW update
LoRA-Muon	8 FP32 polar iterations; 7 inverse-root iterations; $\epsilon = 10^{-5}$
Per-factor Muon	Independent factor EMAs; 5 BF16 polar iterations
sMuon	12 polar iterations; 8 inverse-root coefficient iterations; $\epsilon = 10^{-4}$
Riemannion	Low-rank QR/SVD update
Manifold-LoRA	Fixed-step Stiefel–AdamW; restoration coefficient $1/3$
ProLand	Projected Nesterov momentum; numerical iterations and damping as in sMuon; $\rho = 0.05$, zero weight decay on both factors

Loss-trajectory comparison. Figure 5 uses fixed three-epoch Qwen runs at rank 8: the baselines use Table 11, and ProLand uses $\rho = 0.05$ with zero factor weight decay. We plot nine-update moving averages of training loss and evaluate validation loss after each epoch.

L.4 Feasibility ablations and factor diagnostics

The constraint-violation trajectory experiment and factor diagnostics fine-tune Qwen2.5-3B on the CodeAlpaca split described above. Table 13 collects their training parameters and sampling positions. The separate OpenBookQA control is specified below.

Table 13: Orthogonality and factor-diagnostic settings. Both use $\beta = 0.9$, $\lambda_A = 0$, $\lambda_B = 0$, orthogonal B and zero A initialization, FP32 updates, and BF16 factors and optimizer states.

Parameter	Landing ablation	Factor spectra
Learning rate	7e-6	7e-7 (peak)
Effective batch / epochs	640 / 10	16 / 3
Warmup	None	100 updates
Schedule	Constant	Linear decay to zero; constant
Landing strength ρ	0, 0.05	0, 0.05, 0.45
Validation	Each epoch	Four times per epoch
Late-training interval	Final 30% of updates	Final 30% of updates
Direction replay	First update after epochs 1, 3, 9	Not used
Factor measurements	Across all adapters in each run	FP64; 252 adapters per run, 18 runs

CodeAlpaca Landing ablation. Figure 2 compares ProLand, the same optimizer without Landing, and per-factor Muon. The Landing-free ablation sets $\rho = 0$ while leaving the rest of the implementation unchanged. These runs use rank 16 and LoRA scaling $\alpha = 32$, sequence length 512, and NVIDIA H100 80GB GPUs. A microbatch of 32 sequences with 20 gradient-accumulation steps gives an effective batch of 640; each run completes 290 optimizer updates. Per-factor Muon shares the initial factors, learning rate, and effective weight decay, and uses independent factor EMAs and BF16 polar iterations as in Table 12, without Landing or retraction. For each run, we compute the median, 90th percentile, and maximum constraint violation across adapters; the plotted curves average these statistics and the shading spans their range.

OpenBookQA Landing ablation. Table 5 uses Qwen2.5-3B base with the official OpenBookQA training set, split into 4,709 training and 248 held-out examples (split seed 20260904). Question

stems and full correct answers are rendered with the tokenizer's chat template; only response tokens contribute to loss, with no sequence packing. The frozen model has rank-16 LoRA ($\alpha = 32$, dropout 0) on the query, key, value, output, gate, up, and down projections of all 36 layers (252 factor pairs). Within each shared seed, all variants have byte-identical initial factors: B is obtained by QR of an FP32 Gaussian matrix, $A = 0$, and momentum is zero.

All variants use three epochs (108 updates), maximum length 256, and batch 128 (microbatch 32, accumulation 4), discarding incomplete accumulation windows. The base learning rate is constant at 7×10^{-6} without warmup; the shared `match_rms_adamw` scaling gives $\eta = 7 \times 10^{-6} \cdot 0.2\sqrt{d_{\text{out}} d_{\text{in}}/r}$. Nesterov momentum is 0.9, $(\lambda_A, \lambda_B) = (0.001, 0)$, global gradient clipping is 1.0, and Gram inverse-root regularization is 10^{-4}. Parameters and persistent momentum are BF16, update kernels are FP32, and TF32 is disabled. Each method uses one fixed configuration, zero additional search, and the final checkpoint. Primary results comprise nine H100 runs.

ProLand uses $\rho = 0.05$, subtracting $\rho B(B^\top B - I)$ independently of η. Without Landing sets only $\rho = 0$. With retraction also sets $\rho = 0$ and maps the tentative update $\widetilde{B}$ to $B_{\text{new}} = \widetilde{B}(\widetilde{B}^\top \widetilde{B})^{-1/2}$ at every step, using FP32 symmetric eigendecomposition. This is the standard polar retraction of the basis factor in formulations with B constrained to the Stiefel manifold. All variants then apply the same projected product-momentum projection using the updated FP32 factors before BF16 storage.

We evaluate all 500 test questions by zero-shot continuation likelihood: the prompt is the question stem and candidates are full option texts, with no evaluation chat template. Table 5 reports ordinary accuracy from summed log-likelihoods, computed with FP32 log-softmax on BF16 model outputs. Final constraint violations $\|B^\top B - I\|_{\text{op}}$ are measured in FP64 after BF16 storage. We first compute the median, 90th percentile, and maximum across the 252 factors within each run, then report their means and sample standard deviations across seeds, in units of 10^{-3}. Accuracy uses the same across-seed summary.

Factor spectra. Figures 3 and 4 compare Landing strengths under linear decay and constant learning rates. We choose the peak learning rate in a separate pilot by the mean held-out loss over the final three validation evaluations, with ties resolved in favor of the smaller rate, and keep it fixed across Landing strengths and schedules. For the constraint violation of B, median and percentile curves average the per-run statistics, whereas maximum curves take the worst run and adapter; shading shows the range across runs. The A curves take extrema over all runs and adapters. Gray regions denote warmup.

Direction diagnostics. Figure 6 replays the first update after selected completed epochs of the Landing-ablation runs, as specified in Table 13. For the optimization direction, excluding Landing, we measure $e_H = \|B^\top D_B\|_{\text{F}} / (\|B\|_{\text{op}} \|D_B\|_{\text{F}})$. An FP64 reference applies (7) and (3) to the same stored factors and polar-map inputs. For the exact Gram recurrence in Section 4, we measure $\|C^+ - P(C, D_B)\|_{\text{F}}$ before and after BF16 storage, using each adapter's actual step size, and aggregate over runs and adapters.

L.5 Two-factor numerical example

For the example in Appendix H, all methods use the momentum-projection, polar-map-input, polar-factor, and factor-direction steps of Algorithm 1, recomputed at their own current factors, with $G_t = \nabla f(B_t A_t)$, $\beta = 0.9$, and $\bar{H}_{-1} = 0$. We initialize $B_0 = (1/2, \sqrt{3}/2)^\top$ and $A_0 = 1.5 B_0^\top$. Both factors are updated simultaneously, and each method projects its previous EMA state into its current differential image. The shared schedule is

$$\gamma_t = 0.2\,(0.97)^{\max\{0, t-100\}}, \qquad t = 0, \ldots, 799.$$

ProLand uses $\rho = 0.3$. This decaying schedule is chosen for the illustration and differs from the horizon-tuned schedule of the convergence bounds.