

Matching Accuracy, Different Geometry: Evolution Strategies vs GRPO in LLM Post-Training

William Hoy¹ Binxu Wang^{2*} Xu Pan^{3*}

¹ Department of Industrial and Systems Engineering, University of Miami

² Kempner Institute, Harvard University ³ Harvard University

¹ wjh58@miami.edu ² binxu_wang@hms.harvard.edu ³ xupan@fas.harvard.edu

Abstract

Evolution Strategies (ES) have emerged as a scalable gradient-free alternative to reinforcement learning based LLM fine-tuning, but it remains unclear whether comparable task performance implies comparable solutions in parameter space. We compare ES and Group Relative Policy Optimization (GRPO) across four tasks in both single-task and sequential continual-learning settings. ES matches or exceeds GRPO in single-task accuracy and remains competitive sequentially when its iteration budget is controlled. Despite this similarity in task performance, the two methods produce markedly different model updates: ES makes much larger changes and induces broader off-task KL drift, whereas GRPO makes smaller, more localized updates. Strikingly, the ES and GRPO solutions are linearly connected with no loss barrier, even though their update directions are nearly orthogonal. We develop an analytical theory of ES that explains all these phenomena within a unified framework, showing how ES can accumulate large off-task movement on weakly informative directions while still making enough progress on the task to match gradient-based RL in downstream accuracy. These results show that gradient-free and gradient-based fine-tuning can reach similarly accurate yet geometrically distinct solutions, with important consequences for forgetting and knowledge preservation. The source code is publicly available.¹

1 Introduction

Fine-tuning large language models (LLMs) is essential for adapting them to new tasks and for maintaining performance in continual learning settings. When explicit supervised targets are unavailable, optimization must proceed from evaluative feedback rather than token-level labels. In this regime, gradient-based reinforcement learning methods such as Group Relative Policy Optimization (GRPO) have become a standard component of LLM post-training (Shao et al., 2024; Guo et al., 2025), while Evolution Strategies (ES) have recently emerged as a scalable gradient-free alternative (Qiu et al., 2025; Korotyshova et al., 2025; Sarkar et al., 2025). ES replaces backpropagation with population-level perturbation and reward aggregation, offering potential advantages in memory usage and parallelism.

Despite growing interest in ES for LLM alignment and fine-tuning, its behavior relative to gradient-based methods remains poorly understood. Prior works present a mixed picture. Some studies (Qiu et al., 2025) suggest that ES can achieve competitive performance while retaining low KL divergence to the base model. This raises the possibility that it may be well suited for continual learning, where preserving prior capabilities is important. Other work (Abdi et al., 2026), however, finds that ES can induce substantial forgetting, attributing this to larger and less sparse parameter updates. Taken together, these results leave open a basic question: when ES and gradient-based RL achieve similar task performance, do they

*Equal advising.

¹<https://github.com/Bhoy1/ESvsGRPO>

induce similar off-target effects (e.g. forgetting), and do the solutions have similar geometric properties (e.g. norm and dimensionality)?

In this study, we compare ES and RL in a setup where an LLM is trained sequentially across four diverse tasks and extend the analysis with a geometric characterization of the solutions each method finds. Our analysis reveals that despite equivalent accuracy across four diverse tasks, ES and GRPO navigate the loss landscape in different ways. GRPO identifies a narrow, high gradient direction, making precise and targeted updates that stay close to the base model. ES navigates a broad, low curvature plateau, accumulating large diffuse updates that travel orders of magnitude further from initialization. Strikingly, despite these near orthogonal trajectories and orders of magnitude norm differences, no loss barrier exists between the two solutions. Moreover, when we probe the loss landscape along each method’s learned update direction, the ES direction behaves much more like a random direction on held-out tasks, indicating that a substantial fraction of its displacement lies in task-irrelevant dimensions rather than in a sharply task-aligned subspace.

Finally, we derive a theoretical account of these geometric differences. ES weight updates decompose into two components: an on-manifold (task relevant) component, and an off-manifold (task-invariant) component. In the off-manifold subspace, ES is equivalent to a random walk, which exhibits a characteristic scaling: the squared norm of the weight change grows as $\alpha^2 dT/N$, proportional to parameter dimension d and step count T , and inversely proportional to population size N . In a high-dimensional landscape with many flat dimensions — this off-manifold diffusion dominates the weight change. Empirically, LLM parameters (particularly weight matrices) precisely exhibit this predicted random walk behavior, with the theoretical scaling matching observations at $R^2 > 0.99$. This single mechanism accounts for all the geometric phenomena we observe: the large update norm, the near-orthogonality between ES and GRPO update directions, the flatter loss curvature along ES weight changes, and the linear mode connectivity despite geometric separation.

Our contributions are:

1. Empirical evidence that ES can match or exceed GRPO in task accuracy across both single-task and sequential fine-tuning settings.
2. A geometric characterization of ES versus GRPO solutions, revealing that equivalent task performance can be achieved with orders of magnitude norm differences.
3. A theoretical account of the random-walk-like behavior of ES in high-dimensional weakly informative subspaces, explaining its large update norms and diffuse-like behavior relative to GRPO.

2 Related Work

Evolution strategies for LLMs. Salimans et al. (2017) demonstrated ES as a scalable alternative to reinforcement learning via high parallelizability. Recent work has brought ES to LLM scale: Qiu et al. (2025) showed ES achieving comparable performance to GRPO on reasoning tasks, attributing this to LLMs’ low intrinsic dimensionality (Aghajanyan et al., 2021) and ES’s smoothing of jagged reward landscapes via Gaussian convolution in parameter space (Lehman et al., 2018). Korotyshova et al. (2025) and Sarkar et al. (2025) further showed that ES can be made practical for large-scale alignment through low-rank parameterizations that improve efficiency and scalability. Liang et al. (2026) showed that fine-tuning loss landscapes are low-dimensional in curvature, helping explain why ES succeeds with small populations. However, Abdi et al. (2026) demonstrates that ES causes significant forgetting compared to GRPO on smaller models (Qwen2.5-1.5B and Llama-3.2-1B), attributing it to larger update norms and lower sparsity.

Gradient-based RL for LLMs. Shao et al. (2024) introduced GRPO, which eliminates the critic model by estimating advantages from group statistics, reducing memory overhead compared to PPO. It became a standard post-training method following its use in DeepSeek-R1 (Guo et al., 2025). Several concurrent works have investigated why GRPO preserves

prior capabilities during fine-tuning. Shenfeld et al. (2025) showed that forgetting correlates with KL divergence from the base model, while Chen et al. (2025) showed that GRPO’s robustness may stem more from its use of on-policy data rather than KL regularization or advantage estimation.

3 Methodology

We compare two fundamentally different approaches to fine tune large language models: ES, a gradient free method, and GRPO, a gradient based reinforcement learning algorithm.

3.1 Evolution strategies

ES optimizes model parameters through population based search without computing gradients. At each iteration, we sample N perturbation vectors $\epsilon_i \sim \mathcal{N}(0, I)$, evaluate the perturbed models, and update parameters using z score normalized rewards:

$$\theta_{t+1} = \theta_t + \alpha \cdot \frac{1}{N} \sum_{i=1}^N Z_i \epsilon_i \quad (1)$$

where α is the learning rate, $Z_i = \frac{R_i - \mu_R}{\sigma_R}$ is the z score normalized reward for the i -th perturbation, and $R_i = R(\theta_t + \sigma \epsilon_i)$ is the reward of the perturbed model with noise scale σ . Following Qiu et al. (2025), we maintain $\alpha = \sigma/2$ throughout and apply ES updates to the complete model weights.

3.2 Group relative policy optimization

GRPO (Shao et al., 2024) normalizes advantages within groups of sampled responses, eliminating the need for a critic model. For a prompt x , we sample K responses $\{y_1, \dots, y_K\}$ from the current policy π_θ and compute group-normalized advantages: $\hat{A}_i = \frac{R(y_i) - \mu_R}{\sigma_R + \epsilon}$, where μ_R and σ_R are the within group mean and standard deviation of rewards. The policy is updated by maximizing the clipped surrogate objective: $\mathcal{L}_{\text{GRPO}}(\theta) = \mathbb{E} \left[\frac{1}{K} \sum_{i=1}^K \min(\rho_i(\theta) \hat{A}_i, \text{clip}(\rho_i(\theta), 1 - \epsilon, 1 + \epsilon) \hat{A}_i) \right]$, where $\rho_i(\theta) = \frac{\pi_\theta(y_i|x)}{\pi_{\theta_{\text{old}}}(y_i|x)}$ is the probability ratio between the current and old policy. We disable the KL penalty ($\beta_{\text{KL}} = 0$) during training to induce a clean comparison with ES without explicit regularization (Shenfeld et al., 2025), but track KL divergence to the reference policy for analysis in both methods.

3.3 Experimental setup

All experiments use Qwen3-4B-Instruct-2507 as the base model. We evaluate on four diverse tasks spanning reasoning, mathematics, scientific knowledge, and reading comprehension: Countdown (arithmetic reasoning), Math (mathematical problem solving), SciKnowEval-Chemistry (chemistry domain knowledge), and BoolQ (boolean reading comprehension). Each task uses 200 training samples and 2,000 test samples (500 for Math). In the continual learning experiments, tasks are presented sequentially. Hyperparameter sweeps were conducted for both methods (see Appendix C.3); we selected $N = 30$, $\sigma = 0.0015$, $\alpha = 0.00075$ for ES and $\text{lr} = 1 \times 10^{-5}$ for GRPO. An additional model, Llama-3.2-3B-Instruct, can be found in the Appendix.

4 Empirical Results

4.1 Task performance

Table 1 compares the task accuracy of the base model against ES and GRPO in a single task training setup. Running ES for 300 iterations consistently yields the highest peak accuracy

across all four tasks. For instance, on the Chemistry task, ES (300) achieves 76.5% accuracy, significantly outperforming both ES (100) at 68.1% and GRPO at 74.9%.

However, when the model is trained sequentially across the tasks (Figure 1), utilizing 300 ES iterations causes pronounced degradation in earlier tasks as the training progresses. Limiting ES to 100 iterations offers a balance. While it sacrifices a small degree of peak single task accuracy compared to the 300 iteration configuration, it maintains sequential training stability. Both ES (100) and GRPO remain comparatively stable throughout the entire sequential training pipeline, avoiding the sharp drop-offs seen with ES (300). Because of this, we use an iteration count of 100 for ES in all of the following sequential experiments.

Table 1: Accuracy (%) for single task training.

Task	Base	ES (iter=100)	ES (iter=300)	GRPO
Countdown	20.5	75.1	78.6	74.5
Math	61.8	72.4	74.0	68.8
Chemistry	63.4	68.1	76.5	74.9
BoolQ	83.7	87.2	88.1	86.6

As a side note, ES and GRPO show different transfer dynamics in the sequential task training. ES exhibits notable zero-shot forward transfer, i.e., learning a task increases performance on an unseen task. Chemistry accuracy rises from 63% to approximately 69% before its training stage begins, and Math accuracy increases by roughly 3 points after Countdown training alone. This forward transfer is not universal and depends on the tasks, as BoolQ accuracy dips slightly below the base model before its training stage. GRPO shows little forward transfer by comparison.

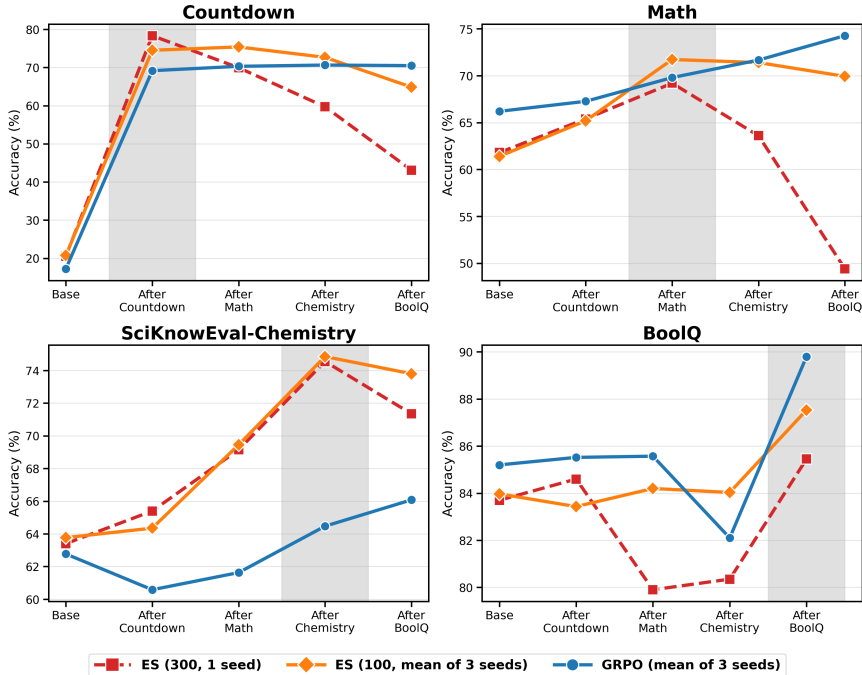


Figure 1: Accuracy (%) on each task throughout sequential training. The shaded region marks the stage at which that task was trained. ES (300) shows clear degradation on earlier tasks as training progresses, while ES (100) and GRPO remain comparatively stable.

4.2 Forgetting

To test whether fine-tuning affects general capabilities beyond training tasks, we evaluate both methods on two held-out benchmarks after each training stage: MMLU for general knowledge, and IFEval (strict) for instruction-following ability. The two methods diverge on

MMLU. ES accuracy drops monotonically from 77.5% to 73.8% across the four training stages (−3.7%), with the largest single-stage drop during Chemistry training (−1.6%). GRPO, by contrast, remains stable or slightly improves throughout, ending at 78.3% (+0.8%). On IFEval, the pattern reverses: ES gains +2.2%, peaking after Math training, while GRPO rises through Chemistry but drops after BoolQ training, ending at −1.3% below baseline.

Table 2: Holdout benchmark accuracy (%) throughout sequential training.

Stage	ES (MMLU)	GRPO (MMLU)	ES (IFEval)	GRPO (IFEval)
Base model	77.5	77.5	79.7	79.7
After Task 1 (Countdown)	77.0	77.0	80.0	80.6
After Task 2 (Math)	76.2	77.5	82.6	80.6
After Task 3 (Chemistry)	74.6	78.0	82.3	81.7
After Task 4 (BoolQ)	73.8	78.3	81.9	78.4
Δ (base $\rightarrow$ final)	−3.7	+0.8	+2.2	−1.3

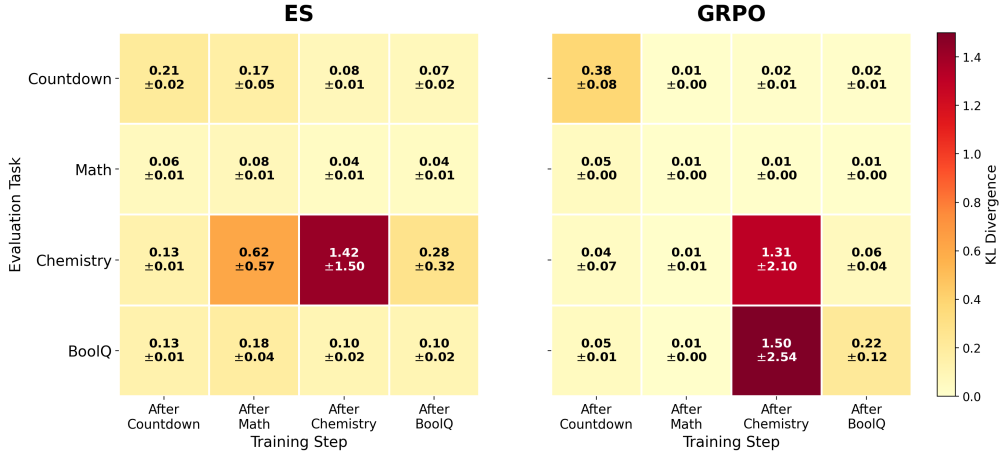


Figure 2: Incremental KL divergence per training step. Each cell shows $KL(\pi_{\text{after}} \parallel \pi_{\text{before}})$ for a given evaluation task (row) after a given training step (column). Both panels share the same color scale. Across three seeds there is large variance for some blocks. We have included median, min and max in Appendix C.4 so the complete picture can be seen

We measure the incremental KL divergence $KL(\pi_{\text{after}} \parallel \pi_{\text{before}})$ on each task’s held-out set after every training step, capturing how much each step shifts the model’s output distribution on every task (Figure 2). Qiu et al. (2025) find that ES produces substantially lower KL divergence than GRPO on a single-task conciseness benchmark. The first column of our heatmap (T0, Countdown only) is broadly consistent: ES’s on-task KL (0.21 nats) is slightly lower than GRPO’s (0.38 nats). In the sequential setting, however, ES produces larger off-diagonal KL shifts—training on any task causes non-trivial drift on all other tasks (e.g., training on Math shifts Chemistry by 0.62 nats for ES vs. 0.01 for GRPO). ES’s on-diagonal drift is also more aggressive: Chemistry training induces 1.42 nats for ES versus 1.31 for GRPO. There is a single example where GRPO induces larger off diagonal than ES but if you compared medians in Appendix C.4 than it goes away. This is consistent with our weight-space findings: ES’s isotropic perturbations spread change broadly across the output distribution, while GRPO’s directed updates concentrate it on the trained task. The discrepancy with Qiu et al. (2025) in the sequential setting may reflect differences in task structure or model, but it suggests the KL advantage of ES is not universal.

To quantify off-diagonal drift directly, we compute the ratio of off-task to on-task incremental KL separately for each seed s :

$$\rho_s = \frac{\sum_t \kappa_{t,s}^{\text{off}}}{\sum_t \kappa_{t,s}^{\text{on}}},$$

where $\kappa_{t,s}^{\text{on}}$ is the test-set incremental KL on the task trained at transition t , and $\kappa_{t,s}^{\text{off}}$ is the mean test-set incremental KL over the other three tasks at the same transition. Thus, $\rho_s \rightarrow 0$ indicates that drift is concentrated on-task, whereas $\rho_s \approx 1$ indicates that average off-task drift is comparable to on-task drift. For Qwen3-4B-Instruct-2507, we compute ρ_s separately for seeds 42, 52, and 62 and report the mean and sample standard deviation across seeds. ES yields $\bar{\rho}_{\text{ES}} = 0.533 \pm 0.407$, compared with $\bar{\rho}_{\text{GRPO}} = 0.228 \pm 0.108$ for GRPO. The ES mean is therefore $2.34\times$ the GRPO mean, indicating greater off-task drift relative to on-task change under ES. The large variation across seeds further indicates substantial seed-dependent heterogeneity in the magnitude of this effect.

We additionally report cumulative test-set KL divergence from the original base model throughout training, $D_{\text{KL}}(\pi_{\theta_t} \parallel \pi_{\theta_0})$ (Appendix C.5).

5 Weight Space Geometry Analysis

Despite both ES and GRPO achieving similar accuracy when trained with a single task, their difference in the sequential task experiments suggests that the weight changes of both could be distinct. We characterize the geometry of their weight changes, seeking a deeper understanding of the difference between the two methods.

5.1 ES induces larger weight changes

We measure the total ℓ_2 norm of each method’s weight difference from the base model:

$$\|\Delta\theta\|_2 = \|\theta_{\text{trained}} - \theta_{\text{base}}\|_2 \quad (2)$$

Table 3 shows this at each sequential checkpoint. ES updates grow from 87.28 to 173.00 across the four tasks, while GRPO updates remain two orders of magnitude smaller (1.00 to 1.84), yielding norm ratios of 87 to $107\times$.

Table 3: Weight update norms ($\|\Delta\theta\|_2$) at each sequential checkpoint, measured from the base model.

Checkpoint	ES norm	GRPO norm	Ratio
After Task 1 (Countdown)	87.28	1.00	$87\times$
After Task 2 (Math)	122.76	1.15	$107\times$
After Task 3 (Chemistry)	149.99	1.65	$91\times$
After Task 4 (BoolQ)	173.00	1.84	$94\times$

5.2 ES updates are denser and higher rank

The ℓ_2 norm alone conflates several ways of geometric change; a large norm can arise from large changes along a few coordinates or from small changes spread broadly across many parameters. To disambiguate these, we measure two other properties of each method’s weight update at every sequential checkpoint: *sparsity*, the fraction of update entries $\Delta\theta_i$ with $|\Delta\theta_i| \leq \varepsilon$ (using $\varepsilon = 1 \times 10^{-5}$), and *stable rank*, $\|\Delta W\|_F^2 / \|\Delta W\|_2^2$ (Frobenius norm over spectral/operator norm), which quantifies the effective dimensionality of the update for matrix-shaped parameters. Table 4 reports both measures. GRPO’s updates are substantially sparser (87.7–88.0% near-zero entries, versus 2.1–3.8% for ES) and lower rank (60–103, versus 1078–1100 for ES). This is consistent with our picture (Sec. 6): ES’s isotropic, high-dimensional diffusion produces dense, high-rank updates, while GRPO’s targeted, gradient-aligned updates are concentrated in a sparse, low-rank subspace.

Table 4: Sparsity (fraction of entries with $|\Delta\theta_i| \leq \varepsilon = 1 \times 10^{-5}$) and stable rank ($\|\Delta W\|_F^2 / \|\Delta W\|_2^2$, averaged across attention QKVO projections, and MLP up/down/gate projections) of ES and GRPO weight updates at each sequential checkpoint.

Stage	Sparsity ES	Sparsity GRPO	Stable Rank ES	Stable Rank GRPO
T0 Countdown	3.8%	88.0%	1100	60
T1 Math	2.9%	88.0%	1090	70
T2 Chemistry	2.4%	87.7%	1084	89
T3 BoolQ	2.1%	87.7%	1078	103

5.3 ES and GRPO solutions are linearly mode-connected

We evaluate whether ES and GRPO solutions occupy the same loss basin by testing linear interpolations (i.e. linear mode connectivity) at each sequential checkpoint:

$$\theta(\alpha) = (1 - \alpha) \theta_{\text{ES}} + \alpha \theta_{\text{GRPO}}, \quad \alpha \in \{0, 0.25, 0.5, 0.75, 1\} \quad (3)$$

A loss barrier would manifest as degraded performance at intermediate α values relative to both endpoints. Figure 3 shows average accuracy across tasks at each interpolation point. No significant loss barrier exists at any checkpoint: performance transitions smoothly, and intermediate points frequently match or outperform the endpoints, demonstrating that ES and GRPO remain in the same loss basin throughout sequential training.

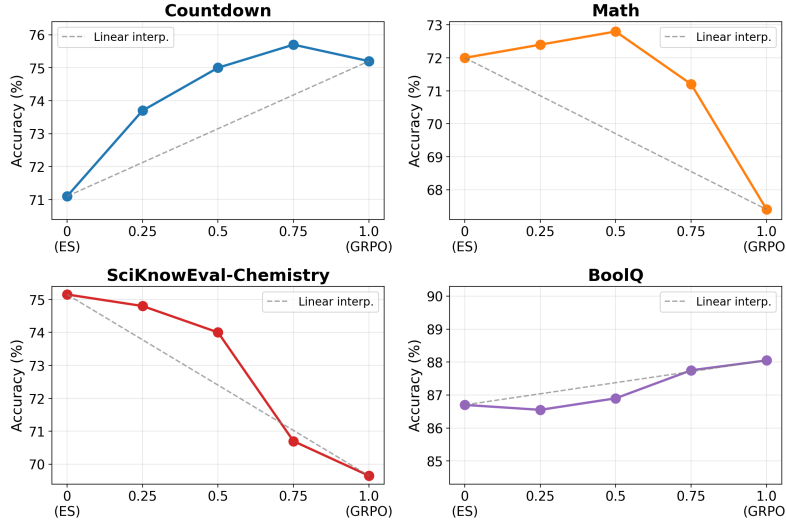


Figure 3: Per-task accuracy along the linear interpolation path between the final ES and GRPO checkpoints. The dashed gray line shows the linear interpolation between endpoint accuracies. Across all tasks, the interpolated model remains close to or above the linear baseline with no catastrophic accuracy drop, indicating that the two solutions lie in the same basin of the loss landscape.

5.4 ES solution direction behaves similarly as random directions

To understand why ES and GRPO produce such different weight geometries despite similar task accuracy, we probe the loss landscape along each method’s learned update direction. Starting from the base model, we move along the normalized ES or GRPO weight delta at increasing magnitudes:

$$\theta_{\text{perturbed}} = \theta_{\text{base}} + m \cdot \frac{\Delta\theta}{\|\Delta\theta\|_2} \quad (4)$$

where m ranges from 0 to $\|\Delta\theta\|_2$ (the trained checkpoint) and beyond. Random directions, averaged over three seeds, serve as a control. By construction, $m = \|\Delta\theta\|_2$ exactly recovers the trained checkpoint.

Figure 4 shows that along the GRPO direction, task accuracy rises rapidly from the base-model level to the checkpoint level within a very small displacement, then degrades quickly once the perturbation magnitude exceeds the trained solution. Along the ES direction, accuracy improves more gradually over a much larger displacement scale. Both methods reach comparable final accuracy, but GRPO concentrates its useful signal into a much smaller norm, while ES spreads it across a larger perturbation, consistent with ES containing a larger component weakly coupled to the target task.

Evaluating perturbed models on holdout tasks reinforces this picture (Fig. 5). The ES and random directions produce similar holdout degradation profiles, consistent with the ES

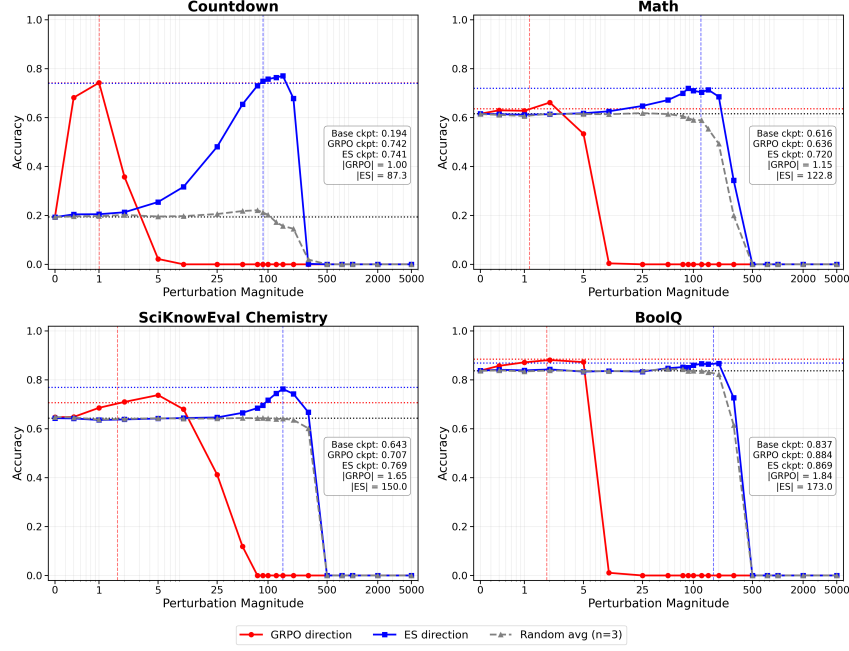


Figure 4: Task accuracy when moving from the base model along ES (blue), GRPO (red), and random (gray) directions at increasing magnitudes. Dashed vertical lines mark $\|\Delta_{GRPO}\|$ (red) and $\|\Delta_{ES}\|$ (blue). Dotted horizontal lines show base and checkpoint accuracies.

update being a non-targeted diffused shift. GRPO’s direction is more effective per unit norm on the target task, but degrades general capabilities faster than either ES or random directions, consistent with its updates being concentrated in a narrow task-specific subspace.

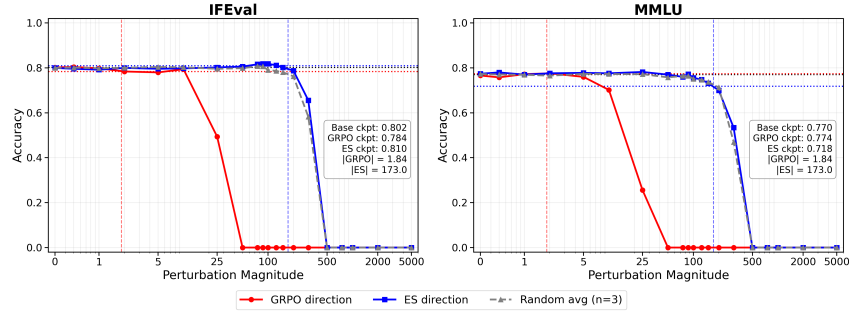


Figure 5: Holdout task accuracy (IFEval, MMLU) when perturbing along ES (blue) and GRPO (red) directions from the final BoolQ checkpoint.

6 Theoretical Interpretation

The experiments show a consistent pattern: ES and GRPO reach comparable task accuracy, yet ES accumulates much larger weight changes, resembles random directions on holdout probes, and remains linearly connected to GRPO solution in accuracy with no loss barrier. We now develop a theoretical account that explains these phenomena within a unified framework (with formal statements in App. D).

Signal-diffusion decomposition. Consider a quadratic potential centered at $\mathbf{0}$ with curvature Q , $\mathcal{L}(\theta) = \frac{1}{2}\theta^\top Q\theta$. Idealizing ES as a fixed step-size Ornstein–Uhlenbeck process surrogate, the displacement after T steps decomposes as (Prop. 4):

$$\theta_{\text{ES}} - \theta_0 = \underbrace{(H^T - I)\theta_0}_{\text{signal (GD-like)}} + \underbrace{\sum_{t=1}^T H^{t-1} \eta_{T-t}}_{\text{diffusion}}, \quad (5)$$

with $H := I - \frac{\alpha\sigma}{\sigma_R}Q$ and i.i.d. noise $\eta_t \sim \mathcal{N}(0, \frac{\alpha^2}{N}I_d)$. The first term drives parameters toward the optimum and resides in the column space of Q , resembling gradient descent with $\frac{\alpha\sigma}{\sigma_R}$ playing the role of learning rate; the second term is diffusion in d dimensions. Along flat directions ($\mathbf{u}$ where $Q\mathbf{u} = 0$), the signal vanishes while the diffusion persists. Noiseless gradient descent – an idealized view of GRPO, produces only the first term (Prop. 6).

Why diffusion dominates On a flat landscape, the ES update is a pure random walk (Prop. 1): $\Delta\theta_t \sim \mathcal{N}(0, \frac{\alpha^2}{N}I_d)$, and the displacement concentrates at its expectation,

$$\mathbb{E}[\|\theta_T - \theta_0\|^2] = \alpha^2 T d / N. \quad (6)$$

This scaling—linear in T and d , inverse in N —persists in the off-manifold (i.e. reward irrelevant) subspace even when the landscape is non-flat. On linear and quadratic landscapes, we derive the full single-step statistics analytically (Prop. 2, 3). On a linear landscape, the fraction of gradient-aligned squared displacement per step is (App. F.3):

$$\chi = (1 + (N+1)s) / (d + (N+1)s), \quad (7)$$

where $s := \sigma^2 \|\mathbf{v}\|^2 / \sigma_R^2 \in [0, 1]$ is the *signal fraction*, with $\|\mathbf{v}\|$ the gradient norm. Since $N \ll d$ in practice ($N = 30, d \approx 4 \times 10^9$), χ is negligible, i.e. most update is not gradient aligned. Further, on a quadratic landscape, with rank- r curvature, over T steps (Prop. 7):

$$\mathbb{E}\|\theta_{\text{ES}} - \theta_0\|^2 = \underbrace{\mathcal{O}(r)}_{\text{on-manifold}} + \underbrace{\alpha^2 T(d-r)/N}_{\text{off-manifold}} \approx \alpha^2 T d / N. \quad (8)$$

Geometric consequences The decomposition yields a norm hierarchy (Propositions 6–8):

$$\|\theta_{\text{GD}} - \theta_0\|^2 = \mathcal{O}(r), \quad \mathbb{E}\|\theta_{\text{ES}} - \theta_0\|^2 = \mathcal{O}(r) + \mathcal{O}(d), \quad \mathbb{E}\|\theta_{\text{ES}} - \theta_{\text{GD}}\|^2 = \mathcal{O}(d), \quad (9)$$

and the expected cosine similarity scales as $\mathcal{O}(\sqrt{r/d})$ (Remark 9): nearly orthogonal despite similar task performance. The key difference is on flat directions: GD freezes at initialization while ES diffuses with variance $\alpha^2 t / N$ (Propositions 4, 5).

6.1 Empirical validation and explanatory power

Random walk scaling. For the full parameter vector, $\|\Delta\theta\|^2$ grows linearly with step count in both models ($r > 0.9999$ for ES against $r \approx 0.87$ for GRPO; Fig. 6A). Inverting the slope to infer the effective dimension yields $d_{\text{eff}}/d_{\text{total}} = 96.4\%$ (Qwen) and 102.4% (Llama)—nearly all dimensions behave as flat, the two models bracketing the parameter-free prediction within 4%. Per-parameter analysis localizes the departure: large weight matrices follow the random walk scaling almost exactly (0.968 ± 0.014 and 1.024 ± 0.031 , $R^2 = 0.9999$), while gains of normalization layers deviate strongly and differently across architectures (1.75 ± 0.87 in Qwen, 0.40 ± 0.40 in Llama; Fig. 6B,C; see App. E.1). In summary, ES dynamics is dominated by a random walk through an almost flat landscape, with a $< 4\%$ correction from the task-relevant directions—where curvature compresses the walk, and a gradient-like component inflates it.

Explaining each observation. Our framework explains: *Large ES update norms* (Table 3): ES is dominated by the $\mathcal{O}(d)$ off-manifold term while GRPO stays at $\mathcal{O}(r)$, producing the $\sim 100\times$ ratio. *Near orthogonal updates* (App. E.2): the cosine similarity of GD and ES weight updates is around $\sim 10^{-4}$ (Tab. 20), consistent with the predicted scale $\mathcal{O}(\sqrt{r/d})$. *Flat mode connectivity* (Fig. 3): after convergence, the $\theta_{\text{ES}} - \theta_{\text{GD}}$ line lies in the off-manifold subspace where loss is flat. *ES resembles random* (Fig. 4): dominant fraction of ES update comes from off-manifold random walk, explaining their similarity and smaller directional

curvature along ES updates. **Holdout degradation** (Fig. 5): the fine-tuning landscape sits atop a broad plateau of base-task performance; off-manifold perturbations beyond a certain radius degrade general capabilities regardless of task relevance, explaining the matched ES–random degradation profiles. This picture is summarized in Figure 6D.

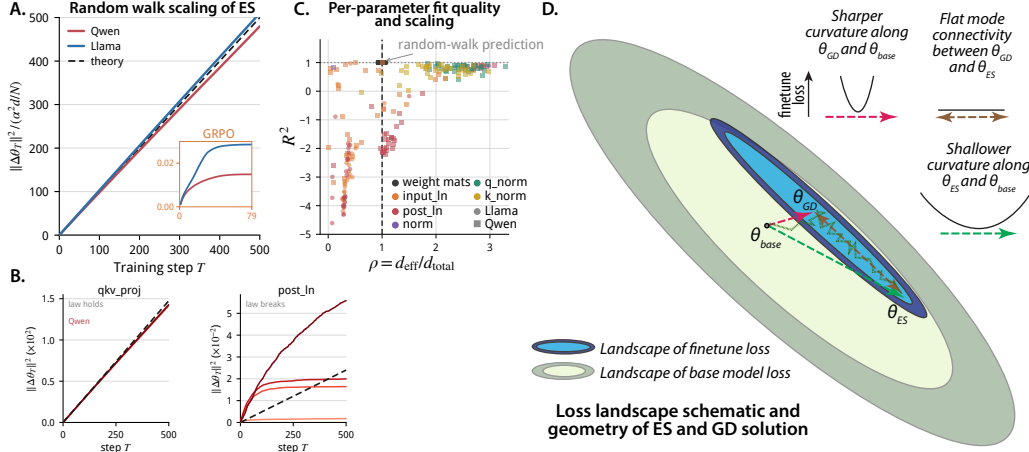


Figure 6: **ES fine-tuning drifts as a random walk through the parameter space; GRPO does not.** **A.** Total drift $\|\Delta\theta_T\|^2$ for Qwen and Llama, each divided by its own random-walk scaling $\alpha^2 d / N$ so that the theory becomes the identity (dashed). Inset: GRPO with the same normalisation. **B.** Individual Qwen tensors against the same scaling (dashed): qkv_proj follows it for the whole run, post_ln saturates away from it. **C.** Effective dimension ratio $\rho = d_{\text{eff}} / d_{\text{total}}$ against fit quality R^2 for every tensor. All weight matrices collapse onto the ideal scaling $R^2 = 1, \rho = 1$, the norm vectors do not. **D.** GRPO moves along the low-dimensional task-relevant subspace (red). ES accumulates both a task-relevant component and a large off-manifold random walk (green), yielding a much larger displacement nearly orthogonal to GRPO. The two solutions are linearly connected because the off-manifold component is loss-invariant.

Remark 1. High-dimensional OU processes are dominated by their flattest dimensions, thus resembling random walk (Antognini & Sohl-Dickstein, 2018). Similar random-walk-like dynamics has been observed in diverse context: SGD-trained neural network weights, evolution of Influenza virus (H3N2) and ES-optimized visual stimuli (Moore et al., 2018; Wang & Ponce, 2022a;b), suggesting high degeneracy and flat subspaces are generic features of high-dimensional landscapes.

7 Discussion

Our results reveal that ES and GRPO find geometrically distinct solutions despite equivalent task performance: near-orthogonal update directions and orders-of-magnitude norm differences. The random walk interpretation from Section 6 provides a unifying explanation: on the high-dimensional subspace of task-irrelevant directions, ES accumulates isotropic noise whose squared norm grows linearly with steps, while the task-relevant component remains small but sufficient for learning. These geometric differences imply that task-matched solutions found by different optimizers need not be equivalent in parameter space or in how they alter the model more broadly. More generally, our results suggest that weight-space geometry is itself an important object of study in post-training, because similar downstream accuracy can mask substantial differences in the structure and scale of the underlying update.

Acknowledgments

We gratefully acknowledge support from the Kempner Research Fellowship (B.W.) and the Schwartz Fellowship (X.P.), as well as computing resources provided by the Kempner Institute cluster. We thank Core Francisco Park for valuable feedback on earlier versions of this work. We thank [Intelligence Cubed](#) and its fellowship program for providing funding to conduct the experiments.

References

- Immanuel Abdi, Akshat Gupta, Micah Mok, Alexander Lu, Nicholas Lee, and Gopala Anumanchipalli. Evolutionary strategies lead to catastrophic forgetting in llms. *arXiv preprint arXiv:2601.20861*, 2026.
- Armen Aghajanyan, Sonal Gupta, and Luke Zettlemoyer. Intrinsic dimensionality explains the effectiveness of language model fine-tuning. pp. 7319–7328, 2021.
- Joseph Antognini and Jascha Sohl-Dickstein. Pca of high dimensional random walks with comparison to neural network training. *Advances in Neural Information Processing Systems*, 31, 2018.
- Howard Chen, Noam Razin, Karthik Narasimhan, and Danqi Chen. Retaining by doing: The role of on-policy data in mitigating forgetting. *arXiv preprint arXiv:2510.18874*, 2025.
- Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.
- L. Isserlis. On a formula for the product-moment coefficient of any order of a normal frequency distribution. *Biometrika*, 12(1–2):134–139, 1918. doi: 10.1093/biomet/12.1-2.134.
- Daria Korotyshova, Boris Shaposhnikov, Alexey Malakhov, Alexey Khokhulin, Nikita Surnachev, Kirill Ovcharenko, George Bredis, Alexey Gorbatsovski, Viacheslav Sinii, and Daniil Gavrilov. Essa: Evolutionary strategies for scalable alignment. *arXiv preprint arXiv:2507.04453*, 2025.
- Joel Lehman, Jay Chen, Jeff Clune, and Kenneth O Stanley. Es is more than just a traditional finite-difference approximator. In *Proceedings of the genetic and evolutionary computation conference*, pp. 450–457, 2018.
- Qiyao Liang, Jinyeop Song, Yizhou Liu, Jeff Gore, Ila Fiete, Risto Miikkulainen, and Xin Qiu. The blessing of dimensionality in llm fine-tuning: A variance-curvature perspective. *arXiv preprint arXiv:2602.00170*, 2026.
- James Moore, Hasan Ahmed, and Rustom Antia. High dimensional random walks can appear low dimensional: Application to influenza H3N2 evolution. *Journal of Theoretical Biology*, 447:56–64, 6 2018. ISSN 0022-5193. doi: 10.1016/J.JTBI.2018.03.022. URL <https://www.sciencedirect.com/science/article/pii/S002251931830136X>.
- Xin Qiu, Yulu Gan, Conor F Hayes, Qiyao Liang, Yinggan Xu, Roberto Dailey, Elliot Meyerson, Babak Hodjat, and Risto Miikkulainen. Evolution strategies at scale: Llm fine-tuning beyond reinforcement learning. *arXiv preprint arXiv:2509.24372*, 2025.
- Tim Salimans, Jonathan Ho, Xi Chen, Szymon Sidor, and Ilya Sutskever. Evolution strategies as a scalable alternative to reinforcement learning. *arXiv preprint arXiv:1703.03864*, 2017.
- Bidipta Sarkar, Mattie Fellows, Juan Agustin Duque, Alistair Letcher, Antonio León Villares, Anya Sims, Clarisse Wibault, Dmitry Samsonov, Dylan Cope, Jarek Liesen, et al. Evolution strategies at the hyperscale. *arXiv preprint arXiv:2511.16652*, 2025.
- Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.
- Idan Shenfeld, Jyothish Pari, and Pulkit Agrawal. RL’s razor: Why online reinforcement learning forgets less. *arXiv preprint arXiv:2509.04259*, 2025.
- Binxu Wang and Carlos R Ponce. High-performance evolutionary algorithms for online neuron control. *Proceedings of the genetic and evolutionary computation conference*, pp. 1308–1316, 2022a.

Binxu Wang and Carlos R Ponce. Tuning landscapes of the ventral stream. *Cell reports*, 41(6), 2022b.

G. C. Wick. The evaluation of the collision matrix. *Physical Review*, 80(2):268–272, 1950. doi: 10.1103/PhysRev.80.268.

Contents of the Appendices

A	Author contributions	15
B	Disclosure of LLM Use	15
C	Experimental Configurations	15
C.1	Sequential Training Results (Qwen3-4B-Instruct)	15
C.2	Replication on a Second Model Family (Llama-3.2-3B-Instruct)	15
C.3	Hyperparameter Sweeps	16
C.4	Seed Variance in Incremental KL Divergence	17
C.5	Cumulative KL Divergence from the Base Model	17
C.6	Single Task Training Curves	18
C.7	Selected Training Configurations	20
C.8	Compute Cost (FLOPs) Comparison	20
C.9	Evaluation Protocol	20
D	Theoretical Results	22
D.1	Evolution Strategy on a Flat Landscape	22
D.2	Evolution Strategy on a Linear Landscape	22
D.3	Evolution Strategy on a Quadratic Landscape	22
D.4	Dynamics of ES on Quadratic Landscape	23
D.5	Dynamics of GD on Quadratic Landscape	24
D.6	GD displacement is low-dimensional	24
D.7	ES displacement decomposes into signal and diffusion	24
D.8	ES–GD difference is off-manifold	24
E	Detailed Validation of Theoretical Predictions	26
E.1	Testing Random Walk Scaling of Weight Drift in ES and GRPO	26
E.2	Testing the Cosine-Similarity Prediction	31
F	Detailed derivations of theoretical results	32
F.1	Z-score ES Update Setup	32
F.2	Flat Landscape Analysis	32
F.3	Linear Landscape Analysis	33
F.4	Quadratic landscape analysis	38
F.5	Multi-step ES analysis for quadratic landscape	45
F.6	Gradient ascent on quadratic landscape	48
F.7	Solution geometry: GD vs. ES on quadratic landscape	49

A Author contributions

W.H. designed the initial experiments, performed most of the experiments and analyses, and wrote the first draft of the manuscript. B.W. contributed to the theoretical framework and appendix derivations. X.P. coordinated the study and contributed broadly to the design and framing of the study. All authors contributed to revising and polishing the final manuscript.

B Disclosure of LLM Use

In the study, LLM is used for coding the experiments, analyzing the results, assisting mathematical derivation, and polishing the writing.

C Experimental Configurations

All experiments use Qwen3-4B-Instruct-2507 with full-parameter fine-tuning, seed 42, 52, 62, and a maximum of 1,024 generated tokens. Each task uses a fixed training set of 200 prompts. An additional model of Llama-3.2-3B-Instruct was used to repeat the experiments with one seed, 42.

C.1 Sequential Training Results (Qwen3-4B-Instruct)

Table 5: Qwen3-4B-Instruct: accuracy (% , mean $\pm$ std over 3 seeds) throughout sequential training with ES (iter=100). Bold indicates the task just trained at that stage.

Stage	Countdown	Math	SciK	BoolQ
Base	20.78 $\pm$ 0.20	61.40 $\pm$ 0.35	63.77 $\pm$ 0.32	83.97 $\pm$ 0.23
After Countdown	74.52 $\pm$ 0.70	65.20 $\pm$ 0.72	64.35 $\pm$ 0.80	83.43 $\pm$ 0.95
After Math	75.40 $\pm$ 0.31	71.73 $\pm$ 0.42	69.45 $\pm$ 0.46	84.20 $\pm$ 1.18
After Chemistry	72.67 $\pm$ 1.03	71.40 $\pm$ 1.40	74.85 $\pm$ 1.01	84.03 $\pm$ 2.29
After BoolQ	64.87 $\pm$ 8.78	69.93 $\pm$ 1.01	73.80 $\pm$ 1.59	87.53 $\pm$ 0.60

Table 6: Qwen3-4B-Instruct: accuracy (% , mean $\pm$ std over 3 seeds) throughout sequential training with GRPO. Bold indicates the task just trained at that stage.

Stage	Countdown	Math	SciK	BoolQ
Base	17.22 $\pm$ 2.11	66.20 $\pm$ 4.01	62.77 $\pm$ 0.68	85.20 $\pm$ 1.39
After Countdown	69.17 $\pm$ 4.86	67.27 $\pm$ 6.47	60.58 $\pm$ 4.85	85.52 $\pm$ 1.78
After Math	70.32 $\pm$ 3.49	69.80 $\pm$ 5.89	61.63 $\pm$ 2.03	85.57 $\pm$ 1.69
After Chemistry	70.65 $\pm$ 4.01	71.67 $\pm$ 5.03	64.47 $\pm$ 7.34	82.10 $\pm$ 8.00
After BoolQ	70.50 $\pm$ 3.77	74.27 $\pm$ 5.80	66.08 $\pm$ 7.05	89.80 $\pm$ 1.31

C.2 Replication on a Second Model Family (Llama-3.2-3B-Instruct)

Table 7: Llama-3.2-3B-Instruct: accuracy (%) throughout sequential training with ES (iter=100). Bold indicates the task just trained at that stage.

Stage	Countdown	Math	SciK	BoolQ
Base	2.4	38.2	60.6	73.4
After Countdown	37.4	41.2	60.9	75.8
After Math	34.9	41.8	64.0	71.6
After Chemistry	29.3	41.0	65.9	68.8
After BoolQ	20.3	38.2	65.7	81.9

Table 8: Llama-3.2-3B-Instruct: accuracy (%) throughout sequential training with GRPO. Bold indicates the task just trained at that stage.

Stage	Countdown	Math	SciK	BoolQ
Base	1.8	39.4	61.5	73.4
After Countdown	37.8	42.2	62.6	75.3
After Math	40.8	42.0	62.4	75.9
After Chemistry	40.4	44.2	65.8	72.2
After BoolQ	40.3	46.4	67.0	83.0

C.3 Hyperparameter Sweeps

Evolution Strategies. We conducted a sweep over three configurations maintaining $\alpha = \sigma/2$:

Table 9: ES hyperparameter configurations. Config 3 selected.

Config	σ	α
1	0.0005	0.00025
2	0.001	0.0005
3	0.0015	0.00075

GRPO. We swept learning rates with KL penalty disabled ($\beta_{\text{KL}} = 0$) and linear decay:

Table 10: GRPO learning rate configurations. Config 3 selected.

Config	Learning rate
1	1×10^{-6}
2	3×10^{-6}
3	1×10^{-5}
4	3×10^{-5}

On disabling the KL penalty. Although the original GRPO formulation (Shao et al., 2024) includes a KL-divergence penalty against the reference policy, we set $\beta_{\text{KL}} = 0$, following now-common practice in the field. The widely used TRL implementation similarly defaults to $\beta = 0$ for this reason.²

²https://huggingface.co/docs/trl/main/en/grpo_trainer#computing-the-loss

C.4 Seed Variance in Incremental KL Divergence

Table 11: ES: median [min, max] incremental test KL divergence across 3 seeds (42, 52, 62). Bold indicates the task just trained at that stage.

Eval. task	After Countdown	After Math	After Chemistry	After BoolQ
Countdown	0.200 [0.187, 0.232]	0.162 [0.124, 0.221]	0.075 [0.074, 0.083]	0.086 [0.047, 0.086]
Math	0.062 [0.054, 0.071]	0.080 [0.067, 0.088]	0.041 [0.033, 0.042]	0.043 [0.036, 0.054]
Chemistry	0.137 [0.116, 0.143]	0.313 [0.272, 1.276]	1.100 [0.106, 3.062]	0.103 [0.090, 0.649]
BoolQ	0.135 [0.126, 0.137]	0.170 [0.151, 0.227]	0.096 [0.086, 0.127]	0.102 [0.090, 0.122]

Table 12: GRPO: median [min, max] incremental test KL divergence across 3 seeds (42, 52, 62). Bold indicates the task just trained at that stage.

Eval. task	After Countdown	After Math	After Chemistry	After BoolQ
Countdown	0.381 [0.293, 0.454]	0.008 [0.006, 0.012]	0.012 [0.004, 0.030]	0.017 [0.005, 0.029]
Math	0.050 [0.048, 0.056]	0.009 [0.003, 0.010]	0.010 [0.007, 0.016]	0.006 [0.005, 0.011]
Chemistry	0.012 [0.000, 0.115]	0.010 [0.000, 0.019]	0.117 [0.072, 3.738]	0.056 [0.017, 0.105]
BoolQ	0.051 [0.035, 0.061]	0.010 [0.003, 0.010]	0.038 [0.028, 4.431]	0.239 [0.094, 0.322]

C.5 Cumulative KL Divergence from the Base Model

Table 13: Qwen3-4B-Instruct: cumulative KL divergence from base model, ES (iter=100). Mean $\pm$ std over 3 seeds (42, 52, 62). Bold indicates the task just trained at that stage.

Stage	Countdown	Math	SciK	BoolQ
After Countdown	0.206 $\pm$ 0.023	0.062 $\pm$ 0.008	0.132 $\pm$ 0.014	0.133 $\pm$ 0.006
After Math	0.436 $\pm$ 0.032	0.178 $\pm$ 0.021	0.755 $\pm$ 0.477	0.355 $\pm$ 0.079
After Chemistry	0.536 $\pm$ 0.059	0.250 $\pm$ 0.033	2.541 $\pm$ 2.390	0.545 $\pm$ 0.211
After BoolQ	0.551 $\pm$ 0.112	0.288 $\pm$ 0.028	2.013 $\pm$ 1.781	0.545 $\pm$ 0.158

Table 14: Qwen3-4B-Instruct: cumulative KL divergence from base model, GRPO. Mean $\pm$ std over 3 seeds (42, 52, 62). Bold indicates the task just trained at that stage.

Stage	Countdown	Math	SciK	BoolQ
After Countdown	0.376 $\pm$ 0.081	0.052 $\pm$ 0.004	0.040 $\pm$ 0.066	0.049 $\pm$ 0.013
After Math	0.420 $\pm$ 0.113	0.069 $\pm$ 0.018	0.087 $\pm$ 0.129	0.072 $\pm$ 0.025
After Chemistry	0.435 $\pm$ 0.143	0.093 $\pm$ 0.038	1.297 $\pm$ 2.120	1.904 $\pm$ 3.140
After BoolQ	0.435 $\pm$ 0.115	0.091 $\pm$ 0.024	1.282 $\pm$ 2.062	0.194 $\pm$ 0.063

Table 15: Llama-3.2-3B-Instruct: cumulative KL divergence from base model, ES (iter=100). Single seed. Bold indicates the task just trained at that stage.

Stage	Countdown	Math	SciK	BoolQ
After Countdown	0.069	0.020	0.074	0.236
After Math	0.071	0.038	0.127	0.536
After Chemistry	0.095	0.058	0.396	1.644
After BoolQ	0.170	0.066	1.362	1.788

Table 16: Llama-3.2-3B-Instruct: cumulative KL divergence from base model, GRPO. Single seed. Bold indicates the task just trained at that stage.

Stage	Countdown	Math	SciK	BoolQ
After Countdown	0.072	0.011	0.094	0.312
After Math	0.081	0.012	0.107	0.347
After Chemistry	0.091	0.022	2.306	1.411
After BoolQ	0.099	0.023	2.330	1.717

C.6 Single Task Training Curves

Figure 7 and Figure 8 show train and test reward over the course of single-task training for GRPO and ES using the configurations selected in Tables 9 and 10 for Qwen3-4b-Instruct. Both methods converge on all four tasks without overfitting, confirming that these hyperparameters yield stable learning before we proceed to the sequential setting.

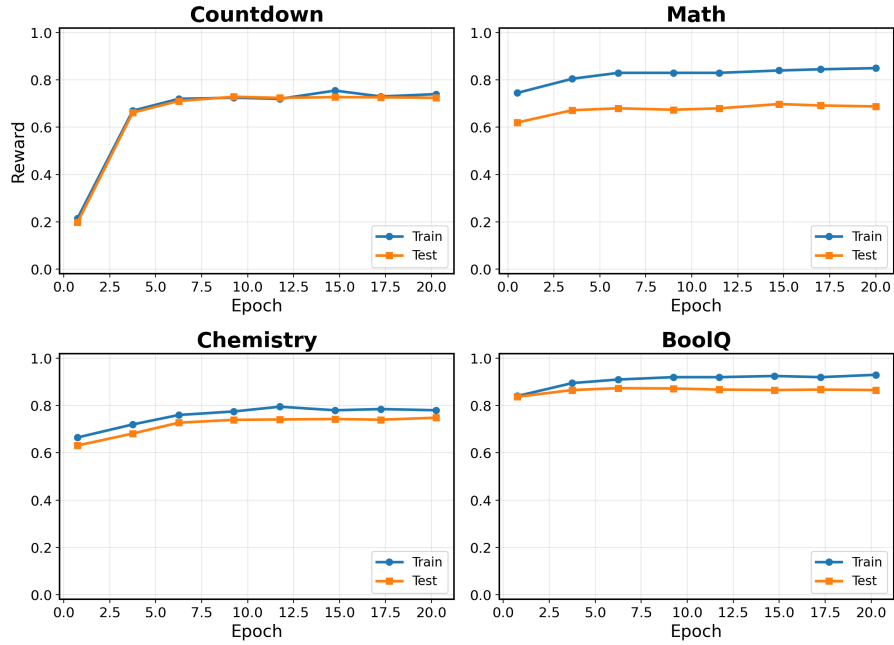
GRPO: Single-Task Training Reward

Figure 7: GRPO single-task training reward across 20 epochs for each task.

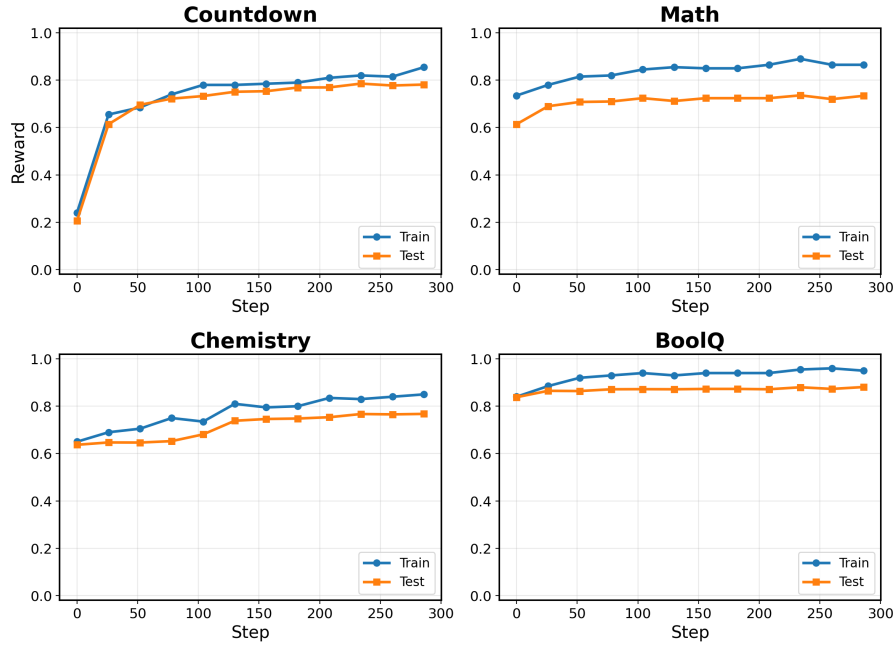
ES: Single-Task Training Reward

Figure 8: ES single-task training reward across 300 steps for each task.

C.7 Selected Training Configurations

The selected configurations are summarized in Table 17. In the single-task setting, ES trains for 300 steps and GRPO for 20 epochs. In the sequential setting, each of the four tasks trains for 100 ES steps or 20 GRPO epochs with identical per-task hyperparameters.

Table 17: Training hyperparameters for ES (left) and GRPO (right).

ES		GRPO	
Noise scale (σ)	0.0015	Learning rate	1×10^{-5}
Step size (α)	0.00075	Optimizer	AdamW
Population size	30	Scheduler	Linear decay
Batch size	200	Max gradient norm	1.0
		KL coefficient (β_{KL})	0
		Rollouts per prompt (K)	8

C.8 Compute Cost (FLOPs) Comparison

For a model with P parameters and sequence length L , we estimate a forward pass at $2PL$ FLOPs and a backward pass at $4PL$.

GRPO. Each response requires a policy forward pass, a reference-model forward pass, and a policy backward pass:

$$\text{FLOPs}_{\text{GRPO}} = T_{\text{GRPO}} \cdot B \cdot G \cdot \underbrace{(2 + 2 + 4)}_{\text{fwd} + \text{ref} + \text{bwd}} \cdot PL = 8 T_{\text{GRPO}} B G PL, \quad (10)$$

where T_{GRPO} is the number of optimizer steps, B the question batch size, and G the number of responses generated per question.

ES. Each population member requires only a forward pass over the evaluation batch:

$$\text{FLOPs}_{\text{ES}} = T_{\text{ES}} \cdot N \cdot D \cdot \underbrace{2}_{\text{fwd}} \cdot PL = 2 T_{\text{ES}} N D PL, \quad (11)$$

where T_{ES} is the number of iterations, N the population size, and D the number of prompts evaluated per iteration.

Applied to our setup. From training logs, GRPO runs $T_{\text{GRPO}} = 80$ steps per task with $B \approx 63$ and $G = 8$; ES uses $N = 30$ and $D = 200$ prompts per iteration, with $T_{\text{ES}} \in \{100, 300\}$. Table 18 reports the resulting FLOP totals:

Table 18: Compute per task, relative to GRPO.

Method	FLOPs	vs. GRPO
GRPO ($T = 80$)	$8(80)(63)(8) PL = 322,560 PL$	$1.00\times$
ES (100 iter)	$2(100)(30)(200) PL = 1,200,000 PL$	$3.72\times$
ES (300 iter)	$2(300)(30)(200) PL = 3,600,000 PL$	$11.16\times$

ES(100), the configuration used throughout Section 4, uses roughly $3.7\times$ the compute of GRPO, and ES(300) roughly $11\times$.

C.9 Evaluation Protocol

All evaluations use greedy decoding (temperature = 0) with a maximum of 1,024 generated tokens and binary rewards (1.0 correct, 0.0 otherwise). Table 19 summarizes

the answer extraction and scoring for each task. Countdown problems are sourced from Jiayi-Pan/Countdown-Tasks-3to4, MATH from nlile/hendrycks-MATH-benchmark, SciKnowEval Chemistry from hicaizju/SciKnowEval v2, BoolQ from google/boolq, MMLU from cais/mmlu, and IFEval from google/IFEval.

Table 19: Evaluation protocol per task. All tasks use greedy decoding (temperature = 0), 1,024 max generated tokens, and binary scoring.

Task	Answer format	$ \mathcal{D}_{\text{test}} $	Answer extraction and scoring
Countdown	<code>\boxed{expr}</code>	2,000	Extract last <code>\boxed{}</code> via regex. Verify the expression contains only digits and <code>+</code> , <code>-</code> , <code>*</code> , <code>/</code> ; each input number is used exactly once; and the evaluated result equals the target (tolerance $< 10^{-5}$).
MATH	<code>\boxed{ans}</code>	500	Extract last <code>\boxed{}</code> with nested-brace handling. Normalize both prediction and gold by stripping LaTeX wrappers (<code>\text</code> , <code>\dfrac</code> , currency/degree symbols). Compare as strings; on mismatch, convert fractions and decimals to floats and compare numerically (tolerance $< 10^{-6}$).
SciKnowEval Chem.	<code>"answer": "X"</code>	2,000	Extract answer letter via regex cascade: (1) JSON-style <code>"answer": "X"</code> , (2) unquoted answer: <code>X</code> , (3) last standalone A/B/C/D token. Use the last match found.
BoolQ	<code>\boxed{yes/no}</code>	2,000	Extract last <code>\boxed{}</code> , normalize to lowercase. Accept <code>{yes, true, 1}</code> as positive and <code>{no, false, 0}</code> as negative.
<i>Holdout tasks (not trained on; used to measure forgetting)</i>			
MMLU	<code>"answer": "X"</code>	14,042	Same regex cascade as SciKnowEval Chemistry.
IFEval	Free-form	541	Check each instruction constraint programmatically (keyword presence, formatting rules, length bounds, etc.). Score 1.0 only if all constraints for a prompt are satisfied (strict prompt-level accuracy).

MMLU and IFEval are evaluated after training to measure whether general capabilities are preserved.

D Theoretical Results

D.1 Evolution Strategy on a Flat Landscape

Proposition 1 (Evolution Strategy on a Flat Landscape). *Let $R(\theta) = \text{const}$, so all reward variation arises from observation noise. Under the ES update $\Delta\theta = \frac{\alpha}{N} \sum_{i=1}^N Z_i \epsilon_i$ with $\epsilon_i \sim \mathcal{N}(0, I_d)$ and z-scored rewards Z_i , the weight update is a pure isotropic random walk:*

$$\Delta\theta \sim \mathcal{N}\left(0, \frac{\alpha^2}{N} I_d\right), \quad \mathbb{E}[\|\Delta\theta\|^2] = \frac{\alpha^2 d}{N}.$$

After T steps of ES with population size N , step size α , and parameter dimension d , the cumulative deviation satisfies

$$\mathbb{E}[\|\theta_T - \theta_0\|^2] = \frac{\alpha^2 T d}{N} = \frac{\sigma^2 T d}{4N},$$

where the last equality substitutes $\alpha = \sigma/2$. In particular, the expected drift grows linearly in T and d , and shrinks inversely in the batch size N . Proof in F.2.

D.2 Evolution Strategy on a Linear Landscape

Proposition 2 (Evolution Strategy on a Linear Landscape). *Let $R(\theta) = -\mathbf{v}^\top \theta$ with observation noise σ_ξ . Define $\sigma_R^2 := \sigma^2 \|\mathbf{v}\|^2 + \sigma_\xi^2$. The mean and covariance of the ES parameter update are*

$$\mathbb{E}[\Delta\theta] = \frac{-\alpha\sigma\mathbf{v}}{\sigma_R} \tag{12}$$

$$\text{Cov}[\Delta\theta] = \frac{\alpha^2}{N} I_d + \frac{\alpha^2 \sigma^2}{N \sigma_R^2} \mathbf{v} \mathbf{v}^\top \tag{13}$$

The mean update is purely aligned with the gradient direction $\mathbf{v}$, and the covariance consists of the same isotropic baseline as the flat landscape plus a rank-1 boost along $\mathbf{v}$. Relative to the quadratic case (Proposition 3), the linear landscape produces identical expressions with $Q^2 = 0$: the curvature-dependent inflation of σ_R^2 and the anisotropic noise term $2\sigma^4 Q^2$ are both absent. Proof in F.3.

Remark 2 (On-manifold alignment of ES update). *Define the signal fraction $s := \sigma^2 \|\mathbf{v}\|^2 / \sigma_R^2 \in [0, 1]$. The on-manifold fraction of total squared displacement is*

$$\rho := \frac{\mathbb{E}[\|P^\parallel \Delta\theta\|^2]}{\mathbb{E}[\|\Delta\theta\|^2]} = \frac{1 + (N+1)s}{d + (N+1)s}.$$

Hence: (i) without signal ($s = 0$), $\rho = 1/d$, i.e. the update is purely diffusive; (ii) in the high-SNR limit ($s \rightarrow 1$), $\rho \rightarrow (N+2)/(d+N+1)$; (iii) ρ increases monotonically in s and N , and decreases in d . Since $N \ll d$ in practice, on-manifold movement per step is negligible relative to off-manifold diffusion.

D.3 Evolution Strategy on a Quadratic Landscape

Proposition 3 (Evolution Strategy on a Quadratic Landscape). *Let $R(\theta) = -\frac{1}{2}\theta^\top Q\theta$, current iterate θ_0 , and $\mathbf{v} := Q\theta_0$ (the negative gradient). Define*

$$\sigma_R^2 := \sigma^2 \|\mathbf{v}\|^2 + \frac{\sigma^4}{2} \text{Tr}[Q^2] + \sigma_\xi^2. \tag{14}$$

The mean and covariance of the ES parameter update are

$$\mathbb{E}[\Delta\theta] = \frac{-\alpha\sigma\mathbf{v}}{\sigma_R}, \tag{15}$$

$$\text{Cov}[\Delta\theta] = \frac{\alpha^2}{N \sigma_R^2} \left(\sigma^2 \mathbf{v} \mathbf{v}^\top + 2\sigma^4 Q^2 + \sigma_R^2 I_d \right). \tag{16}$$

Relative to the linear case (Proposition 2), the quadratic term Q introduces two additional contributions: (i) a curvature-dependent inflation of σ_R^2 through $\text{Tr}[Q^2]$, which attenuates the mean gradient step; and (ii) an anisotropic noise term $2\sigma^4 Q^2 / \sigma_R^2$ in the covariance, which increases update variance along the principal high curvature directions of Q . Proof in F.4.

Remark 3 (On-manifold alignment on quadratic landscape). The on-manifold fraction of total expected squared displacement is

$$\rho = \frac{(N+1)\sigma^2\|\mathbf{v}\|^2 + 2\sigma^4\hat{\mathbf{v}}^\top Q^2\hat{\mathbf{v}} + \sigma_R^2}{(N+1)\sigma^2\|\mathbf{v}\|^2 + 2\sigma^4\text{Tr}[Q^2] + d\sigma_R^2},$$

where $\hat{\mathbf{v}} := \mathbf{v}/\|\mathbf{v}\|$. Setting $Q^2 = 0$ and $\sigma_R^2 = \sigma^2\|\mathbf{v}\|^2 + \sigma_\xi^2$ recovers the linear-landscape ratio $\rho = (1 + (N+1)s)/(d + (N+1)s)$. The quadratic landscape modifies ρ in two ways: the curvature inflates σ_R^2 , which grows the isotropic term $d\sigma_R^2$ in the denominator; and the $2\sigma^4 Q^2$ noise splits between numerator (the projection $\hat{\mathbf{v}}^\top Q^2\hat{\mathbf{v}}$) and denominator (the full trace $\text{Tr}[Q^2]$). When the gradient aligns with the high-curvature eigenvectors, much of the curvature-induced noise is on-manifold and ρ is comparable to the linear case; when the curvature is predominantly orthogonal to $\hat{\mathbf{v}}$, the denominator grows while the numerator does not, suppressing ρ .

D.4 Dynamics of ES on Quadratic Landscape

Proposition 4 (Dynamics of ES on Quadratic Landscape). Consider the simplified evolution

$$\theta_{t+1} = H\theta_t + \eta_t, \quad \eta_t \sim \mathcal{N}\left(0, \frac{\alpha^2}{N}I_d\right),$$

where $H = I - \frac{\alpha\sigma}{\sigma_R}Q = \sum_k \left(1 - \frac{\alpha\sigma}{\sigma_R}\lambda_k\right) \mathbf{u}_k \mathbf{u}_k^\top$.³ Let $\gamma_k := 1 - \frac{\alpha\sigma}{\sigma_R}\lambda_k$ denote the contraction factor along eigendirection $\mathbf{u}_k$. Then the projected mean and variance evolve as

$$\begin{aligned} \mathbb{E}[\mathbf{u}_k^\top \theta_t] &= \gamma_k^t \mathbf{u}_k^\top \theta_0, \\ \text{Var}[\mathbf{u}_k^\top \theta_t] &= \frac{\alpha^2}{N} \cdot \begin{cases} \frac{1 - \gamma_k^{2t}}{1 - \gamma_k^2}, & \gamma_k^2 \neq 1, \\ t, & \gamma_k^2 = 1. \end{cases} \end{aligned}$$

The process converges in mean and variance along direction $\mathbf{u}_k$ if and only if $|\gamma_k| < 1$, i.e.,

$$0 < \lambda_k < \frac{2\sigma_R}{\alpha\sigma}.$$

In this regime, the asymptotic variance is

$$\lim_{t \rightarrow \infty} \text{Var}[\mathbf{u}_k^\top \theta_t] = \frac{\alpha^2}{N} \cdot \frac{1}{1 - \gamma_k^2},$$

and the convergence timescale is $\tau_k = -\log 2 / \log \gamma_k^2$. Proof in F.5.

Remark 4 (Stability conditions). The three failure modes each correspond to a violation of $|\gamma_k| < 1$. Flat directions ($\lambda_k = 0$) satisfy $\gamma_k = 1$, so both the mean and variance grow as a pure random walk: the optimizer drifts diffusively and never localizes. Concave directions ($\lambda_k < 0$) give $\gamma_k > 1$, causing exponential divergence of both mean and variance — the landscape actively repels the iterates. Overly sharp convex directions ($\lambda_k \geq 2\sigma_R/\alpha\sigma$) give $|\gamma_k| \geq 1$ for a different reason: the ES step overshoots, analogous to gradient descent with too large a learning rate. Note that because the ES learning rate is $\alpha = \sigma/2$, this threshold becomes $\lambda_k = 4\sigma_R/\sigma^2$.

Remark 5 (Optimal curvature). Within the stable regime, the convergence speed and asymptotic variance are jointly controlled by γ_k^2 . Directions with $\lambda_k \approx \sigma_R/\alpha\sigma$ achieve $\gamma_k \approx 0$, which simultaneously minimizes the convergence timescale ($\tau_k \rightarrow 1$ iteration) and the asymptotic variance ($\approx \alpha^2/N$). Directions near the stability boundary $\lambda_k \rightarrow 0^+$ or $\lambda_k \rightarrow (2\sigma_R/\alpha\sigma)^-$ suffer slow convergence and inflated variance. Thus the landscape curvature relative to the ratio $\sigma_R/\alpha\sigma$ plays the role of an effective condition number for the ES dynamics.

³ σ_R here is a fixed constant, not changing across iterations as in the full ES update (Eq. 14).

Remark 6 (Role of noise scale). The asymptotic variance $\frac{\alpha^2}{N(1-\gamma_k^2)}$ reveals the expected tension between exploration and precision: increasing the population size N suppresses variance linearly, while the landscape geometry (through γ_k^2) sets a floor on the minimum achievable uncertainty. In well-conditioned directions, $1 - \gamma_k^2 \approx 2\frac{\alpha\sigma}{\sigma_R}\lambda_k$, so the asymptotic variance scales as $\frac{\alpha\sigma_R}{2N\sigma\lambda_k}$, inversely proportional to both curvature and the signal-to-noise ratio σ/σ_R .

D.5 Dynamics of GD on Quadratic Landscape

Proposition 5 (Dynamics of GD on Quadratic Landscape). Let $R(\theta) = -\frac{1}{2}\theta^\top Q\theta$ with $Q \succeq 0$, and let gradient descent with learning rate β iterate as $\theta_{t+1} = (I - \beta Q)\theta_t$. Define $H_{\text{GD}} := I - \beta Q$ and let $\gamma_k^{\text{GD}} := 1 - \beta\lambda_k$. The iterate is

$$\theta_t = H_{\text{GD}}^t \theta_0 = \sum_{k=1}^d (\gamma_k^{\text{GD}})^t (\mathbf{u}_k^\top \theta_0) \mathbf{u}_k.$$

Direction $\mathbf{u}_k$ converges if and only if $|\gamma_k^{\text{GD}}| < 1$, i.e., $0 < \lambda_k < 2/\beta$. In this regime the convergence timescale is $\tau_k^{\text{GD}} = -\log 2 / \log(\gamma_k^{\text{GD}})^2$, and the residual error is exactly zero: $\lim_{t \rightarrow \infty} \mathbf{u}_k^\top \theta_t = 0$. Proof in F.6.

Remark 7 (Stability conditions for GD). The three failure modes mirror those of the ES dynamics (Proposition 4): flat directions ($\lambda_k = 0$) give $\gamma_k^{\text{GD}} = 1$, so the iterate is frozen at $\mathbf{u}_k^\top \theta_0$ for all t — unlike ES, where the iterate drifts diffusively. Concave directions ($\lambda_k < 0$) give $\gamma_k^{\text{GD}} > 1$, causing exponential divergence. Overly sharp convex directions ($\lambda_k \geq 2/\beta$) give $|\gamma_k^{\text{GD}}| \geq 1$, and the iterate overshoots with growing or non-decaying oscillations.

D.6 GD displacement is low-dimensional

Proposition 6 (GD displacement is low-dimensional). Starting from θ_0 on a rank- r quadratic landscape ($r \ll d$), the GD displacement

$$\theta_{\text{GD}} - \theta_0 = \sum_{k: \lambda_k > 0} [(1 - \beta\lambda_k)^T - 1] (\mathbf{u}_k^\top \theta_0) \mathbf{u}_k$$

lies entirely within the r -dimensional column space of Q , with $\|\theta_{\text{GD}} - \theta_0\|^2 = \mathcal{O}(r)$.

D.7 ES displacement decomposes into signal and diffusion

Proposition 7 (ES displacement decomposes into signal and diffusion). The ES displacement from initialization decomposes as

$$\theta_{\text{ES}} - \theta_0 = \underbrace{(H_{\text{ES}}^T - I)\theta_0}_{\text{signal (on-manifold)}} + \underbrace{\sum_{t=1}^T H_{\text{ES}}^{t-1} \eta_{T-t}}_{\text{noise (all directions)}}.$$

The signal term lies within the column space of Q , identical in structure to the GD displacement with $\beta \leftrightarrow \alpha\sigma/\sigma_R$. The noise term has support in all d dimensions. For $r \ll d$ and large T ,

$$\mathbb{E}\|\theta_{\text{ES}} - \theta_0\|^2 = \underbrace{\mathcal{O}(r)}_{\text{on-manifold}} + \underbrace{\frac{\alpha^2 T(d-r)}{N}}_{\text{off-manifold diffusion}} \approx \frac{\alpha^2 Td}{N}.$$

D.8 ES–GD difference is off-manifold

Proposition 8 (ES–GD difference is off-manifold). When both methods have converged on the active directions ($(\gamma_k^{\text{ES}})^T \approx (\gamma_k^{\text{GD}})^T \approx 0$), the expected squared difference between the two solutions is dominated by off-manifold diffusion:

$$\mathbb{E}\|\theta_{\text{ES}} - \theta_{\text{GD}}\|^2 \approx \frac{\alpha^2 T(d-r)}{N} = \mathcal{O}(d).$$

Remark 8 (Geometric summary). *The three pairwise squared distances satisfy the hierarchy (for $r \ll d$, large T):*

$$\begin{aligned}\|\theta_{\text{GD}} - \theta_0\|^2 &= \mathcal{O}(r), \\ \mathbb{E}\|\theta_{\text{ES}} - \theta_0\|^2 &= \mathcal{O}(r) + \mathcal{O}(d), \\ \mathbb{E}\|\theta_{\text{ES}} - \theta_{\text{GD}}\|^2 &= \mathcal{O}(d).\end{aligned}$$

The GD displacement is low-dimensional and task-aligned. The ES displacement superposes a comparable on-manifold component with a much larger isotropic off-manifold diffusion. The ES–GD difference is dominated by the off-manifold term alone, since both methods converge to the same optimum within the active subspace. This accounts for the empirical observation that ES achieves similar task performance to GD (shared on-manifold signal) while being far from it in parameter space (off-manifold diffusion).

Remark 9 (Cosine similarity between displacements). *Since $\theta_{\text{GD}} - \theta_0$ lies within the column space of Q and the ES noise is zero-mean, the expected inner product picks up only the on-manifold signal of ES. For large T with both converged on active directions, the expected cosine similarity satisfies*

$$\mathbb{E} [\cos \angle(\theta_{\text{ES}} - \theta_0, \theta_{\text{GD}} - \theta_0)] \sim \mathcal{O}\left(\sqrt{r/d}\right),$$

which is small for $r \ll d$: the two displacement vectors are nearly orthogonal despite leading to similar task performance.

E Detailed Validation of Theoretical Predictions

E.1 Testing Random Walk Scaling of Weight Drift in ES and GRPO

Proposition 1 predicts that, on a flat landscape, the squared norm of parameter drift grows linearly in the step index t :

$$\mathbb{E}[\|\Delta\theta_t\|^2] = \frac{\alpha^2 t d}{N}.$$

This is a parameter-free prediction: d is the number of parameters and α, N are ES hyperparameters, so the predicted trajectory is fixed before any data are seen. It is also the null model one would obtain if the landscape were entirely flat and every parameter irrelevant to the task; deviations from it are therefore informative about which parts of the network the task actually constrains.

The proposition applies to any set of coordinates evolving under the same update, so we test it at two granularities: for the parameter vector as a whole, and independently for each named parameter tensor.

Setup We fine-tune two language models of different families and scales on a single reasoning task:

- **Qwen3-4B-Instruct-2507**, $d = 4,022,468,096$ parameters, 36 transformer blocks, $d_{\text{model}} = 2560$;
- **Llama-3.2-3B**, $d = 3,212,749,824$ parameters, 28 transformer blocks, $d_{\text{model}} = 3072$.

Each is trained with two optimizers:

- **Evolution Strategy (ES)**: z-scored ES with perturbation standard deviation $\sigma = 0.0015$, step size $\alpha = \sigma/2 = 7.5 \times 10^{-4}$, and population size $N = 30$, run for $T = 500$ steps. The run was extended from 300 to 500 steps for the analysis reported here.
- **GRPO**: Group Relative Policy Optimization, a gradient-based RL fine-tuning method, run for $T = 79$ steps.

At each step t we record, for every named parameter $\theta^{(k)}$,

$$\|\Delta\theta_t^{(k)}\|^2 := \|\theta_t^{(k)} - \theta_0^{(k)}\|^2,$$

the squared Euclidean distance from initialization.

Whole-model drift vs. theory (Figure 9) Figure 9 plots the empirical $\sum_k \|\Delta\theta_t^{(k)}\|^2$ against the predicted curve. ES tracks it closely in both models, with $R^2 = 0.9943$ (Qwen) and 0.9978 (Llama) measured against the parameter-free prediction itself rather than against a fitted line.

The residual is a small error in the *slope*, not a departure from linearity. Refitting the slope freely — still through the origin, as the law has no intercept — raises R^2 to 0.9999 and 1.0000 respectively, so the growth is linear to four or five decimal places. Writing $\rho := d_{\text{eff}}/d$ for the ratio of the fitted to the predicted slope:

Model	d	predicted slope	fitted slope	ρ	R^2 (free fit)
Qwen	4.02×10^9	75.42	72.74	0.964	0.9999
Llama	3.21×10^9	60.24	61.66	1.024	1.0000

The two models bracket the prediction: Qwen’s aggregate drift runs 3.6% below it and Llama’s 2.4% above. Both mechanisms available in the theory are consistent with a deviation of this size — curvature along task-relevant directions suppresses diffusion and drives ρ below unity (Prop. 4), while a systematic gradient-like component adds displacement and

drives it above (Prop. 3) — so we do not read a single mechanism into the aggregate scaling residual. What the aggregate does establish is that neither effect is large: across two architectures the total drift rate is within 4% of a prediction with no fitted parameters.

Note that these aggregate values are exactly the parameter-count-weighted mean of the per-tensor ρ reported below, agreeing to four decimal places.

GRPO behaves qualitatively differently. Its drift saturates within roughly 50 steps rather than growing linearly, reaching a peak $\|\Delta\theta\|^2$ of 1.12 (Qwen) and 1.72 (Llama). Compared at the last step both optimizers reached, $T = 79$, ES has already moved the weights 5.2×10^3 (Qwen) and 2.9×10^3 (Llama) times farther. This is consistent with gradient-based fine-tuning making targeted updates within a low-dimensional subspace rather than diffusing through parameter space.

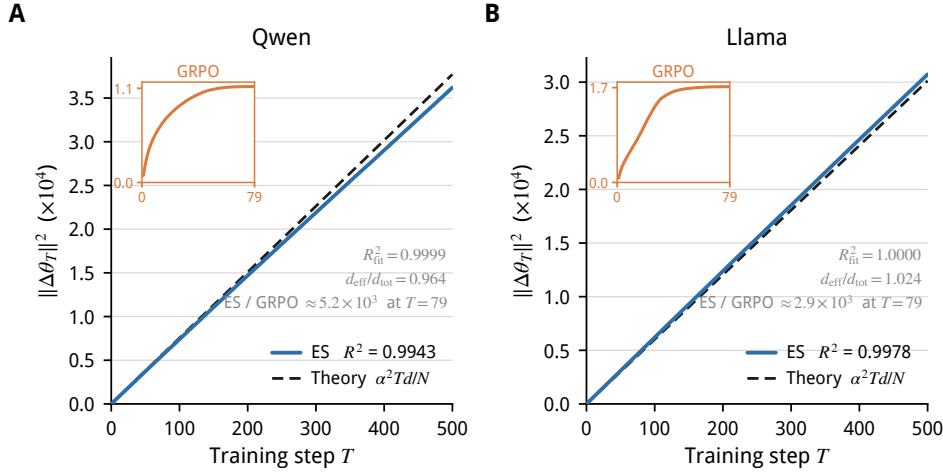


Figure 9: **Total weight drift under ES follows the random-walk law; under GRPO it does not.** Squared displacement of the full parameter vector $\|\Delta\theta_T\|^2$ against training step T for Qwen (A, $d = 4.02 \times 10^9$) and Llama (B, $d = 3.21 \times 10^9$). Solid blue, ES; dashed black, the prediction $\alpha^2 T d / N$ with $\alpha = \sigma / 2$, $\sigma = 0.0015$ and $N = 30$. The R^2 in the legend is measured against that prediction, which has no fitted parameter; the annotation gives instead the R^2 of a freely fitted slope and the effective dimension $\rho = d_{\text{eff}} / d_{\text{total}}$ that slope implies. Insets show GRPO on its own axes over its full 79 steps, annotated with the ES/GRPO displacement ratio at the matched step $T = 79$. Values and interpretation in the text.

Per-parameter effective dimension To characterize which parameters drive the drift, we fit a no-intercept linear regression

$$\|\Delta\theta_t^{(k)}\|^2 \approx s^{(k)} \cdot t$$

for each parameter k independently, obtaining a slope $s^{(k)}$ and a goodness of fit R_k^2 . Because the fit has no intercept, R_k^2 is not bounded below by zero: a tensor whose drift is poorly described by *any* line through the origin will receive a negative value, and several families below do. Inverting Proposition 1 gives the effective random walk dimension

$$d_{\text{eff}}^{(k)} := \frac{s^{(k)} N}{\alpha^2} = \frac{4 s^{(k)} N}{\sigma^2}, \quad \rho^{(k)} := \frac{d_{\text{eff}}^{(k)}}{d^{(k)}},$$

where $d^{(k)}$ is the number of elements in parameter k . A value $\rho^{(k)} \approx 1$ indicates the parameter diffuses isotropically through its full space as predicted; $\rho^{(k)} \ll 1$ indicates constrained diffusion; $\rho^{(k)} > 1$ indicates excess displacement relative to dimension, potentially from a gradient-like drift component (Prop. 3).

One tensor is excluded throughout. Qwen’s `model.layers.34.input_layernorm.weight` has exactly zero displacement at every recorded step, leaving $s^{(k)}$, $\rho^{(k)}$ and R_k^2 undefined; it is dropped from all statistics and figures ($n = 35$ rather than 36 for that family).

Large weight matrices The matrix-shaped parameters — embedding, `qkv.proj`, `o.proj`, `gate_up.proj`, `down.proj`; 145 tensors in Qwen and 113 in Llama — behave as near-isotropic random walkers in both models:

Model	n	ρ (mean $\pm$ SD)	R^2 (mean $\pm$ SD)
Qwen	145	0.968 ± 0.014	0.999888 ± 0.000014
Llama	113	1.024 ± 0.031	0.999968 ± 0.000022

Every such tensor in either model lies within 10% of $\rho = 1$ and above $R^2 = 0.9998$, directly confirming the flat-landscape regime of Proposition 1 for the bulk of the parameter count.

The small residual is nonetheless systematic rather than noisy, and it is what distinguishes the two models: 75% of Llama’s large tensors exceed $\rho = 1$, reaching 1.07 for `o.proj`, whereas every one of Qwen’s 145 falls below it, with a maximum of 0.999. Llama also fits the law about four times more precisely (median $R^2 = 0.999975$ against 0.999890). We report this as an observation; the figures do not isolate its cause.

Norm and layer-norm parameters The remaining parameters are RMSNorm gain vectors (144 tensors in Qwen, 57 in Llama). Both models place one before the attention sublayer of each block (`input_layernorm`) and one before the MLP sublayer — the latter named `post_attention_layernorm` for its position in the block rather than because it normalizes the attention output — plus a single output norm before the language-model head. Each is a per-channel gain of length d_{model} , with no bias and no centering. Qwen adds QK-norms, applied within each attention head to queries and keys; their gain is of length $d_{\text{head}} = 128$ and is shared across all heads, so these are the smallest tensors in either model by three to five orders of magnitude.

These parameters depart from the law substantially, and — unlike the matrices — they depart differently in the two architectures:

Family	Model	n	ρ (mean $\pm$ SD)	R^2 (mean $\pm$ SD)
<code>input_layernorm</code>	Qwen	35	1.00 ± 0.93	-0.38 ± 1.31
	Llama	28	0.37 ± 0.15	-2.69 ± 1.22
<code>post_attention_layernorm</code>	Qwen	36	1.55 ± 0.79	-0.98 ± 1.24
	Llama	28	0.34 ± 0.25	-3.11 ± 1.10
<code>q.norm</code>	Qwen	36	2.46 ± 0.33	0.81 ± 0.12
<code>k.norm</code>	Qwen	36	2.00 ± 0.50	0.70 ± 0.23
<code>model.norm</code>	Qwen	1	0.10	0.83
	Llama	1	2.99	0.89

Three observations. First, the strongly negative R^2 values for the per-block LayerNorms are not scatter around a line but evidence of the wrong functional form: Figure 11 shows the underlying trajectories, which rise steeply and then saturate. Llama’s pass 90% of their final displacement by $T \approx 130$, against $T = 450$ for a straight line, and end well below the predicted trajectory. A saturating curve, not a random walk, is the appropriate description — which is consistent with the theory describing an OU process under quadratic potential (Eq. 130).

Second, the sign of the deviation is architecture-dependent. Llama’s per-block LayerNorms diffuse in roughly a third of their nominal dimensions ($\rho \approx 0.35$), while Qwen’s sit at or above unity ($\rho = 1.00$ and 1.55), with correspondingly less negative R^2 . The single output norm flips outright between models, $\rho = 2.99$ in Llama against 0.10 in Qwen.

Third, Qwen’s QK-norms show a consistent excess, $\rho \approx 2.0\text{--}2.5$ with $R^2 \approx 0.7\text{--}0.8$. At $d = 128$ entries they are also where the concentration underlying $\alpha^2 T d_p / N$ is weakest. Their excess drift may reflect a genuinely higher signal-to-noise ratio in these low-dimensional subspaces, but that caveat should be kept in view.

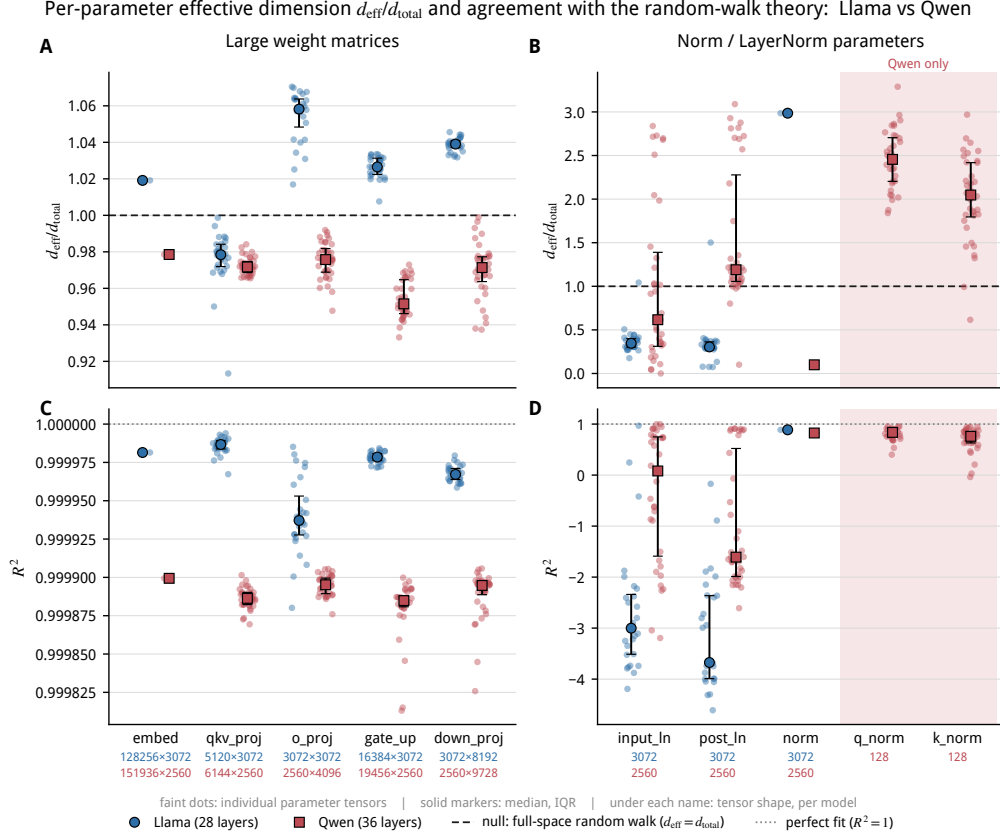


Figure 10: Per-parameter effective dimension and agreement with the random-walk theory, Llama vs Qwen. For every parameter tensor we fit $\|\Delta\theta_T\|^2 = \alpha^2 T d_{\text{eff}} / N$ through the origin over 500 ES steps and report $\rho = d_{\text{eff}} / d_{\text{total}}$ (**A**, **B**) and the R^2 of that fit (**C**, **D**), for the large weight matrices (left) and the RMSNorm gain vectors (right). Faint dots are individual tensors — 28 per per-layer family in Llama, 36 in Qwen — and solid markers give the median with the interquartile range. Two reference levels are drawn, distinguished by colour: the dashed black line is the null model in which a tensor random-walks through its full parameter space, $\rho = 1$; the dotted grey line is a perfect fit, $R^2 = 1$. Under each name is the tensor’s shape in each model (blue, Llama; red, Qwen); shading marks the QK-norm families, present only in Qwen. The fit has no intercept, so R^2 is free to be negative, as it is throughout much of **D**; the axis in **C** is correspondingly expanded, spanning only the top 2×10^{-4} of the R^2 range. The large matrices cluster tightly at $\rho \approx 1$ with R^2 indistinguishable from unity at ordinary resolution, while the norm parameters scatter across the full range of both quantities; the weight-growth curves underlying the latter are shown in Figure 11. Qwen’s layers. 34. input_layernorm is excluded throughout ($n = 35$ for that family; see text).

Summary The random-walk law describes ES weight drift accurately where the theory’s assumptions are best met and fails in an interpretable way where they are not. At the whole-model level it predicts total displacement across two architectures to within 4% with no fitted parameters. At the tensor level it holds to within 10% in ρ and 2×10^{-4} in R^2 for every large weight matrix — which is to say, for essentially the entire parameter count. It

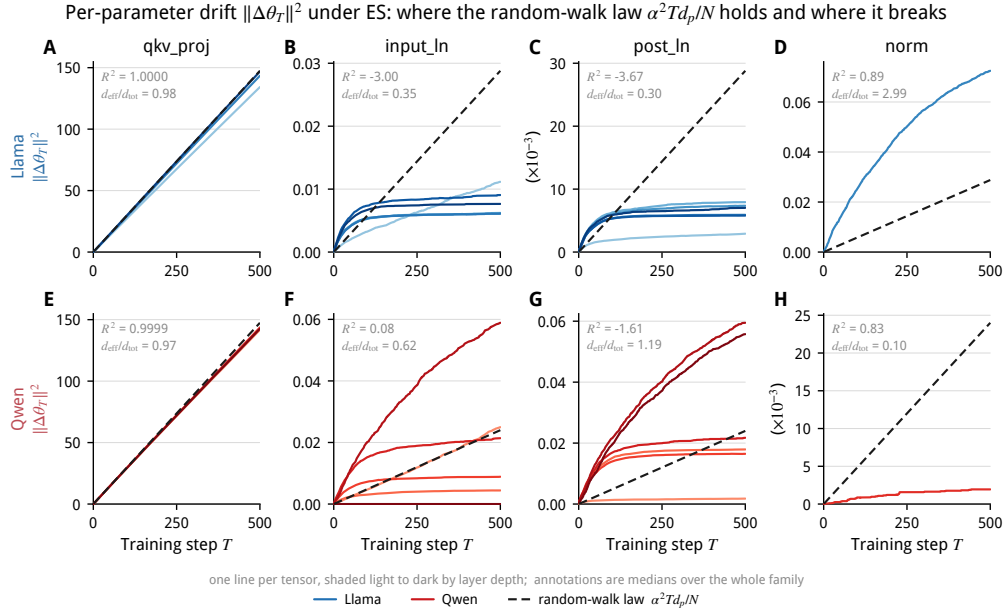


Figure 11: **Per-parameter weight drift under ES: where the random-walk law holds and where it breaks.** Squared displacement $\|\Delta\theta_T\|^2$ against ES step T for individual parameter tensors of Llama (A–D, blue) and Qwen (E–H, red), one parameter family per column. Dashed lines give the random-walk prediction $\alpha^2 T d_p / N$, where d_p is the number of entries in the tensor ($\alpha = \sigma/2$, $\sigma = 0.0015$, $N = 30$); it is common to every curve in a panel, since all tensors in a family share a shape. Curves are individual tensors shaded light to dark by layer depth — three evenly spaced layers for qkv_proj, six for input_ln and post_ln, and the single final norm for norm. Annotations give the median R^2 and median $\rho = d_{\text{eff}} / d_{\text{total}}$ over the *entire* family, not only the curves drawn; R^2 is that of a fit through the origin, as the law has no intercept, and may therefore be negative. The large weight matrices (A, E) lie on the predicted line for the whole run, while the RMSNorm gains rise steeply and then saturate — the departure that the negative R^2 of a through-origin fit records. Values in the text.

fails for the RMSNorm gain vectors, which are three to five orders of magnitude smaller, and the failed fit is due to weight change saturating: those coordinates stop moving, which is what one expects when a quadratic-like landscape constrains the random walk (Sec. F.5).

E.2 Testing the Cosine-Similarity Prediction

Remark 9 predicts that the expected cosine similarity between the ES and GRPO update directions scales as $\mathbb{E}[\cos \angle(\theta_{\text{ES}} - \theta_0, \theta_{\text{GRPO}} - \theta_0)] \sim \mathcal{O}(\sqrt{r/d})$, where r is the task-relevant (GD-aligned) rank and d is the full parameter dimension. This prediction was previously verified only indirectly, through its consequences for update norms and mode connectivity. Here we report the direct empirical measurement.

For each sequential checkpoint, we compute the cosine similarity between the full parameter-space update vectors $\Delta\theta_{\text{ES}} = \theta_{\text{ES}} - \theta_{\text{base}}$ and $\Delta\theta_{\text{GRPO}} = \theta_{\text{GRPO}} - \theta_{\text{base}}$, using float64 key-wise accumulation over the full parameter vector ($d \approx 3\text{--}4 \times 10^9$ for Llama-3.2-3B and Qwen3-4B respectively). Table 20 reports the results for both model families.

Table 20: Empirical cosine similarity between ES and GRPO update directions at each sequential checkpoint.

Stage	Qwen3-4B cos	Llama-3.2-3B cos
T0 Countdown	1.0×10^{-4}	0.8×10^{-4}
T1 Math	1.6×10^{-4}	1.0×10^{-4}
T2 Chemistry	2.2×10^{-4}	1.0×10^{-4}
T3 BoolQ	2.7×10^{-4}	1.3×10^{-4}

All measured cosines are $\sim 10^{-4}$, corresponding to almost orthogonal updates. We noticed that, when plugging a curvature rank estimate $r \approx 100$ with parameter dimension $d \approx 3\text{--}4 \times 10^9$ into Remark 9, the predicted scale of the cosine $\sqrt{r/d} \approx 1.6 \times 10^{-4}$ will match the scale of the observed cosine values. A more principled way to estimate the rank r of the curvature of the landscape is left for future work.

F Detailed derivations of theoretical results

F.1 Z-score ES Update Setup

We detail the ES update rule as implemented in LLM fine-tuning. The update at step t is

$$\theta_{t+1} = \theta_t + \alpha \cdot \frac{1}{N} \sum_{i=1}^N Z_i \epsilon_i \quad (17)$$

where each $\epsilon_i \sim \mathcal{N}(0, I_d)$ is an independent Gaussian perturbation, and $Z_i = \frac{R_i - \mu_R}{\sigma_R}$ is the z-scored return for the i -th sample. Here, R_i is the reward, with μ_R and σ_R denoting the mean and standard deviation of the R_i across the population. With slight abuse of notation we let $R(\theta_t + \sigma \epsilon_i)$ be the true or expected reward at the perturbed location, and R_i the observed reward with certain observation noise. The step size is set to $\alpha = \sigma/2$, and a population size of $N = 30$ is typical. Defining the update increment as

$$\Delta\theta = \alpha \cdot \frac{1}{N} \sum_{i=1}^N Z_i \epsilon_i \quad (18)$$

the algorithm iterates as $\theta_{t+1} = \theta_t + \Delta\theta$.

F.2 Flat Landscape Analysis

Proof of Proposition 1. The key assumption is that Z_i is independent of ϵ_i , i.e., $Z_i \perp \epsilon_i$. Then, conditioned on a set of Z_i ,

$$\text{Var}[\Delta\theta | Z_i] = \left(\frac{\alpha^2}{N^2} \sum_{i=1}^N Z_i^2 \right) I_d \quad (19)$$

and the distribution is isotropic Gaussian:

$$\Delta\theta | Z_i \sim \mathcal{N} \left(0, \left(\frac{\alpha^2}{N^2} \sum_{i=1}^N Z_i^2 \right) I_d \right) \quad (20)$$

By definition,⁴

$$\begin{aligned} \sum_{i=1}^N Z_i^2 &= \sum_{i=1}^N \left(\frac{R_i - \mu_R}{\sigma_R} \right)^2 \\ &= \frac{1}{\sigma_R^2} \sum_{i=1}^N (R_i - \mu_R)^2 \\ &= N \quad (\text{or } N-1 \text{ with unbiased estimator}) \end{aligned} \quad (21)$$

Therefore, in general the scaling is:

$$\begin{aligned} \text{Var}[\Delta\theta | Z_i] &= \frac{\alpha^2}{N^2} N I_d \\ &= \frac{\alpha^2}{N} I_d \end{aligned} \quad (22)$$

Since this is independent of Z_i , it follows after marginalizing:

$$\Delta\theta \sim \mathcal{N} \left(0, \frac{\alpha^2}{N} I_d \right) \quad (23)$$

⁴Note that this constraint depends on the implementation of the z-score $Z_i = \frac{R_i - \mu_R}{\sigma_R}$. In our code, we compute the standard deviation as $\sigma_R^2 = \frac{1}{N} \sum_{i=1}^N (R_i - \mu_R)^2$. In the unbiased variance estimator (denominator $N-1$), the identity reads $\sum_{i=1}^N Z_i^2 = N-1$.

$$\text{Var}[\Delta\theta] = \frac{\alpha^2}{N} I_d \quad (24)$$

On a flat loss, $\Delta\theta$ is independent between steps, so after T steps, the combined weight change is

$$\bar{\Delta}\theta_T = \theta_T - \theta_0 = \sum_{t=1}^T \Delta\theta_t \quad (25)$$

and the total variance is

$$\text{Var}[\bar{\Delta}\theta_T] = \frac{\alpha^2 T}{N} I_d \quad (26)$$

The expected squared norm of total deviation is

$$\mathbb{E}\|\bar{\Delta}\theta_T\|^2 = \frac{\alpha^2 T d}{N} \quad (27)$$

Finally, substituting $\alpha = \frac{\sigma}{2}$ gives

$$\frac{\alpha^2}{N} = \frac{\sigma^2}{4N} \quad (28)$$

and

$$\mathbb{E}\|\bar{\Delta}\theta_T\|^2 = \frac{\sigma^2 T d}{4N} \quad (29)$$

□

F.3 Linear Landscape Analysis

Proof of Proposition 2. Consider a linear landscape with i.i.d. observational noise $\xi \sim \mathcal{N}(0, 1)$, assumed independent of ϵ , i.e., $\xi \perp \epsilon$. The linear reward is

$$R(\theta) = -\mathbf{v}^\top \theta \quad (30)$$

and the observed reward for each sample is

$$\begin{aligned} R_i &= R(\theta_0 + \sigma\epsilon_i) + \sigma\xi_i \\ &= -\mathbf{v}^\top (\theta_0 + \sigma\epsilon_i) + \sigma\xi_i \\ &\equiv -\sigma\mathbf{v}^\top \epsilon_i + \sigma\xi_i \end{aligned} \quad (31)$$

Decompose the exploration as

$$\epsilon_i = \epsilon_i^\perp + \epsilon_i^\parallel \quad (32)$$

where

$$P^\perp := I - \frac{\mathbf{v}\mathbf{v}^\top}{\|\mathbf{v}\|^2} \quad (33)$$

$$P^\parallel := \frac{\mathbf{v}\mathbf{v}^\top}{\|\mathbf{v}\|^2} \quad (34)$$

so that $\epsilon_i^\perp := P^\perp \epsilon_i$, $\epsilon_i^\parallel := P^\parallel \epsilon_i$. Then $\epsilon_i^\perp \sim \mathcal{N}(0, P^\perp)$, $\epsilon_i^\parallel \sim \mathcal{N}(0, P^\parallel)$, and $\epsilon_i^\perp \perp \epsilon_i^\parallel$.

The reward is

$$R_i = -\sigma\mathbf{v}^\top \epsilon_i^\parallel + \sigma\xi_i \quad (35)$$

and thus $R_i \perp \epsilon_i^\perp$.

The update is:

$$\Delta\theta = \alpha \cdot \frac{1}{N} \sum_{i=1}^N Z_i \epsilon_i \quad (36)$$

and can be split as

$$\Delta\theta = P^\perp \Delta\theta + P^\parallel \Delta\theta \quad (37)$$

Off-manifold component

$$\begin{aligned}
P^\perp \Delta \theta &= \alpha \cdot \frac{1}{N} \sum_{i=1}^N Z_i P^\perp \epsilon_i \\
&= \alpha \cdot \frac{1}{N} \sum_{i=1}^N Z_i \epsilon_i^\perp
\end{aligned} \tag{38}$$

Conditioned on the observed normalized reward Z_i , $P^\perp \Delta \theta$ is a weighted sum of Gaussian random vectors, thus Gaussian:

$$P^\perp \Delta \theta \mid \{Z_i\}_{i=1}^N \sim \mathcal{N}\left(0, \frac{\alpha^2}{N^2} \sum_{i=1}^N Z_i^2 P^\perp\right) \tag{39}$$

Since $\{Z_i\}_{i=1}^N$ are z-scored $\{R_i\}_{i=1}^N$, the sum satisfies the following identity.

$$P^\perp \Delta \theta \mid \{Z_i\}_{i=1}^N \sim \mathcal{N}\left(0, \frac{\alpha^2}{N} P^\perp\right) \tag{40}$$

Because of the following equations:

$$\sigma_R^2 = \frac{1}{N} \sum_{i=1}^N (R_i - \mu_R)^2 \tag{41}$$

$$Z_i = \frac{R_i - \mu_R}{\sigma_R} \tag{42}$$

Thus, the conditional distribution is

$$P^\perp \Delta \theta \mid \{Z_i\}_{i=1}^N \sim \mathcal{N}\left(0, \frac{\alpha^2}{N^2} N P^\perp\right) \tag{43}$$

Since the conditional distribution $P^\perp \Delta \theta \mid \{Z_i\}_{i=1}^N$ is independent of the observed reward, marginalizing over Z_i we have

$$P^\perp \Delta \theta \sim \mathcal{N}\left(0, \frac{\alpha^2}{N} P^\perp\right) \tag{44}$$

The norm of the off-manifold component is

$$\begin{aligned}
\mathbb{E} \|P^\perp \Delta \theta\|^2 &= \text{Tr} \left[\frac{\alpha^2}{N} P^\perp \right] \\
&= \frac{\alpha^2}{N} (d - 1)
\end{aligned} \tag{45}$$

On-manifold component

$$\begin{aligned}
P^\parallel \Delta \theta &= \alpha \cdot \frac{1}{N} \sum_{i=1}^N Z_i P^\parallel \epsilon_i \\
&= \alpha \cdot \frac{1}{N} \sum_{i=1}^N Z_i \epsilon_i^\parallel
\end{aligned} \tag{46}$$

The on-manifold variable can be expressed as $\epsilon_i^\parallel = \frac{\mathbf{v}}{\|\mathbf{v}\|} n_i$, with $n_i \sim \mathcal{N}(0, 1)$, thus the reward can be written as

$$\begin{aligned}
R_i &= -\sigma \mathbf{v}^\top \epsilon_i^\parallel + \sigma_\xi \tilde{\zeta}_i \\
&= -\sigma \frac{\mathbf{v}^\top \mathbf{v}}{\|\mathbf{v}\|} n_i + \sigma_\xi \tilde{\zeta}_i \\
&= -\sigma \|\mathbf{v}\| n_i + \sigma_\xi \tilde{\zeta}_i
\end{aligned} \tag{47}$$

Recall the definition of observed reward and z-score,

$$Z_i = \frac{R_i - \mu_R}{\sigma_R} \quad (48)$$

To simplify derivation, we use the population expectation and standard deviation here:

$$\mu_R \approx \mathbb{E}[R_i] = 0 \quad (49)$$

$$\sigma_R^2 \approx \text{Var}[R_i] = \sigma^2 \|\mathbf{v}\|^2 + \sigma_\xi^2 \quad (50)$$

Under these simplifications,

$$\begin{aligned} P^\parallel \Delta\theta &= \frac{\alpha}{N} \sum_{i=1}^N Z_i \epsilon_i^\parallel \\ &= \frac{\alpha}{N} \sum_{i=1}^N \frac{-\sigma \|\mathbf{v}\| n_i + \sigma_\xi \xi_i - \mu_R}{\sigma_R} n_i \frac{\mathbf{v}}{\|\mathbf{v}\|} \\ &= \frac{\alpha}{N \sigma_R} \sum_{i=1}^N (-\sigma \|\mathbf{v}\| n_i + \sigma_\xi \xi_i) n_i \frac{\mathbf{v}}{\|\mathbf{v}\|} \end{aligned} \quad (51)$$

The expectation reads

$$\begin{aligned} \mathbb{E}[P^\parallel \Delta\theta] &= \frac{\alpha}{N \sigma_R} \sum_{i=1}^N \mathbb{E} [(-\sigma \|\mathbf{v}\| n_i + \sigma_\xi \xi_i) n_i] \frac{\mathbf{v}}{\|\mathbf{v}\|} \\ &= \frac{\alpha}{N \sigma_R} \mathbb{E} [-\sigma \|\mathbf{v}\| \sum_{i=1}^N n_i^2 + \sigma_\xi \sum_{i=1}^N \xi_i n_i] \frac{\mathbf{v}}{\|\mathbf{v}\|} \\ &= \frac{\alpha}{N \sigma_R} (-\sigma \|\mathbf{v}\| N) \frac{\mathbf{v}}{\|\mathbf{v}\|} \\ &= \frac{-\sigma \alpha}{\sigma_R} \mathbf{v} \\ &= \frac{-\sigma \alpha \mathbf{v}}{\sqrt{\sigma^2 \|\mathbf{v}\|^2 + \sigma_\xi^2}} \end{aligned} \quad (52)$$

where we used independence, $\mathbb{E}[\xi_i n_i] = 0$, and $\mathbb{E}[\sum_{i=1}^N n_i^2] = N$.

Next, consider the variance. Consider the individual term, $X_i = -\sigma \|\mathbf{v}\| n_i^2 + \sigma_\xi \xi_i n_i$, and its variance:

$$\begin{aligned} \text{Var}[X_i] &= \mathbb{E}[(-\sigma \|\mathbf{v}\| n_i^2 + \sigma_\xi \xi_i n_i)^2] - \mathbb{E}[-\sigma \|\mathbf{v}\| n_i^2 + \sigma_\xi \xi_i n_i]^2 \\ &= \mathbb{E}[\sigma^2 \|\mathbf{v}\|^2 \underbrace{n_i^4}_3 - 2\sigma \|\mathbf{v}\| \sigma_\xi \underbrace{\xi_i}_0 \underbrace{n_i^3}_0 + \sigma_\xi^2 \underbrace{\xi_i^2}_1 \underbrace{n_i^2}_1] - (\sigma \|\mathbf{v}\|)^2 \\ &= 3\sigma^2 \|\mathbf{v}\|^2 + \sigma_\xi^2 - (\sigma \|\mathbf{v}\|)^2 \\ &= 2\sigma^2 \|\mathbf{v}\|^2 + \sigma_\xi^2 \end{aligned} \quad (53)$$

Therefore,

$$\begin{aligned} \text{Var}[P^\parallel \Delta\theta] &= \frac{\alpha^2}{N^2 \sigma_R^2} N \text{Var}[X_i] \frac{\mathbf{v} \mathbf{v}^\top}{\|\mathbf{v}\|^2} \\ &= \frac{\alpha^2 (2\sigma^2 \|\mathbf{v}\|^2 + \sigma_\xi^2)}{N \sigma_R^2} P^\parallel \\ &= \frac{\alpha^2 (2\sigma^2 \|\mathbf{v}\|^2 + \sigma_\xi^2)}{N (\sigma^2 \|\mathbf{v}\|^2 + \sigma_\xi^2)} P^\parallel \end{aligned} \quad (54)$$

$$\begin{aligned}
\mathbb{E}[\|P^\parallel \Delta\theta\|^2] &= \text{Tr}[\text{Var}[P^\parallel \Delta\theta]] + \mathbb{E}[P^\parallel \Delta\theta]^\top \mathbb{E}[P^\parallel \Delta\theta] \\
&= \frac{\alpha^2(2\sigma^2\|\mathbf{v}\|^2 + \sigma_\xi^2)}{N(\sigma^2\|\mathbf{v}\|^2 + \sigma_\xi^2)} + \frac{(\sigma\alpha)^2\|\mathbf{v}\|^2}{\sigma^2\|\mathbf{v}\|^2 + \sigma_\xi^2} \\
&= \frac{\alpha^2}{N} + \left(1 + \frac{1}{N}\right) \frac{\sigma^2\|\mathbf{v}\|^2\alpha^2}{\sigma^2\|\mathbf{v}\|^2 + \sigma_\xi^2}
\end{aligned} \tag{55}$$

Overall distribution of the update Since the off-manifold component $P^\perp \Delta\theta$ is independent of the rewards and hence of $P^\parallel \Delta\theta$, we can combine the on- and off-manifold results to obtain the full single-step statistics. Writing $\sigma_R^2 := \sigma^2\|\mathbf{v}\|^2 + \sigma_\xi^2$,

$$\mathbb{E}[\Delta\theta] = \frac{-\sigma\alpha\mathbf{v}}{\sigma_R}, \tag{56}$$

$$\text{Cov}[\Delta\theta] = \frac{\alpha^2}{N} I_d + \frac{\alpha^2\sigma^2}{N\sigma_R^2} \mathbf{v}\mathbf{v}^\top. \tag{57}$$

The covariance has the same isotropic baseline $\frac{\alpha^2}{N} I_d$ as the flat landscape, plus a rank-1 boost along the gradient direction. The total expected squared norm is

$$\mathbb{E}\|\Delta\theta\|^2 = \frac{\sigma^2\alpha^2\|\mathbf{v}\|^2}{\sigma_R^2} + \frac{\alpha^2 d}{N} + \frac{\alpha^2\sigma^2\|\mathbf{v}\|^2}{N\sigma_R^2}. \tag{58}$$

□

Manifold Alignment of Update Consider the ratio of the on- and off-manifold movement:

$$\mathbb{E}[\|P^\parallel \Delta\theta\|^2] = \frac{\alpha^2}{N} + \left(1 + \frac{1}{N}\right) \frac{\sigma^2\|\mathbf{v}\|^2\alpha^2}{\sigma^2\|\mathbf{v}\|^2 + \sigma_\xi^2} \tag{59}$$

$$\mathbb{E}\|P^\perp \Delta\theta\|^2 = \frac{\alpha^2(d-1)}{N} \tag{60}$$

The fraction of on-manifold squared deviation in the total deviation is:

$$\begin{aligned}
\frac{\mathbb{E}[\|P^\parallel \Delta\theta\|^2]}{\mathbb{E}[\|\Delta\theta\|^2]} &= \frac{\mathbb{E}[\|P^\parallel \Delta\theta\|^2]}{\mathbb{E}[\|P^\parallel \Delta\theta\|^2] + \mathbb{E}\|P^\perp \Delta\theta\|^2} \\
&= \frac{\frac{\alpha^2}{N} + \left(1 + \frac{1}{N}\right) \frac{\sigma^2\|\mathbf{v}\|^2\alpha^2}{\sigma^2\|\mathbf{v}\|^2 + \sigma_\xi^2}}{\frac{\alpha^2(d-1)}{N} + \frac{\alpha^2}{N} + \left(1 + \frac{1}{N}\right) \frac{\sigma^2\|\mathbf{v}\|^2\alpha^2}{\sigma^2\|\mathbf{v}\|^2 + \sigma_\xi^2}} \\
&= \frac{\frac{1}{N} + \frac{N+1}{N} \frac{\sigma^2\|\mathbf{v}\|^2}{\sigma^2\|\mathbf{v}\|^2 + \sigma_\xi^2}}{\frac{d}{N} + \frac{N+1}{N} \frac{\sigma^2\|\mathbf{v}\|^2}{\sigma^2\|\mathbf{v}\|^2 + \sigma_\xi^2}} \\
&= \frac{1 + (N+1) \frac{\sigma^2\|\mathbf{v}\|^2}{\sigma^2\|\mathbf{v}\|^2 + \sigma_\xi^2}}{d + (N+1) \frac{\sigma^2\|\mathbf{v}\|^2}{\sigma^2\|\mathbf{v}\|^2 + \sigma_\xi^2}}
\end{aligned} \tag{61}$$

where $s := \frac{\sigma^2\|\mathbf{v}\|^2}{\sigma^2\|\mathbf{v}\|^2 + \sigma_\xi^2}$.

This formulation reveals the key scaling relation of on and off manifold movement

- Signal to noise ratio $s := \frac{\sigma^2 \|\mathbf{v}\|^2}{\sigma^2 \|\mathbf{v}\|^2 + \sigma_\xi^2}$, which range from $s \in [0, 1]$
 - When no gradient exists, $\|\mathbf{v}\|^2 = 0$, then independent of number of sample N , $\rho = 1/d$. i.e. the norm on one axis among d axis is $1/d$.
 - Similarly, when $\sigma_\xi^2 \gg \sigma^2 \|\mathbf{v}\|^2$ i.e. the reward noise variance is much higher than the variation of signal under exploration range σ , then it's effectively no gradient and no guidance.
 - In the other extreme, when very little reward noise, $\sigma_\xi^2 \ll \sigma^2 \|\mathbf{v}\|^2$, $s \approx 1$, then $\rho = \frac{N+2}{d+N+1}$.
 - Via differentiation, one can show, ρ monotonically depends on signal to noise ratio s
- Full dimension d . Increase the full space dim d will monotonically decrease the on manifold fraction ρ .
- Sample size per generation N . Increase the sample size N will monotonically increase the on manifold fraction ρ .
 - Note this scaling is almost linear, so to get half of the variance on manifold, one need $(N+1)s \approx d$. In reality, $d \sim 10^9$, which makes this impossible.
 - In most realistic regimes $N \ll d$, which makes this on manifold ratio per step nearly negligible.

F.4 Quadratic landscape analysis

Proof of Proposition 3. Consider a quadratic potential centered, i.e. peaked at the origin $\theta^* = 0$.

$$R(\theta) = -\frac{1}{2}\theta^\top Q\theta \quad (62)$$

At current state θ_0 , the reward at perturbed locations with observation noise are, where $\xi_i \sim \mathcal{N}(0, 1)$, $\epsilon_i \sim \mathcal{N}(0, I_d)$

$$\begin{aligned} R_i &= R(\theta_0 + \sigma\epsilon_i) + \sigma\xi_i \\ &= -\frac{1}{2}(\theta_0 + \sigma\epsilon_i)^\top Q(\theta_0 + \sigma\epsilon_i) + \sigma\xi_i \\ &= -\frac{1}{2}\theta_0^\top Q\theta_0 - \sigma\theta_0^\top Q\epsilon_i - \frac{\sigma^2}{2}\epsilon_i^\top Q\epsilon_i + \sigma\xi_i \end{aligned} \quad (63)$$

Expectation of reward reads, which is the population mean of reward samples

$$\mathbb{E}[R_i] = -\frac{1}{2}\theta_0^\top Q\theta_0 - \frac{\sigma^2}{2}\text{Tr}Q \quad (64)$$

Center the rewards with population mean, we define

$$\tilde{R}_i := R_i - \mathbb{E}[R_i] \quad (65)$$

Centered reward has three terms, linear A , quadratic B , and noise C

$$\tilde{R}_i = \underbrace{-\sigma\theta_0^\top Q\epsilon_i}_{\text{linear } A} - \underbrace{\frac{\sigma^2}{2}(\epsilon_i^\top Q\epsilon_i - \text{Tr}Q)}_{\text{quad } B} + \underbrace{\sigma\xi_i}_{\text{scalar noise } C} \quad (66)$$

One can abbreviate $\mathbf{v} := Q^\top\theta_0$ as the negative gradient.

The weight update reads,

$$\begin{aligned} \Delta\theta &= \alpha \cdot \frac{1}{N} \sum_{i=1}^N Z_i \epsilon_i \\ &= \frac{\alpha}{N\sigma_R} \sum_{i=1}^N \tilde{R}_i \epsilon_i \end{aligned} \quad (67)$$

Since perturbations $\tilde{R}_i \epsilon_i$ are i.i.d. sampled

$$\mathbb{E}[\Delta\theta] = \frac{\alpha}{\sigma_R} \mathbb{E}[\tilde{R}_i \epsilon_i] \quad (68)$$

and

$$\text{Cov}[\Delta\theta] = \frac{\alpha^2}{N\sigma_R^2} \text{Cov}[\tilde{R}_i \epsilon_i] \quad (69)$$

One observe that both $\tilde{R}_i$ and $\tilde{R}_i \epsilon_i$ are **polynomial functions of gaussian random variables** ϵ_i and ξ_i (up to 3rd order), thus we can compute **their moment of any order** via Wick-Isserlis theorem (Wick, 1950; Isserlis, 1918), including mean and covariance of them.

Here are the specific calculations:

Reward variance Note that all three terms A, B, C are centered

$$\begin{aligned} \mathbb{E}[A] &= 0 \\ \mathbb{E}[B] &= 0 \\ \mathbb{E}[C] &= 0 \end{aligned} \quad (70)$$

Further $A \perp C$, $B \perp C$ due to independence of noise, and

$$\mathbb{E}[AB] = 0 \quad (71)$$

as it only has polynomial of ϵ_i of odd order 1 and 3, thus its expectation is 0. Thus, A, B, C are not correlated with one another. Thus,

$$\begin{aligned} \text{Var}[\tilde{R}_i] &= \mathbb{E}[\tilde{R}_i^2] \\ &= \mathbb{E}[A^2] + \mathbb{E}[B^2] + \mathbb{E}[C^2] \end{aligned} \quad (72)$$

all cross terms vanish due to no correlation.

$$\begin{aligned} \mathbb{E}[A^2] &= \mathbb{E}[(-\sigma \theta_0^\top Q \epsilon_i) \cdot (-\sigma \theta_0^\top Q \epsilon_i)] \\ &= \sigma^2 \theta_0^\top Q Q^\top \theta_0 \\ &= \sigma^2 \|\mathbf{v}\|^2 \end{aligned} \quad (73)$$

$$\mathbb{E}[C^2] = \sigma_\xi^2 \quad (74)$$

$$\begin{aligned} \mathbb{E}[B^2] &= \mathbb{E}\left[\left(\frac{\sigma^2}{2}(\epsilon_i^\top Q \epsilon_i - \text{Tr}Q)\right)^2\right] \\ &= \frac{\sigma^4}{4} \mathbb{E}[\epsilon_i^\top Q \epsilon_i \cdot (\epsilon_i^\top Q \epsilon_i - \text{Tr}Q) - \text{Tr}Q \cdot (\epsilon_i^\top Q \epsilon_i - \text{Tr}Q)] \\ &= \frac{\sigma^4}{4} \left(\mathbb{E}[\epsilon_i^\top Q \epsilon_i \cdot \epsilon_i^\top Q \epsilon_i] - (\text{Tr}Q)^2 \right) \end{aligned} \quad (75)$$

The first quintic polynomial of ϵ_i can be tackled by Wick-Isserelis theorem. With slight mis-use of notation, here we let ϵ_a denote the a th element of ϵ_i vector.

Then

$$\mathbb{E}[\epsilon_a \epsilon_b \epsilon_c \epsilon_d] = I_{ab} I_{cd} + I_{ad} I_{bc} + I_{ac} I_{bd} \quad (76)$$

thus

$$\begin{aligned} \mathbb{E}[\epsilon_i^\top Q \epsilon_i \cdot \epsilon_i^\top Q \epsilon_i] &= \mathbb{E}\left[\sum_{a,b,c,d} Q_{ab} Q_{cd} \epsilon_a \epsilon_b \epsilon_c \epsilon_d\right] \\ &= \sum_{a,b,c,d} Q_{ab} Q_{cd} \mathbb{E}[\epsilon_a \epsilon_b \epsilon_c \epsilon_d] \\ &= \sum_{a,b,c,d} Q_{ab} Q_{cd} (\delta_{ab} \delta_{cd} + \delta_{ac} \delta_{bd} + \delta_{ad} \delta_{bc}) \\ &= \text{Tr}Q \text{Tr}Q + \text{Tr}[Q^\top Q] + \text{Tr}[QQ] \end{aligned} \quad (77)$$

Thus

$$\begin{aligned} \mathbb{E}[B^2] &= \frac{\sigma^4}{4} \left(\mathbb{E}[\epsilon_i^\top Q \epsilon_i \cdot \epsilon_i^\top Q \epsilon_i] - (\text{Tr}Q)^2 \right) \\ &= \frac{\sigma^4}{4} \left((\text{Tr}Q)^2 + 2\text{Tr}[Q^2] - (\text{Tr}Q)^2 \right) \\ &= \frac{\sigma^4}{2} \text{Tr}[Q^2] \end{aligned} \quad (78)$$

In sum, the population variance of reward reads,

$$\text{Var}[\tilde{R}_i] = \sigma^2 \|\mathbf{v}\|^2 + \frac{\sigma^4}{2} \text{Tr}[Q^2] + \sigma_\xi^2 \quad (79)$$

and the population standard deviation is

$$\sigma_R = \sqrt{\sigma^2 \|\mathbf{v}\|^2 + \frac{\sigma^4}{2} \text{Tr}[Q^2] + \sigma_\xi^2} \quad (80)$$

Expectation of parameter update Next let's examine expectation of parameter change, where the key is $\mathbb{E}[\tilde{R}_i \epsilon_i]$, decomposing into three terms

$$\mathbb{E}[\tilde{R}_i \epsilon_i] = \mathbb{E}[A \epsilon_i] + \mathbb{E}[B \epsilon_i] + \mathbb{E}[C \epsilon_i] \quad (81)$$

Second term vanish due to the odd order of Gaussian variable,

$$\mathbb{E}[B \epsilon_i] = 0 \quad (82)$$

Third term vanish due to independence $\epsilon_i \perp \xi_i$

$$\mathbb{E}[C \epsilon_i] = 0 \quad (83)$$

First term is easy to calculate

$$\begin{aligned} \mathbb{E}[A \epsilon_i] &= \mathbb{E}[(-\sigma \theta_0^\top Q \epsilon_i) \epsilon_i] \\ &= -\sigma \mathbb{E}[\epsilon_i (\epsilon_i^\top \mathbf{v})] \\ &= -\sigma \mathbf{v} \end{aligned} \quad (84)$$

So in sum,

$$\mathbb{E}[\tilde{R}_i \epsilon_i] = -\sigma \mathbf{v} \quad (85)$$

$$\begin{aligned} \mathbb{E}[\Delta \theta] &= \frac{\alpha}{\sigma_R} \mathbb{E}[\tilde{R}_i \epsilon_i] \\ &= \frac{-\alpha \sigma \mathbf{v}}{\sigma_R} \\ &= \frac{-\alpha \sigma \mathbf{v}}{\sqrt{\sigma^2 \|\mathbf{v}\|^2 + \frac{\sigma^4}{2} \text{Tr}[Q^2] + \sigma_\xi^2}} \end{aligned} \quad (86)$$

which is on expectation aligned with local gradient direction, but scaled by signal to noise ratio.

Variance of parameter update Finally, let's examine variance of parameter change, where the key is $\text{Cov}[\tilde{R}_i \epsilon_i]$,

$$\begin{aligned} \text{Cov}[\tilde{R}_i \epsilon_i] &= \mathbb{E}[\tilde{R}_i \epsilon_i (\tilde{R}_i \epsilon_i)^\top] - \mathbb{E}[\tilde{R}_i \epsilon_i] \mathbb{E}[\tilde{R}_i \epsilon_i]^\top \\ &= \mathbb{E}[\tilde{R}_i^2 \epsilon_i \epsilon_i^\top] - \sigma^2 \mathbf{v} \mathbf{v}^\top \end{aligned} \quad (87)$$

Tackling the first term, using the same logic (odd order and independence) to eliminate terms,

$$\begin{aligned} \mathbb{E}[\tilde{R}_i^2 \epsilon_i \epsilon_i^\top] &= \mathbb{E}[(A + B + C)^2 \epsilon_i \epsilon_i^\top] \\ &= \mathbb{E}[(A^2 + B^2 + C^2 + 2AB + 2BC + 2AC) \epsilon_i \epsilon_i^\top] \\ &= \mathbb{E}[(A^2 + B^2 + C^2) \epsilon_i \epsilon_i^\top] \end{aligned} \quad (88)$$

Tackle C term.

$$\begin{aligned} \mathbb{E}[C^2 \epsilon_i \epsilon_i^\top] &= \mathbb{E}[C^2] \mathbb{E}[\epsilon_i \epsilon_i^\top] \\ &= \sigma_\xi^2 I_d \end{aligned} \quad (89)$$

Tackle A term

$$\begin{aligned} \mathbb{E}[A^2 \epsilon_i \epsilon_i^\top] &= \mathbb{E}[(-\sigma \theta_0^\top Q \epsilon_i)^2 \epsilon_i \epsilon_i^\top] \\ &= \sigma^2 \mathbb{E}[(\mathbf{v}^\top \epsilon_i)^2 \epsilon_i \epsilon_i^\top] \end{aligned} \quad (90)$$

using the same Wick theorem trick we have each element

$$\begin{aligned}
\mathbb{E}[(\mathbf{v}^\top \epsilon_i)^2 \epsilon_i \epsilon_i^\top]_{cd} &= \mathbb{E}[\sum_{a,b} \mathbf{v}_a \mathbf{v}_b \epsilon_a \epsilon_b \epsilon_c \epsilon_d] \\
&= \sum_{a,b} \mathbf{v}_a \mathbf{v}_b \mathbb{E}[\epsilon_a \epsilon_b \epsilon_c \epsilon_d] \\
&= \sum_{a,b} \mathbf{v}_a \mathbf{v}_b (\delta_{ab} \delta_{cd} + \delta_{ac} \delta_{bd} + \delta_{ad} \delta_{cb}) \\
&= \|\mathbf{v}\|^2 \delta_{cd} + \mathbf{v}_c \mathbf{v}_d + \mathbf{v}_d \mathbf{v}_c
\end{aligned} \tag{91}$$

Thus,

$$\mathbb{E}[A^2 \epsilon_i \epsilon_i^\top] = \sigma^2 (\|\mathbf{v}\|^2 I_d + 2\mathbf{v}\mathbf{v}^\top) \tag{92}$$

Finally, tackle B term

$$\begin{aligned}
\mathbb{E}[B^2 \epsilon_i \epsilon_i^\top] &= \mathbb{E}[(\frac{\sigma^2}{2} (\epsilon_i^\top Q \epsilon_i - \text{Tr} Q))^2 \epsilon_i \epsilon_i^\top] \\
&= \frac{\sigma^4}{4} \mathbb{E}[(\epsilon_i^\top Q \epsilon_i)^2 - 2\text{Tr} Q \cdot \epsilon_i^\top Q \epsilon_i + (\text{Tr} Q)^2] \epsilon_i \epsilon_i^\top
\end{aligned} \tag{93}$$

where we encounter 2nd, 4th and 6th order terms.

2nd order term

$$\mathbb{E}[(\text{Tr} Q)^2 \epsilon_i \epsilon_i^\top] = (\text{Tr} Q)^2 I_d \tag{94}$$

4th order term

$$\begin{aligned}
\mathbb{E}[\epsilon_i^\top Q \epsilon_i \cdot \epsilon_i \epsilon_i^\top]_{cd} &= \mathbb{E}[\sum_{a,b} Q_{ab} \epsilon_a \epsilon_b \epsilon_c \epsilon_d] \\
&= \sum_{a,b} Q_{ab} (\delta_{ab} \delta_{cd} + \delta_{ac} \delta_{bd} + \delta_{ad} \delta_{cb}) \\
&= \text{Tr} Q \cdot \delta_{cd} + Q_{cd} + Q_{dc}
\end{aligned} \tag{95}$$

Thus

$$\begin{aligned}
\mathbb{E}[\epsilon_i^\top Q \epsilon_i \cdot \epsilon_i \epsilon_i^\top] &= \text{Tr} Q \cdot I_d + Q + Q^\top \\
&= \text{Tr} Q \cdot I_d + 2Q
\end{aligned} \tag{96}$$

6th order term

$$\mathbb{E}[(\epsilon_i^\top Q \epsilon_i)^2] \epsilon_i \epsilon_i^\top = \mathbb{E}[\sum_{a,b,c,d} Q_{ab} Q_{cd} \epsilon_a \epsilon_b \epsilon_c \epsilon_d \epsilon_e \epsilon_f] \tag{97}$$

Note this has $C_6^2 = 15$ pairings, with some diagrammatic reasoning, we can find topologically we have three types

$$\begin{aligned}
\mathbb{E}[(\epsilon_i^\top Q \epsilon_i)^2] \epsilon_i \epsilon_i^\top &= \mathbb{E}[\sum_{a,b,c,d} Q_{ab} Q_{cd} \epsilon_a \epsilon_b \epsilon_c \epsilon_d \epsilon_e \epsilon_f] \\
&= \sum_{a,b,c,d} Q_{ab} Q_{cd} \left(\underbrace{\delta_{ab} \delta_{cd} \delta_{ef}}_{1\text{case}} + \underbrace{\delta_{ac} \delta_{bd} \delta_{ef}}_{2\text{case}} + \dots + \right. \\
&\quad \left. \underbrace{\delta_{ae} \delta_{bf} \delta_{cd}}_{4\text{case}} + \dots + \underbrace{\delta_{ae} \delta_{cf} \delta_{bd}}_{8\text{case}} + \dots \right) \\
&= ((\text{Tr} Q)^2 + 2\text{Tr} Q^2) \delta_{ef} + 4\text{Tr} Q \cdot Q_{ef} + 8[Q^2]_{ef}
\end{aligned} \tag{98}$$

Thus

$$\mathbb{E}[(\epsilon_i^\top Q \epsilon_i)^2] \epsilon_i \epsilon_i^\top = ((\text{Tr} Q)^2 + 2\text{Tr} Q^2) I_d + 4\text{Tr} Q \cdot Q + 8Q^2 \tag{99}$$

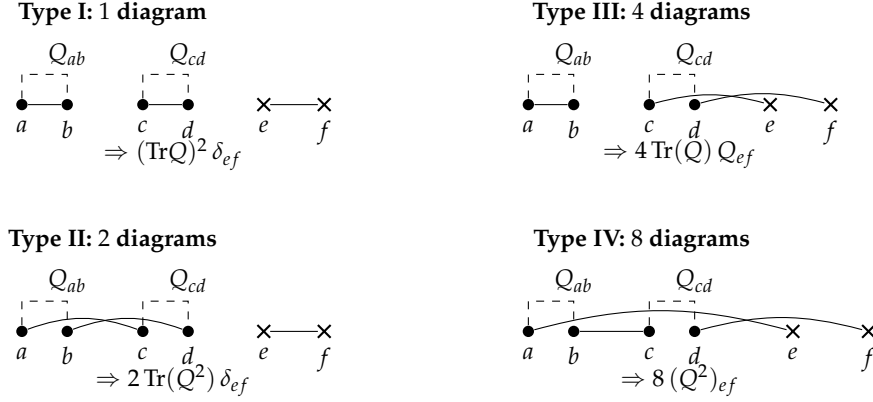


Figure 12: Wick contraction classes for $\mathbb{E}[(\epsilon^\top Q \epsilon)^2 \epsilon_e \epsilon_f]$, with external loose ends e, f shown as crosses.

In summary

$$\begin{aligned}
 \mathbb{E}[B^2 \epsilon_i \epsilon_i^\top] &= \frac{\sigma^4}{4} \mathbb{E}[(\epsilon_i^\top Q \epsilon_i)^2 - 2 \text{Tr } Q \cdot \epsilon_i^\top Q \epsilon_i + (\text{Tr } Q)^2] \epsilon_i \epsilon_i^\top \\
 &= \frac{\sigma^4}{4} \left[((\text{Tr } Q)^2 + 2 \text{Tr } Q^2) I_d + 4 \text{Tr } Q \cdot Q + 8 Q^2 \right. \\
 &\quad \left. - 2 \text{Tr } Q \cdot (\text{Tr } Q \cdot I_d + 2Q) + (\text{Tr } Q)^2 I_d \right] \\
 &= \frac{\sigma^4}{4} \left[(2 \text{Tr } Q^2) I_d + 8 Q^2 \right] \\
 &= \frac{\sigma^4}{2} \left[(\text{Tr } Q^2) \cdot I_d + 4 Q^2 \right]
 \end{aligned} \tag{100}$$

Thus

$$\begin{aligned}
 \mathbb{E}[\tilde{R}_i^2 \epsilon_i \epsilon_i^\top] &= \mathbb{E}[(A^2 + B^2 + C^2) \epsilon_i \epsilon_i^\top] \\
 &= \sigma^2 (\|\mathbf{v}\|^2 I_d + 2 \mathbf{v} \mathbf{v}^\top) + \frac{\sigma^4}{2} \left[(\text{Tr } Q^2) \cdot I_d + 4 Q^2 \right] + \sigma_\xi^2 I_d
 \end{aligned} \tag{101}$$

The covariance term reads

$$\begin{aligned}
 \text{Cov}[\tilde{R}_i \epsilon_i] &= \mathbb{E}[\tilde{R}_i^2 \epsilon_i \epsilon_i^\top] - \sigma^2 \mathbf{v} \mathbf{v}^\top \\
 &= \sigma^2 (\|\mathbf{v}\|^2 I_d + \mathbf{v} \mathbf{v}^\top) + \frac{\sigma^4}{2} \left[(\text{Tr } Q^2) \cdot I_d + 4 Q^2 \right] + \sigma_\xi^2 I_d
 \end{aligned} \tag{102}$$

Recall that $\text{Var}[\tilde{R}_i] = \sigma^2 \|\mathbf{v}\|^2 + \frac{\sigma^4}{2} \text{Tr}[Q^2] + \sigma_\xi^2$

$$\text{Cov}[\tilde{R}_i \epsilon_i] = \sigma^2 \mathbf{v} \mathbf{v}^\top + 2 \sigma^4 Q^2 + \sigma_R^2 I_d \tag{103}$$

Thus in summary we have mean and covariance of parameter update, $\mathbf{v} := Q \theta_0$

$$\begin{aligned}
 \mathbb{E}[\Delta \theta] &= \frac{\alpha}{\sigma_R} \mathbb{E}[\tilde{R}_i \epsilon_i] \\
 &= \frac{-\alpha \sigma \mathbf{v}}{\sigma_R}
 \end{aligned} \tag{104}$$

$$\begin{aligned}
\text{Cov}[\Delta\theta] &= \frac{\alpha^2}{N\sigma_R^2} \text{Cov}[\tilde{R}_i \epsilon_i] \\
&= \frac{\alpha^2}{N\sigma_R^2} \left(\sigma^2 \mathbf{v} \mathbf{v}^\top + 2\sigma^4 Q^2 + \sigma_R^2 I_d \right) \\
&= \frac{\alpha^2}{N} I_d + \frac{\alpha^2}{N\sigma_R^2} \left(\sigma^2 \mathbf{v} \mathbf{v}^\top + 2\sigma^4 Q^2 \right)
\end{aligned} \tag{105}$$

The covariance of weight update has three terms

- An isotropic variance term from the Gaussian exploration per se. $\frac{\alpha^2}{N} I_d$
- A rank-1 term along current gradient direction $\mathbf{v}$, $\frac{\alpha^2 \sigma^2}{N\sigma_R^2} \mathbf{v} \mathbf{v}^\top$
- A term corresponding to landscape curvature, i.e. higher variance along higher curvature directions, $\frac{2\sigma^4 \alpha^2}{N\sigma_R^2} Q^2$

□

Order analysis of quadratic landscape Consider the relative magnitudes of the covariance terms, using the first term as a reference:

- The second term (aligned with the gradient direction) is of order $\frac{\sigma^2}{\sigma_R^2} \|\mathbf{v}\|^2$.
- The third term (along each direction $\mathbf{u}$) is of order $\frac{2\sigma^4}{\sigma_R^2} \mathbf{u}^\top Q^2 \mathbf{u}$.

Since in practice $2\sigma^2$ is very small (e.g., 4.5×10^{-6}), the quadratic term generally acts as a small perturbation. To determine when the third (quadratic) term becomes comparable to the second, compare the terms along a specific direction $\mathbf{u}$:

$$\begin{aligned}
\frac{2\sigma^4}{\sigma_R^2} \mathbf{u}^\top Q^2 \mathbf{u} &\approx \frac{\sigma^2}{\sigma_R^2} \|\mathbf{v}\|^2 \\
2\sigma^2 \mathbf{u}^\top Q^2 \mathbf{u} &\approx \theta_0^\top Q^2 \theta_0 \\
\mathbf{u}^\top Q^2 \mathbf{u} &\approx \frac{1}{2\sigma^2} \theta_0^\top Q^2 \theta_0
\end{aligned} \tag{106}$$

Assume $\mathbf{u}$ is an eigenvector of Q with eigenvalue λ_k ,

$$\begin{aligned}
\lambda_k^2 &\approx \frac{1}{2\sigma^2} \theta_0^\top Q^2 \theta_0 \\
\lambda_k &\approx \frac{\|\mathbf{v}\|}{\sqrt{2}\sigma}
\end{aligned} \tag{107}$$

$$\lambda_k^2 \approx \frac{1}{2\sigma^2} \sum_{i=1}^d \lambda_i^2 (\theta_0^\top \mathbf{u}_i)^2 \tag{108}$$

This shows that the current gradient norm $\|\mathbf{v}\|$ needs to be much smaller (at least σ times) and the eigenvalue / curvature of the probe direction needs to be sharp enough for the 3rd term to be comparable to the 2nd term. Two scenarios this could happen

- For really sharply curved directions λ_k this is usually be true.
- The current state θ_0 is close to optimum $\theta^* = 0$ or lies in the vast valley of degenerate solution, i.e. $Q\theta_0 = 0$, θ_0 in the null space of Q .

Manifold Alignment of Update Define the projections onto the gradient direction $\mathbf{v} = Q\theta_0$ and its orthogonal complement:

$$P^\parallel := \frac{\mathbf{v}\mathbf{v}^\top}{\|\mathbf{v}\|^2}, \quad P^\perp := I_d - P^\parallel. \quad (109)$$

The on-manifold expected squared norm decomposes as

$$\mathbb{E}\|P^\parallel \Delta\theta\|^2 = \|\mathbb{E}[P^\parallel \Delta\theta]\|^2 + \text{Tr}[\text{Cov}[P^\parallel \Delta\theta]]. \quad (110)$$

For the first term, since $\mathbb{E}[\Delta\theta] = \frac{-\alpha\sigma\mathbf{v}}{\sigma_R}$ is already along $\mathbf{v}$, we have $P^\parallel \mathbb{E}[\Delta\theta] = \mathbb{E}[\Delta\theta]$, so

$$\|\mathbb{E}[P^\parallel \Delta\theta]\|^2 = \|\mathbb{E}[\Delta\theta]\|^2 = \frac{\alpha^2 \sigma^2 \|\mathbf{v}\|^2}{\sigma_R^2}. \quad (111)$$

For the second term, we project the covariance (16) onto $P^\parallel$:

$$\begin{aligned} \text{Cov}[P^\parallel \Delta\theta] &= P^\parallel \text{Cov}[\Delta\theta] P^\parallel \\ &= \frac{\alpha^2}{N\sigma_R^2} P^\parallel \left(\sigma^2 \mathbf{v}\mathbf{v}^\top + 2\sigma^4 Q^2 + \sigma_R^2 I_d \right) P^\parallel. \end{aligned} \quad (112)$$

Evaluating each term using $P^\parallel \mathbf{v}\mathbf{v}^\top P^\parallel = \|\mathbf{v}\|^2 P^\parallel$, $P^\parallel Q^2 P^\parallel = (\hat{\mathbf{v}}^\top Q^2 \hat{\mathbf{v}}) P^\parallel$ where $\hat{\mathbf{v}} := \mathbf{v}/\|\mathbf{v}\|$, and $P^\parallel I_d P^\parallel = P^\parallel$:

$$\text{Cov}[P^\parallel \Delta\theta] = \frac{\alpha^2}{N\sigma_R^2} \left(\sigma^2 \|\mathbf{v}\|^2 + 2\sigma^4 \hat{\mathbf{v}}^\top Q^2 \hat{\mathbf{v}} + \sigma_R^2 \right) P^\parallel. \quad (113)$$

Taking the trace ($\text{Tr}[P^\parallel] = 1$) and combining:

$$\mathbb{E}\|P^\parallel \Delta\theta\|^2 = \frac{\alpha^2 \sigma^2 \|\mathbf{v}\|^2}{\sigma_R^2} + \frac{\alpha^2}{N\sigma_R^2} \left(\sigma^2 \|\mathbf{v}\|^2 + 2\sigma^4 \hat{\mathbf{v}}^\top Q^2 \hat{\mathbf{v}} + \sigma_R^2 \right). \quad (114)$$

The total expected squared norm follows analogously using $\text{Tr}[\mathbf{v}\mathbf{v}^\top] = \|\mathbf{v}\|^2$, $\text{Tr}[Q^2] = \text{Tr}[Q^2]$, and $\text{Tr}[I_d] = d$:

$$\mathbb{E}\|\Delta\theta\|^2 = \frac{\alpha^2 \sigma^2 \|\mathbf{v}\|^2}{\sigma_R^2} + \frac{\alpha^2}{N\sigma_R^2} \left(\sigma^2 \|\mathbf{v}\|^2 + 2\sigma^4 \text{Tr}[Q^2] + d \sigma_R^2 \right). \quad (115)$$

Factoring out $\alpha^2/(N\sigma_R^2)$ from both expressions — noting that the mean-squared term $\frac{\alpha^2 \sigma^2 \|\mathbf{v}\|^2}{\sigma_R^2}$ contributes $N \sigma^2 \|\mathbf{v}\|^2$ after extracting the common factor, which combines with the variance contribution to yield $(N+1) \sigma^2 \|\mathbf{v}\|^2$ — the on-manifold fraction is

$$\rho := \frac{\mathbb{E}\|P^\parallel \Delta\theta\|^2}{\mathbb{E}\|\Delta\theta\|^2} = \frac{(N+1) \sigma^2 \|\mathbf{v}\|^2 + 2\sigma^4 \hat{\mathbf{v}}^\top Q^2 \hat{\mathbf{v}} + \sigma_R^2}{(N+1) \sigma^2 \|\mathbf{v}\|^2 + 2\sigma^4 \text{Tr}[Q^2] + d \sigma_R^2}. \quad (116)$$

Setting $Q^2 = 0$ and $\sigma_R^2 = \sigma^2 \|\mathbf{v}\|^2 + \sigma_\xi^2$ recovers the linear-landscape ratio $\rho = (1 + (N+1)s)/(d + (N+1)s)$ with $s = \sigma^2 \|\mathbf{v}\|^2 / \sigma_R^2$.

E.5 Multi-step ES analysis for quadratic landscape

We could write down the ES iteration as

$$\begin{aligned}\theta_1 - \theta_0 &= \frac{-\alpha\sigma\mathbf{v}}{\sigma_R} + \eta_0 \\ &= -\frac{\alpha\sigma}{\sigma_R} Q\theta_0 + \eta_0\end{aligned}\quad (117)$$

Thus

$$\theta_{t+1} = \left(I - \frac{\alpha\sigma}{\sigma_R} Q\right) \theta_t + \eta_t \quad (118)$$

where,

$$\begin{aligned}\text{Cov}[\eta_t] &= \frac{\alpha^2}{N\sigma_R^2} \left(\sigma^2\mathbf{v}\mathbf{v}^\top + 2\sigma^4 Q^2 + \sigma_R^2 I_d\right) \\ &= \frac{\alpha^2}{N} I_d + \frac{\alpha^2}{N\sigma_R^2} \left(\sigma^2 Q\theta_t\theta_t^\top Q + 2\sigma^4 Q^2\right)\end{aligned}\quad (119)$$

and

$$\begin{aligned}\sigma_R &= \sqrt{\sigma^2\|\mathbf{v}\|^2 + \frac{\sigma^4}{2}\text{Tr}[Q^2] + \sigma_\xi^2} \\ &= \sqrt{\sigma^2\theta_t^\top Q^2\theta_t + \frac{\sigma^4}{2}\text{Tr}[Q^2] + \sigma_\xi^2}\end{aligned}\quad (120)$$

Simplified ES-optimization process Consider the simplified ES optimization, where we assume the σ_R is held constant. To simplify it even more, assume η_t is Gaussian and isotropic each step $\frac{\alpha^2}{N} I_d$, i.e. ignore the 2nd and 3rd term $\frac{\alpha^2}{N\sigma_R^2} \left(\sigma^2 Q\theta_t\theta_t^\top Q + 2\sigma^4 Q^2\right)$

Thus the optimization iteration reads,

$$\theta_{t+1} = \left(I - \frac{\alpha\sigma}{\sigma_R} Q\right) \theta_t + \eta_t, \quad \eta_t \sim \mathcal{N}(0, \frac{\alpha^2}{N} I_d) \quad (121)$$

This process is known as vector autoregression VAR(1) or discrete-time OU process.

Proof of Proposition 4. We can write down the generative process of θ_T recursively.

$$\begin{aligned}\theta_t &= H\theta_{t-1} + \eta_{t-1} \\ &= H(H\theta_{t-2} + \eta_{t-2}) + \eta_{t-1} \\ &= H^2\theta_{t-2} + H\eta_{t-2} + \eta_{t-1} \\ &= H^3\theta_{t-3} + H^2\eta_{t-3} + H\eta_{t-2} + \eta_{t-1} \\ &= H^t\theta_0 + \sum_{i=1}^t H^{i-1}\eta_{t-i}\end{aligned}\quad (122)$$

Note let $\mathbf{u}_k, \lambda_k$ be the eigenvector and eigenvalue of Q , then

$$\begin{aligned}H &= I - \frac{\alpha\sigma}{\sigma_R} Q \\ &= \sum_{k=1}^d \left(1 - \frac{\alpha\sigma}{\sigma_R} \lambda_k\right) \mathbf{u}_k \mathbf{u}_k^\top\end{aligned}\quad (123)$$

In this simplification, the distribution comes from the independent Gaussian random vectors $\{\eta_i\}$, thus it's still Gaussian distributed.

Thus we can write down the expectation, which only depends on the first term.

$$\begin{aligned}\mathbb{E}[\theta_t] &= H^t \theta_0 \\ &= \sum_{k=1}^d \left(1 - \frac{\alpha\sigma}{\sigma_R} \lambda_k\right)^t \mathbf{u}_k \mathbf{u}_k^\top \theta_0\end{aligned}\quad (124)$$

Asymptotically,

$$\begin{aligned}\lim_{t \rightarrow \infty} \mathbb{E}[\theta_t] &= \lim_{t \rightarrow \infty} H^t \theta_0 \\ &= \lim_{t \rightarrow \infty} \sum_{k=1}^d \left(1 - \frac{\alpha\sigma}{\sigma_R} \lambda_k\right)^t \mathbf{u}_k \mathbf{u}_k^\top \theta_0\end{aligned}\quad (125)$$

- For flat dimensions $\lambda_k = 0$, then $\lim_{t \rightarrow \infty} \left(1 - \frac{\alpha\sigma}{\sigma_R} \lambda_k\right)^t = 1$, thus on expectation the solution won't move on that direction $\lim_{t \rightarrow \infty} \mathbb{E}[\mathbf{u}_k^\top \theta_t] = \mathbf{u}_k^\top \theta_0$
- If $|1 - \frac{\alpha\sigma}{\sigma_R} \lambda_k| > 1$, this will diverge to infinity. $\lim_{t \rightarrow \infty} \mathbb{E}[\mathbf{u}_k^\top \theta_t] = \infty$
 - Specifically, when $\lambda_k < 0$ concave dimensions
 - or for $\frac{\alpha\sigma}{\sigma_R} \lambda_k > 2$, this will diverge. This could happen for very sharply curved directions where curvature exceeds $\lambda_k > 2 \frac{\sigma_R}{\alpha\sigma}$. Note for gradient descent this could also be caused by too high learning rate, but here $\alpha = \sigma/2$.

The covariance depends on the summation $\sum_{i=1}^t H^{i-1} \eta_{t-i}$, using the i.i.d. property of the noise $\{\eta_i\}$ and the Gaussian distribution of noise,

$$\begin{aligned}\text{Cov}[\theta_t] &= \text{Cov}\left[\sum_{i=1}^t H^{i-1} \eta_{t-i}\right] \\ &= \sum_{i=1}^t \text{Cov}[H^{i-1} \eta_{t-i}] \\ &= \sum_{i=1}^t (H^{i-1}) \text{Cov}[\eta_{t-i}] (H^{i-1})^\top \\ &= \sum_{i=1}^t (H^{i-1})^2 \frac{\alpha^2}{N} I_d \\ &= \frac{\alpha^2}{N} \sum_{i=0}^{t-1} H^{2i}\end{aligned}\quad (126)$$

Thus the variance along one eigen axis is the sum of this series.

$$\begin{aligned}\text{Cov}[\mathbf{u}_k^\top \theta_t] &= \frac{\alpha^2}{N} \mathbf{u}_k^\top \sum_{i=0}^{t-1} H^{2i} \mathbf{u}_k \\ &= \frac{\alpha^2}{N} \sum_{i=0}^{t-1} \left(1 - \frac{\alpha\sigma}{\sigma_R} \lambda_k\right)^{2i}\end{aligned}\quad (127)$$

Using the formula for finite term sum of geometric series,

$$\sum_{k=0}^{t-1} x^k = \begin{cases} \frac{1-x^t}{1-x}, & x \neq 1, \\ t, & x = 1. \end{cases}\quad (128)$$

note that for finite terms this identity holds for $|x| > 1$.

The key quantity here is $\gamma^2 := (1 - \frac{\alpha\sigma}{\sigma_R}\lambda_k)^2$ and its relation to 1. $(1 - \frac{\alpha\sigma}{\sigma_R}\lambda_k)^2 = 1$ entails $\lambda_k = 0$ or $\lambda_k = \frac{2\sigma_R}{\alpha\sigma}$. Thus, along an eigen direction, the variance reads,

$$\text{Cov}[\mathbf{u}_k^\top \theta_t] = \frac{\alpha^2}{N} \begin{cases} \frac{1 - (1 - \frac{\alpha\sigma}{\sigma_R}\lambda_k)^{2t}}{1 - (1 - \frac{\alpha\sigma}{\sigma_R}\lambda_k)^2}, & \text{otherwise} \\ t, & \lambda_k \in \{0, \frac{2\sigma_R}{\alpha\sigma}\} \end{cases} \quad (129)$$

$$\text{Cov}[\mathbf{u}_k^\top \theta_t] = \frac{\alpha^2}{N} \begin{cases} \frac{1 - \gamma^{2t}}{1 - \gamma^2}, & \text{otherwise} \\ t, & \gamma^2 = 0, \text{ i.e. } \lambda_k \in \{0, \frac{2\sigma_R}{\alpha\sigma}\} \end{cases} \quad (130)$$

□

Asymptotic behavior

- In the flat direction, $\lambda_k = 0$, then $\text{Cov}[\mathbf{u}_k^\top \theta_t] = \frac{\alpha^2}{N}t$, recovering the linear scaling of random walk. Similarly for $\lambda_k = \frac{2\sigma_R}{\alpha\sigma}$, it increase linearly.
- In the concave directions, i.e. $\lambda_k < 0$, $(1 - \frac{\alpha\sigma}{\sigma_R}\lambda_k)^2 > 1$, then this series will diverge towards infinity. $\lim_{t \rightarrow \infty} \text{Cov}[\mathbf{u}_k^\top \theta_t] = \infty$
- For $\lambda_k \geq \frac{2\sigma_R}{\alpha\sigma}$, convex direction with too high curvature, then $|1 - \frac{\alpha\sigma}{\sigma_R}\lambda_k| \geq 1$, then this series will diverge $\lim_{t \rightarrow \infty} \text{Cov}[\mathbf{u}_k^\top \theta_t] = \infty$.
- For $0 < \lambda_k < \frac{2\sigma_R}{\alpha\sigma}$, then $|1 - \frac{\alpha\sigma}{\sigma_R}\lambda_k| < 1$, then this series will converge, this is the desirable case for optimization. Asymptotically,

$$\begin{aligned} \lim_{t \rightarrow \infty} \text{Cov}[\mathbf{u}_k^\top \theta_t] &= \frac{\alpha^2}{N} \frac{1}{1 - (1 - \frac{\alpha\sigma}{\sigma_R}\lambda_k)^2} \\ &= \frac{\alpha^2}{N} \frac{1}{1 - \gamma^2} \end{aligned} \quad (131)$$

- The convergence speed is inversely proportional to $\gamma^2 := (1 - \frac{\alpha\sigma}{\sigma_R}\lambda_k)^2$. When γ^2 close to 0, the series γ^{2t} converges faster; when $\gamma^2 \sim 1$, the convergence time gets longer and longer.
- Convergence timescale can be computed $1 - \gamma^{2\tau} = 1/2$, then $\log \frac{1}{2} = \tau \log \gamma^2$, $\tau = -\frac{\log 2}{\log \gamma^2} = -\frac{\log 2}{\log(1 - \frac{\alpha\sigma}{\sigma_R}\lambda_k)^2}$ is the characteristic time scale.
- The converged level is also controlled by γ^2 , specifically $\frac{\alpha^2}{N} \frac{1}{1 - \gamma^2}$. When γ^2 close to 0, converged level is lower; when $\gamma^2 \sim 1$ the converged level grow towards infinity
- Thus directions where $\lambda_k \sim \frac{\sigma_R}{\alpha\sigma}$ converge the fastest and the variance are the tightest $\lim_{t \rightarrow \infty} \text{Cov}[\mathbf{u}_k^\top \theta_t] \sim \frac{\alpha^2}{N}$. Showing that these directions are the most well controlled by the landscape asymptotically. Specifically when $\lambda_k = \frac{\sigma_R}{\alpha\sigma}$, then this direction converges in one iteration $H\theta_0$.

F.6 Gradient ascent on quadratic landscape

For comparison with the ES dynamics of Section F.5, we analyze standard (noiseless) gradient ascent⁵ on the same quadratic landscape $R(\theta) = -\frac{1}{2}\theta^\top Q\theta$.

The negative gradient at θ_t is $-\nabla R(\theta_t) = Q\theta_t = \mathbf{v}_t$. Gradient ascent with learning rate β updates as

$$\theta_{t+1} = \theta_t + \beta \nabla R(\theta_t) = \theta_t - \beta Q\theta_t = (I - \beta Q)\theta_t. \quad (132)$$

Define $H_{\text{GD}} := I - \beta Q$, with eigendecomposition

$$H_{\text{GD}} = \sum_{k=1}^d (1 - \beta\lambda_k) \mathbf{u}_k \mathbf{u}_k^\top, \quad (133)$$

where λ_k and $\mathbf{u}_k$ are the eigenvalues and eigenvectors of Q . Let $\gamma_k^{\text{GD}} := 1 - \beta\lambda_k$ denote the contraction factor along eigendirection $\mathbf{u}_k$.

Trajectory Since the iteration is deterministic, recursion gives

$$\theta_t = H_{\text{GD}}^t \theta_0 = \sum_{k=1}^d (\gamma_k^{\text{GD}})^t (\mathbf{u}_k^\top \theta_0) \mathbf{u}_k. \quad (134)$$

Each eigendirection evolves independently:

$$\mathbf{u}_k^\top \theta_t = (1 - \beta\lambda_k)^t \mathbf{u}_k^\top \theta_0. \quad (135)$$

The squared distance to the optimum $\theta^* = 0$ is

$$\|\theta_t\|^2 = \sum_{k=1}^d (1 - \beta\lambda_k)^{2t} (\mathbf{u}_k^\top \theta_0)^2. \quad (136)$$

Convergence conditions Direction $\mathbf{u}_k$ converges if and only if $|\gamma_k^{\text{GD}}| < 1$, i.e.,

$$0 < \lambda_k < \frac{2}{\beta}. \quad (137)$$

The three non-convergent modes mirror those of the ES dynamics:

- **Flat directions** ($\lambda_k = 0$): $\gamma_k^{\text{GD}} = 1$, so $\mathbf{u}_k^\top \theta_t = \mathbf{u}_k^\top \theta_0$ for all t , i.e. along flat directions, the gradient ascent freezes at the initial projection.
This is dramatically different from ES, where along flat directions, the iterate undergoes unbounded diffusion with variance growing as $\frac{\alpha^2 t}{N}$.
- **Concave directions** ($\lambda_k < 0$): $\gamma_k^{\text{GD}} > 1$, causing exponential divergence $|\mathbf{u}_k^\top \theta_t| \rightarrow \infty$.
- **Overly sharp convex directions** ($\lambda_k \geq 2/\beta$): $|\gamma_k^{\text{GD}}| \geq 1$, and the iterate overshoots, oscillating with growing (or non-decaying) amplitude. This explosion can happen due to too large learning rate, or due to too high landscape curvature.

Convergence rate Within the stable regime $0 < \lambda_k < 2/\beta$, the convergence timescale along $\mathbf{u}_k$ is

$$\tau_k^{\text{GD}} = \frac{-\log 2}{\log(\gamma_k^{\text{GD}})^2} = \frac{-\log 2}{2 \log |1 - \beta\lambda_k|}. \quad (138)$$

⁵Since we are maximizing the reward, it's gradient ascent, but still we call it GD for simplicity

The optimal eigenvalue is $\lambda_k = 1/\beta$, which achieves $\gamma_k^{\text{GD}} = 0$ and convergence in a single iteration ($\tau_k^{\text{GD}} = 0$, since $\mathbf{u}_k^\top \theta_1 = 0$ exactly). The overall convergence rate is governed by the slowest direction,

$$\tau_{\max}^{\text{GD}} = \max_{k: \lambda_k > 0} \frac{-\log 2}{2 \log |1 - \beta \lambda_k|}, \quad (139)$$

which for small learning rate $\beta \lambda_k \ll 1$ gives $\tau_k^{\text{GD}} \approx \frac{1}{2\beta \lambda_k}$, inversely proportional to the curvature. The condition number $\kappa := \lambda_{\max}/\lambda_{\min}$ (over positive eigenvalues) then controls the ratio of fastest to slowest timescales.

Comparison with ES dynamics The simplified ES iteration (Proposition 4) takes the form

$$\theta_{t+1} = H_{\text{ES}} \theta_t + \eta_t, \quad H_{\text{ES}} = I - \frac{\alpha \sigma}{\sigma_R} Q, \quad (140)$$

with $\gamma_k^{\text{ES}} = 1 - \frac{\alpha \sigma}{\sigma_R} \lambda_k$. The two iterations are structurally identical, with the correspondence $\beta \leftrightarrow \frac{\alpha \sigma}{\sigma_R}$, up to the additive noise η_t in the ES case. This yields three key differences:

1. **Effective learning rate.** The ES effective learning rate $\frac{\alpha \sigma}{\sigma_R}$ is self-normalized by the reward standard deviation σ_R , which depends on the current iterate through $\|\mathbf{v}\|^2 = \|Q\theta_t\|^2$ (in the non-simplified version). In contrast, the GD learning rate β is a fixed hyperparameter. This self-normalization provides ES with a degree of automatic step-size adaptation absent in vanilla GD.
2. **Stability threshold.** GD diverges when $\lambda_k > 2/\beta$; ES diverges when $\lambda_k > 2\sigma_R/(\alpha\sigma)$. The ES threshold is typically more permissive since $\sigma_R \geq \sigma_{\xi}$ inflates the denominator, though this comes at the cost of a smaller effective step size.
3. **Asymptotic behavior.** GD converges to $\theta^* = 0$ exactly along stable directions, with zero residual error. ES converges only in distribution, with asymptotic variance $\frac{\alpha^2}{N(1-(\gamma_k^{\text{ES}})^2)}$ per direction — a noise floor set by the exploration–exploitation tradeoff that cannot be eliminated without increasing N . On flat directions ($\lambda_k = 0$), GD stays frozen at the initial value, while ES undergoes unbounded diffusion with variance growing as $\frac{\alpha^2 t}{N}$.

These comparisons are summarized by:

	Gradient Descent	Evolution Strategy
Eff. learning rate	β	$\alpha \sigma / \sigma_R$
Contraction factor γ_k	$1 - \beta \lambda_k$	$1 - \frac{\alpha \sigma}{\sigma_R} \lambda_k$
Stability bound	$\lambda_k < 2/\beta$	$\lambda_k < 2\sigma_R/(\alpha\sigma)$
Optimal curvature	$\lambda_k = 1/\beta$	$\lambda_k = \sigma_R/(\alpha\sigma)$
Residual error (stable)	0	$\frac{\alpha^2}{N(1-\gamma_k^2)}$
Flat direction ($\lambda_k = 0$)	frozen	diffusion $\sim \frac{\alpha^2 t}{N}$

(141)

F.7 Solution geometry: GD vs. ES on quadratic landscape

We formalize the geometric claims of the main text by comparing the GD and ES solutions after T steps on the same quadratic landscape $R(\theta) = -\frac{1}{2}\theta^\top Q\theta$, starting from the same initialization θ_0 . Let $Q = \sum_k \lambda_k \mathbf{u}_k \mathbf{u}_k^\top$ with $\lambda_k \geq 0$.

GD solution From Section F.6, the GD iterate is deterministic:

$$\theta_{\text{GD}} := \theta_T^{\text{GD}} = \sum_{k=1}^d (1 - \beta \lambda_k)^T (\mathbf{u}_k^\top \theta_0) \mathbf{u}_k. \quad (142)$$

The displacement from initialization is

$$\theta_{\text{GD}} - \theta_0 = \sum_{k=1}^d [(1 - \beta\lambda_k)^T - 1] (\mathbf{u}_k^\top \theta_0) \mathbf{u}_k. \quad (143)$$

Crucially, directions with $\lambda_k = 0$ contribute nothing: $(1 - \beta \cdot 0)^T - 1 = 0$. Thus $\theta_{\text{GD}} - \theta_0$ lies entirely within the column space of Q (the “task-relevant” subspace). If Q has rank $r \ll d$, the GD displacement is confined to an r -dimensional subspace, regardless of the number of steps T .

The squared displacement along each active direction is

$$|\mathbf{u}_k^\top (\theta_{\text{GD}} - \theta_0)|^2 = [1 - (1 - \beta\lambda_k)^T]^2 (\mathbf{u}_k^\top \theta_0)^2, \quad (144)$$

and for converged directions ($|\gamma_k^{\text{GD}}| \ll 1$, large T) this saturates to $(\mathbf{u}_k^\top \theta_0)^2$, i.e., the iterate reaches the optimum $\mathbf{u}_k^\top \theta^* = 0$.

ES solution From Section F.5, the simplified ES iterate is

$$\theta_{\text{ES}} := \theta_T^{\text{ES}} = H_{\text{ES}}^T \theta_0 + \sum_{t=1}^T H_{\text{ES}}^{t-1} \eta_{T-t}, \quad (145)$$

with $H_{\text{ES}} = I - \frac{\alpha\sigma}{\sigma_R} Q$ and $\eta_t \sim \mathcal{N}(0, \frac{\alpha^2}{N} I_d)$.

The displacement decomposes as

$$\theta_{\text{ES}} - \theta_0 = \underbrace{(H_{\text{ES}}^T - I)\theta_0}_{\text{signal (deterministic)}} + \underbrace{\sum_{t=1}^T H_{\text{ES}}^{t-1} \eta_{T-t}}_{\text{noise (stochastic)}}. \quad (146)$$

The signal term has exactly the same structure as the GD displacement, with $\beta \leftrightarrow \alpha\sigma/\sigma_R$: it lies within the column space of Q and vanishes on flat directions. The noise term, however, is isotropic in origin — each η_t is drawn from $\mathcal{N}(0, \frac{\alpha^2}{N} I_d)$ — and thus has support in all d dimensions.

Along each eigendirection, the mean and variance of the ES displacement are:

$$\mathbb{E}[\mathbf{u}_k^\top (\theta_{\text{ES}} - \theta_0)] = [(\gamma_k^{\text{ES}})^T - 1] \mathbf{u}_k^\top \theta_0, \quad (147)$$

$$\text{Var}[\mathbf{u}_k^\top (\theta_{\text{ES}} - \theta_0)] = \frac{\alpha^2}{N} \cdot \begin{cases} \frac{1 - (\gamma_k^{\text{ES}})^{2T}}{1 - (\gamma_k^{\text{ES}})^2}, & (\gamma_k^{\text{ES}})^2 \neq 1, \\ T, & (\gamma_k^{\text{ES}})^2 = 1. \end{cases} \quad (148)$$

For flat directions ($\lambda_k = 0, \gamma_k^{\text{ES}} = 1$), the mean displacement is zero but the variance grows as $\frac{\alpha^2 T}{N}$ — a random walk. For active directions ($\lambda_k > 0, |\gamma_k^{\text{ES}}| < 1$), the mean displacement converges to $-\mathbf{u}_k^\top \theta_0$ (same target as GD) while the variance saturates to $\frac{\alpha^2}{N(1 - (\gamma_k^{\text{ES}})^2)}$.

Decomposition of squared displacements The total expected squared displacement of ES from initialization decomposes as

$$\begin{aligned} \mathbb{E}\|\theta_{\text{ES}} - \theta_0\|^2 &= \sum_{k: \lambda_k > 0} \left([1 - (\gamma_k^{\text{ES}})^T]^2 (\mathbf{u}_k^\top \theta_0)^2 + \frac{\alpha^2}{N} \cdot \frac{1 - (\gamma_k^{\text{ES}})^{2T}}{1 - (\gamma_k^{\text{ES}})^2} \right) \\ &\quad + \sum_{k: \lambda_k = 0} \frac{\alpha^2 T}{N}. \end{aligned} \quad (149)$$

The first sum runs over the r active directions and contains both signal and noise; the second sum runs over the $d - r$ flat directions and is pure noise. For $r \ll d$ (low-rank task structure), the flat-direction contribution dominates:

$$\mathbb{E} \|\theta_{\text{ES}} - \theta_0\|^2 \approx \underbrace{\|\theta_{\text{GD}} - \theta_0\|^2}_{\mathcal{O}(r)} + \underbrace{\frac{\alpha^2 T d}{N}}_{\text{off-manifold diffusion, } \mathcal{O}(d)}. \quad (150)$$

By contrast, GD has $\|\theta_{\text{GD}} - \theta_0\|^2 = \mathcal{O}(r)$ with no off-manifold contribution.

Geometry of $\theta_{\text{ES}} - \theta_{\text{GD}}$ The difference between the two solutions is

$$\theta_{\text{ES}} - \theta_{\text{GD}} = \underbrace{(H_{\text{ES}}^T - H_{\text{GD}}^T)\theta_0}_{\text{signal mismatch}} + \underbrace{\sum_{t=1}^T H_{\text{ES}}^{t-1} \eta_{T-t}}_{\text{ES noise}}. \quad (151)$$

The signal mismatch term lies within the column space of Q and reflects the difference in effective learning rates ($\alpha\sigma/\sigma_R$ vs. β). For converged directions where both methods have reached the optimum, this term vanishes: $(\gamma_k^{\text{ES}})^T \approx (\gamma_k^{\text{GD}})^T \approx 0$. The noise term is identical to the ES noise analyzed above.

Thus, in the large- T regime where both methods have converged on the active directions:

$$\begin{aligned} \mathbb{E} \|\theta_{\text{ES}} - \theta_{\text{GD}}\|^2 &\approx \sum_{k: \lambda_k > 0} \frac{\alpha^2}{N(1 - (\gamma_k^{\text{ES}})^2)} + \sum_{k: \lambda_k = 0} \frac{\alpha^2 T}{N} \\ &\approx \frac{\alpha^2 T(d - r)}{N} + \mathcal{O}(r), \end{aligned} \quad (152)$$

which is dominated by the off-manifold random walk for $r \ll d$.

Summary of geometric structure The three pairwise distances satisfy the approximate hierarchy (for $r \ll d$, large T):

$$\begin{aligned} \|\theta_{\text{GD}} - \theta_0\|^2 &= \mathcal{O}(r), \\ \mathbb{E} \|\theta_{\text{ES}} - \theta_0\|^2 &= \mathcal{O}(r) + \mathcal{O}(d), \\ \mathbb{E} \|\theta_{\text{ES}} - \theta_{\text{GD}}\|^2 &= \mathcal{O}(d). \end{aligned} \quad (153)$$

The first displacement is low-dimensional and lies within the column space of Q . The second is the sum of a comparable on-manifold component and a much larger isotropic off-manifold diffusion. The third is dominated by the off-manifold component alone, since both methods converge to the same optimum within the active subspace. This triangle of displacements accounts for the empirical observation that θ_{ES} achieves similar task performance to θ_{GD} (shared on-manifold component) while being far from it in parameter space (off-manifold diffusion).

Remark 10 (Cosine similarity). *The cosine similarity between the two displacement vectors is*

$$\cos \angle(\theta_{\text{ES}} - \theta_0, \theta_{\text{GD}} - \theta_0) = \frac{(\theta_{\text{ES}} - \theta_0)^\top (\theta_{\text{GD}} - \theta_0)}{\|\theta_{\text{ES}} - \theta_0\| \|\theta_{\text{GD}} - \theta_0\|}. \quad (154)$$

Since $\theta_{\text{GD}} - \theta_0$ lies within the column space of Q and the ES noise is zero-mean, the expected inner product picks up only the on-manifold signal of ES:

$$\mathbb{E}[(\theta_{\text{ES}} - \theta_0)^\top (\theta_{\text{GD}} - \theta_0)] = (\theta'_{\text{GD}} - \theta_0)^\top (\theta_{\text{GD}} - \theta_0), \quad (155)$$

where θ'_{GD} denotes GD with learning rate $\alpha\sigma/\sigma_R$ instead of β . For large T with both converged on active directions, this approaches $\|\theta_{\text{GD}} - \theta_0\|^2 = \mathcal{O}(r)$. Meanwhile, $\|\theta_{\text{ES}} - \theta_0\| = \mathcal{O}(\sqrt{d})$, so

$$\mathbb{E} [\cos \angle(\theta_{\text{ES}} - \theta_0, \theta_{\text{GD}} - \theta_0)] \sim \mathcal{O}(\sqrt{r/d}), \quad (156)$$

which is small for $r \ll d$: the two displacement vectors are nearly orthogonal despite leading to similar task performance.