

# ESSA: Evolutionary Strategies for Scalable Alignment

**Daria Korotyshova**  
T-Tech

**Boris Shaposhnikov**  
T-Tech

**Alexey Malakhov**  
T-Tech

**Alexey Khokhulin<sup>†</sup>**  
Yandex

**Nikita Surnachev**  
T-Tech

**Kirill Ovcharenko<sup>†</sup>**  
Yandex

**George Bredis**  
T-Tech

**Alexey Gorbатовski**  
T-Tech

**Viacheslav Sinii**  
T-Tech

**Daniil Gavrilov**  
T-Tech

## Abstract

Online alignment of large language models (LLMs) is dominated by reinforcement learning from human feedback (RLHF) with gradient-based optimizers such as PPO or GRPO. While effective, these pipelines require backpropagation through long rollouts, gradient synchronization across devices, and careful hyperparameter tuning, all of which become increasingly costly at scale. We present **ESSA** (Evolutionary Strategies for Scalable Alignment), a gradient-free online alignment stage that follows supervised fine-tuning (SFT) and replaces the gradient loop with inference-only black-box search. ESSA optimizes only the singular values of low-rank adaptation (LoRA) factors after a short SFT warm-start, restricting the search to a compact, task-aligned subspace where evolutionary search is practical even for 72B-parameter models. Because the loop is inference-only, ESSA is compatible with INT4/INT8 weight quantization and reduces inter-GPU communication to a few bytes per iteration. Across instruction following (IFEval), preference-based assistant tuning (HelpSteer2, HH-RLHF), and mathematical reasoning (GSM8K, MATH500), ESSA matches or exceeds LoRA-GRPO in the reported LoRA comparisons; on GSM8K it also outperforms Online DPO and PPO, while remaining competitive with both methods on IFEval. At scale, ESSA reaches a fixed MATH500 accuracy on Qwen2.5-32B/PRM800K up to  $7.8\times$  faster than LoRA-GRPO on 128 GPUs.

---

<sup>†</sup>Work performed while at T-Tech.

## 1 Introduction

Aligning large language models (LLMs) to follow instructions, respect safety constraints, and produce preferred answers is now a standard step in modern NLP pipelines (Ouyang et al., 2022a; Grattafiori et al., 2024; Lambert et al., 2024). The dominant tool for online alignment is reinforcement learning from human feedback (RLHF), typically with PPO (Schulman et al., 2017) or GRPO (Shao et al., 2024). Full-parameter PPO/GRPO implementations train an actor (and often a critic) through backpropagation over long rollouts, require all-reduce of model-sized gradient tensors at every step, and rely on careful tuning of learning rate, KL penalty, clipping, and batch size. As model size grows, these mechanics become a dominant practical cost: communication, memory, and engineering overhead all scale with the parameter count rather than with task difficulty (Zheng et al., 2023; Sheng et al., 2024).

We revisit a much simpler primitive: evolutionary search with only forward passes. Evolutionary strategies (ES) are gradient-free, embarrassingly parallel, communicate only random seeds and scalar rewards, and place no requirements on the inference engine beyond the ability to sample from the model (Salimans et al., 2017; Hansen and Ostermeier, 2001). Their classical limitation is the curse of dimensionality: applied naively to billions of parameters, ES is sample-inefficient and unstable. The question is whether a sufficiently compact, task-aligned parameterization can make ES a competitive alternative to the gradient loop in modern alignment pipelines after supervised fine-tuning.

We answer this question affirmatively with **ESSA** (Evolutionary Strategies for Scalable Alignment). ESSA targets this post-SFT online alignment stage<sup>1</sup> and combines three ingredients: (i) restricting updates to low-rank adaptation (LoRA) modules in the attention matrices (Hu et al., 2021); (ii) decomposing each LoRA factor via singular value decomposition (SVD) and freezing the singular vectors, leaving only  $\mathcal{O}(r)$  singular values per factor as trainable parameters (Sun et al., 2025); and (iii) optimizing this compact vector with CMA-ES (Hansen and Ostermeier, 2001), evaluating each population member with a single inference pass. The loop has three consequences (Figure 1):

- **Inference-only.** Because no backward pass is performed, ESSA can run natively with INT8/INT4-quantized weights, enabling one-candidate-per-GPU evaluation for models with up to 72B parameters on a single GPU.
- **Communication-minimal.** Workers exchange a 32-bit seed and a 32-bit scalar reward per evaluation, eliminating the all-reduce of model-sized tensors that bottlenecks full-parameter gradient RLHF.
- **Hyperparameter-light.** ESSA has three principal swept algorithmic knobs (LoRA rank, population size, and the fraction  $\alpha$  of singular values optimized) and we observe broad plateaus of stable accuracy for all three (Section 4.6).

**Key insight:** In our controlled comparison, population-based gradient-free optimization substantially outperforms SVD-GRPO in the same SFT-seeded singular-value space (Section 4.2).

**Contributions.** (1) We introduce ESSA, a gradient-free method for post-SFT online alignment that applies CMA-ES to the singular values of SVD-decomposed LoRA factors, targeting parameter-efficient settings with inexpensive reward evaluation. (2) We give a fair comparison against Online DPO, PPO, LoRA-GRPO, full-parameter GRPO, and SVD-GRPO across instruction following (IFEval), preference-based assistant tuning (HelpSteer2, Anthropic HH-RLHF) and verifiable-reward math (GSM8K, MATH500); ESSA matches or exceeds

LoRA-GRPO in the reported LoRA comparisons, outperforms all LoRA baselines on GSM8K, and remains competitive with PPO and Online DPO on IFEval. ESSA also matches full-parameter GRPO on math while training  $\sim 10^6 \times$  fewer parameters. (3) We characterize ESSA’s scaling from 16 to 128 GPUs using wall-clock measurements and a theoretical iteration-time bound, and show that INT4 quantization reduces accuracy by approximately one percentage point relative to BF16.

## 2 Related Work

**Online vs. offline alignment.** Online RLHF (Ouyang et al., 2022b) with PPO (Schulman et al., 2017), REINFORCE (Williams, 1992), RLOO (Ahmadian et al., 2024), GRPO (Shao et al., 2024), REINFORCE++ (Hu, 2025), or DAPO (Yu et al., 2025) remains the standard recipe for verifiable-reward and preference-based tasks. Offline preference learning (DPO (Rafailov et al., 2023), ORPO (Hong et al., 2024), SimPO (Meng et al., 2024)) removes online rollouts but is bounded by the static preference dataset and underperforms online methods (Tang et al., 2024; Xu et al., 2024a). Accordingly, we compare ESSA against LoRA-based Online DPO, PPO, and LoRA-GRPO, which share its online setting; offline methods represent an orthogonal design choice.

**Parameter-efficient and singular-value adaptation.** LoRA (Hu et al., 2021), adapters (Houlsby et al., 2019), and prefix-tuning (Li and Liang, 2021) all reduce the number of trainable parameters. A separate line of work, including Transformer<sup>2</sup> (Sun et al., 2025), SVFT (Lingam et al., 2024), SV-Fit (Sun et al., 2024), and Singular Value Fine-Tuning (Sun et al., 2022), freezes singular vectors and optimizes only singular values using gradient-based objectives in supervised fine-tuning settings. ESSA shares the singular-value parameterization but differs in two crucial ways. (i) *Online alignment setting:* prior singular-value methods are evaluated for supervised fine-tuning rather than post-SFT online policy optimization from scalar rewards. (ii) *Optimization regime:* our controlled comparisons using the same SFT initialization and singular-value coordinates show that population-based gradient-free optimization is substantially more effective than SVD-GRPO in this space (Section 4.2). Among the evaluated methods, CMA-ES offers the strongest practical trade-off between con-

<sup>1</sup>Standard alignment pipelines, such as Instruct-GPT (Ouyang et al., 2022b), Tulu 3 (Lambert et al., 2024), and Llama 3 (Grattafiori et al., 2024), apply an SFT warm-start before the RL stage, and we follow this convention. ESSA replaces only the RL stage; the SFT cost is shared with all baselines and reported separately.

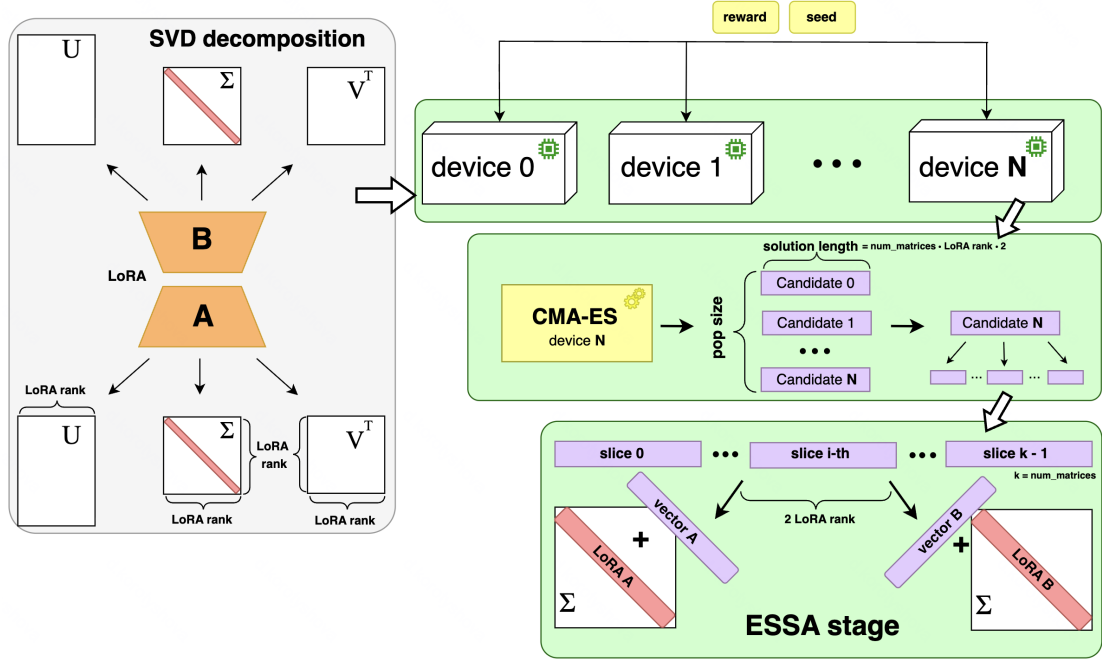


Figure 1: ESSA framework. After SFT, each LoRA factor  $A, B$  is decomposed by SVD; orthogonal bases  $U, V$  are frozen, and only the top- $\alpha$  fraction of the singular values is optimized by CMA-ES. Each device evaluates one candidate by reconstructing the LoRA factors and running an inference pass; workers exchange only random seeds and scalar rewards. **Slice**  $n$  denotes the contiguous block of singular values corresponding to one LoRA matrix (e.g.  $W_Q, W_K, W_V, W_O$  at a given transformer layer), and **device**  $N$  denotes the GPU worker.

vergence speed and tuning requirements, reaching strong performance rapidly without task-specific step-size tuning (Hansen and Ostermeier, 2001). ESSA’s contribution is the demonstration that this parameterization supports practical gradient-free online alignment at LLM scale.

**Evolutionary strategies for LLMs.** Classical ES variants (CMA-ES, NES (Wierstra et al., 2011), ARS (Mania et al., 2018), GES (Maheswaranathan et al., 2019)) offer gradient-free, highly parallel optimization with linear scaling in the number of evaluators. Zero-order methods (Zhang et al., 2024) approximate gradients from loss differences but underperform on complex reasoning. Recent ES-for-LLM work (GENOME/GENOME+ (Zhang et al., 2025), LoRAHub (Huang et al., 2024), DFO (Jin et al., 2024), ES-FT (Qiu et al., 2025), and EGGROLL (Sarkar et al., 2025)) either targets hyperscale full-parameter search or restricts ES to model merging (GENOME/GENOME+, LoRAHub), i.e. searching only for coefficients that combine pretrained checkpoints rather than learning new behaviour. Among ES-based methods for LLM fine-tuning (Appendix Table 7), ESSA uses the lowest-dimensional search space among these methods, which makes population-based search

practical at LLM scale (Salimans et al., 2017), and is the first ES-based method we are aware of to be applied to LLMs at the 72B parameter scale on a single 80 GB GPU. We also outperform EGGROLL and match ES-FT on Countdown while using a much smaller search space (Appendix E).

### 3 Method

**Setting and scope.** We consider post-SFT online alignment, where the model has already learned the task’s input/output format and is further optimized using a scalar reward provided by either a rule-based verifier or a reward model. ESSA is neither a replacement for SFT nor a method for pretraining from scratch: its singular-value updates assume that the SVD basis of the SFT-trained LoRA factors already contains task-relevant directions. We validate this assumption using a controlled MNIST analogue of the full pipeline (Appendix Figure 7). Our pipeline therefore consists of two stages: (1) standard SFT on a small subset of the data, followed by (2) ESSA optimization on a disjoint on-line split.

**LoRA + SVD parameterization.** For each attention matrix  $W_0 \in \mathbb{R}^{m \times n}$  we introduce a LoRA update  $\Delta W = BA$  with  $B \in \mathbb{R}^{m \times r}$ ,  $A \in \mathbb{R}^{r \times n}$ ,

$r \ll \min(m, n)$  (Hu et al., 2021). After SFT, we SVD-decompose each factor:

$$A = U_A \Sigma_A V_A^\top, \quad B = U_B \Sigma_B V_B^\top.$$

The orthogonal bases  $U_A, V_A, U_B, V_B$  are frozen; only the top- $\alpha$  fraction of the singular values in  $\Sigma_A, \Sigma_B$  is exposed to the optimizer. This shrinks the trainable space per factor from  $\mathcal{O}(r(m+n))$  to  $\mathcal{O}(r)$ .

**Evolutionary optimization.** Let  $\theta \in \mathbb{R}^d$  concatenate all trainable singular-value offsets ( $d \approx 1,800$  for Qwen2.5-7B with  $r = 8$  and four attention matrices per layer). At iteration  $t$ , CMA-ES (Hansen and Ostermeier, 2001) samples a population  $\{x_t^{(k)}\}_{k=1}^\lambda$  from  $\mathcal{N}(m_t, \sigma_t^2 C_t)$ , where  $m_t$  is the distribution mean,  $\sigma_t$  is the global step size, and  $C_t$  is the covariance matrix. It evaluates the scalar reward  $F(x_t^{(k)})$  for each candidate and updates  $(m_{t+1}, \sigma_{t+1}, C_{t+1})$  based on the reward rankings. Crucially, the workers share a single random-number stream, allowing each worker to locally regenerate any population member from a seed; communication is therefore reduced to  $\langle \text{seed}, \text{reward} \rangle$  pairs. The full procedure is given in Algorithm 1 (Appendix A).

**Iteration-time bound.** Let  $G$  be the number of GPUs,  $M_{\text{params}}$  the model size in bytes, and  $\eta_{\text{gen}}, \eta_{\text{fb}}$  the per-sample generation and forward-backward times. Under mild assumptions on tensor-parallel sizes (Appendix B), the ESSA iteration is strictly faster than the idealized full-parameter gradient pipeline whenever

$$N_{\text{pop}} B^{\text{essa}} < B^{\text{grad}} \left( 1 + \frac{\eta_{\text{fb}}}{G \eta_{\text{gen}}} \right) + \frac{(G-1) M_{\text{params}}}{\text{peak\_bw} G \eta_{\text{gen}}}. \quad (1)$$

Equation 1 models the full-parameter pipeline and does not describe the LoRA-GRPO baseline evaluated in Figure 4. The right-hand side grows linearly in  $M_{\text{params}}$ , so the ESSA-feasible population scales with model size (Section 4.3).

## 4 Experiments

**Tasks and backbones.** We evaluate on four task families that span the modern alignment spectrum: (i) *Instruction following*: train on an IFEval-like dataset (Xu et al., 2024b), evaluate on IFEval (Zhou et al., 2023); backbone Llama-3.1-8B (Grattafiori et al., 2024). (ii) *Preference-based assistant tuning*: train and evaluate on

HelpSteer2 (Wang et al., 2024b) with the ArmoRM reward model (Wang et al., 2024a), backbone Llama-3.1-8B; safety/helpfulness on Anthropic HH-RLHF (Bai et al., 2022) with the OpenAssistant DeBERTa-v3 reward model, backbone Mistral-7B (Jiang et al., 2023). (iii) *Mathematical reasoning*: GSM8K (Cobbe et al., 2021) and PRM800K $\rightarrow$ MATH500 (Lightman et al., 2023) plus AIME’24/’25 (Zhang and Math-AI, 2024, 2025), MinervaMath (Lewkowycz et al., 2022), OlympiadBench (He et al., 2024), AMC’23 (Li et al., 2024); backbones Qwen2.5-7B/-Math-7B/-32B/-72B (Qwen Team, 2024; Yang et al., 2024). (iv) *Symbolic puzzle*: Countdown (Appendix E).

**Baselines.** Throughout the paper, we use *LoRA-GRPO* to denote GRPO updates restricted to LoRA parameters; full-parameter GRPO and SVD-GRPO are named explicitly. We compare against four online LoRA baselines: Online DPO, PPO, LoRA-GRPO, and SVD-GRPO (GRPO restricted to singular-value updates), as well as full-parameter GRPO. For the gradient-baseline experiments reported in the main text, we train with the verl library (Sheng et al., 2024) and select the learning rate from  $\{10^{-6}, 5 \times 10^{-6}, 10^{-5}, 5 \times 10^{-5}, 10^{-4}, 5 \times 10^{-4}, 10^{-3}, 5 \times 10^{-3}, 10^{-2}\}$ . Run-level LoRA-GRPO and SVD-GRPO sweep results are reported in Appendix Tables 17, 18, and 19. SFT checkpoints are shared between ESSA and all baselines; the SFT cost is reported separately (Appendix K) and contributes an identical constant to every method. Unless noted, all methods run in BFLOAT16; the 72B experiments additionally report ESSA under INT4 (see Section 4.6 for the precision study).

### 4.1 LoRA-budget alignment

Table 1 reports test accuracy at the checkpoint from the final validation evaluation on GSM8K (verifiable rewards, Qwen2.5-7B) and IFEval (rule-based rewards, Llama-3.1-8B). The corresponding training configurations and policy-generation and reward-query counts are reported in Appendix Table 16. Among the parameter-efficient methods, ESSA achieves the highest accuracy on GSM8K while optimizing only approximately 2K–4K singular-value coordinates and requiring no backpropagation. On GSM8K, ESSA improves over the strongest LoRA baseline, LoRA-GRPO, by 4.4 percentage points (0.867 vs. 0.823). On IFEval, ESSA outperforms LoRA-GRPO by 7.2 per-

| Method                       | Optimized dim. | GSM8K                               | IFEval                              |
|------------------------------|----------------|-------------------------------------|-------------------------------------|
| SFT only                     | —              | 0.723                               | 0.451                               |
| Online DPO                   | 13/17 M        | $0.818 \pm 0.001$                   | $0.559 \pm 0.008$                   |
| PPO                          | 26/34 M        | $0.791 \pm 0.010$                   | $0.559 \pm 0.007$                   |
| LoRA-GRPO                    | 13/17 M        | $0.823 \pm 0.012$                   | $0.483 \pm 0.002$                   |
| Full-parameter GRPO (no SFT) | 7/8 B          | <b><math>0.870 \pm 0.009</math></b> | <b><math>0.692 \pm 0.008</math></b> |
| <b>ESSA (ours)</b>           | 2/4 K          | <u><math>0.867 \pm 0.010</math></u> | $0.555 \pm 0.009$                   |

Table 1: Test accuracy at the checkpoint from the final validation evaluation on **GSM8K (Qwen2.5-7B)** at LoRA rank 16 and **IFEval (Llama-3.1-8B)** at LoRA rank 8. Results for the online alignment methods are reported as mean  $\pm$  standard deviation over five independent seeds; SFT only denotes the shared initialization checkpoint. Optimized dim. denotes the number of optimized scalar parameters for **Qwen2.5-7B** on **GSM8K** / **Llama-3.1-8B** on **IFEval**. For PPO, the reported dimension includes the LoRA parameters optimized in both the policy and value models. For these main comparisons, the gradient-based LoRA baselines use the best learning rate selected from a sweep of nine values. Complete run-level LoRA-GRPO learning-rate sweep results are reported in Appendix Tables 17 and 18. Full-parameter GRPO is trained without SFT and is included as an unconstrained reference. Training configurations and policy-generation and reward-query counts are reported in Appendix Table 16. Underlining denotes the best parameter-efficient result for each task.

centage points (0.555 vs. 0.483), and is within 0.4 percentage points of Online DPO and PPO (both 0.559).

Full-parameter GRPO, which updates billions of model weights rather than the approximately 2K–4K coordinates optimized by ESSA, reaches higher accuracy on IFEval (0.692 vs. 0.555), whereas the two methods achieve comparable accuracy on GSM8K (0.870 vs. 0.867).

Figure 2 provides a separate comparison for ESSA and LoRA-GRPO on GSM8K using Qwen2.5-7B at LoRA rank 8. Additional ESSA–LoRA-GRPO comparisons across backbones, datasets, and LoRA ranks are reported in Appendix H.

## 4.2 Optimizer and subspace controls

To identify which components of ESSA drive its performance, we compare alternative optimization methods in the same singular-value space. All methods start from the same SFT checkpoint.

**SVD-GRPO ablation.** We isolate the optimization regime from the parameterization by apply-

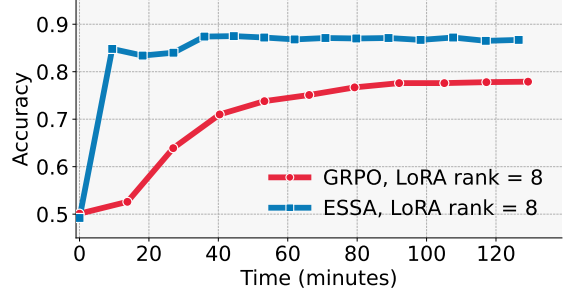


Figure 2: Validation accuracy on **GSM8K** with **Qwen2.5-7B** in a separate LoRA-rank-8 comparison. ESSA (blue): batch size 100; final test accuracy over five seeds  $0.853 \pm 0.008$ . LoRA-GRPO (red): learning rate  $1 \times 10^{-4}$ , global batch size 512, minibatch size 64; final test accuracy over five seeds  $0.847 \pm 0.014$ . ESSA reaches its plateau rapidly, while LoRA-GRPO improves more gradually. Full configurations and policy-generation and reward-query counts are reported in Appendix Table 16; complete run-level LoRA-GRPO learning-rate sweep results are reported in Appendix Table 17.

ing SVD-GRPO to the same SFT-initialized SVD-LoRA coordinates used by ESSA. For each LoRA rank  $r \in \{2, 4, 8, 16\}$ , we sweep nine learning rates and select the checkpoint using validation accuracy. At the checkpoint from the final validation evaluation, SVD-GRPO achieves MATH500 test accuracies of  $0.422 \pm 0.008$ ,  $0.322 \pm 0.012$ ,  $0.488 \pm 0.001$ , and  $0.510 \pm 0.002$  for ranks 2, 4, 8, and 16, respectively. In contrast, ESSA achieves  $0.723 \pm 0.005$  even with LoRA rank 2 (Figure 3). Thus, the parameterization alone does not explain ESSA’s performance. Configurations and generation counts are reported in Appendix Table 16; complete SVD-GRPO learning-rate sweeps are reported in Appendix Table 19.

**Alternative gradient-free optimizers.** We next compare CMA-ES with plain isotropic  $(\mu, \lambda)$ -ES, ARS, random search, and MeZO. The methods use the same SFT-seeded search space and a per-step budget of 96 candidate evaluations. CMA-ES uses the same fixed initial global step size in all experiments; step sizes for the alternative optimizers are tuned separately. Results are reported in Table 2, with convergence details in Appendix Table 10. Full configurations and generation counts are reported in Appendix Table 9.

Plain isotropic ES achieves the highest test accuracy at the checkpoint from the final validation evaluation on GSM8K, while ARS performs best on MATH500. We retain CMA-ES because it makes

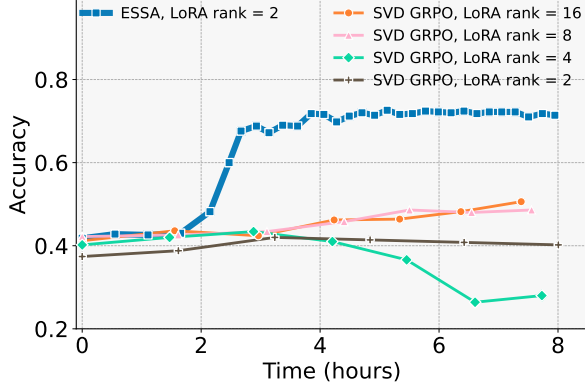


Figure 3: Validation accuracy over time on **PRM800K** with **Qwen2.5-7B**. ESSA (blue) uses LoRA rank 2, population size 24, batch size 300, and  $\alpha = 1.0$ . SVD-GRPO (singular-value updates only) uses LoRA ranks 2/4/8/16, learning rate  $1 \times 10^{-2}$ , global batch size 512, and minibatch size 64. At the checkpoints from the final validation evaluation, ESSA achieves a MATH500 test accuracy of  $0.723 \pm 0.005$  even with rank 2, whereas SVD-GRPO achieves  $0.422 \pm 0.008$ ,  $0.322 \pm 0.012$ ,  $0.488 \pm 0.001$ , and  $0.510 \pm 0.002$  for ranks 2, 4, 8, and 16, respectively. Configurations and generation counts are reported in Appendix Table 16; complete SVD-GRPO learning-rate sweeps are reported in appendix Table 19.

stronger early progress: on MATH500 it reaches 0.732 validation accuracy after approximately 2K candidate evaluations, earlier than plain ES and ARS (Appendix Table 10).

The SFT-derived basis provides an additional but smaller benefit. Under a matched 37-step GSM8K budget, it reaches 0.838 test accuracy, compared with 0.823 for a random orthogonal basis. ESSA is also stable across SFT-data fractions from 5% to 100%, with test accuracy between 0.699 and 0.728. These controls are reported in Appendix J.

### 4.3 Parallel scaling

Figure 4 compares the wall-clock time required by LoRA-GRPO and ESSA to reach a fixed MATH500 accuracy of 0.835 on Qwen2.5-32B/PRM800K as the number of GPUs increases from 16 to 128. Across this range, ESSA reduces its time to the target by approximately an order of magnitude, whereas LoRA-GRPO improves by roughly  $2.6\times$ . Consequently, ESSA’s relative wall-clock advantage increases with the number of workers and is largest in the 128-GPU configuration, where ESSA reaches the target approximately  $7.8\times$  faster. Per-GPU-count wall-clock times and the corresponding policy-generation and reward-query

| Optimizer                            | GSM8K                               | MATH500                             |
|--------------------------------------|-------------------------------------|-------------------------------------|
| CMA-ES (ESSA)                        | $0.867 \pm 0.010$                   | $0.706 \pm 0.009$                   |
| Plain isotropic $(\mu, \lambda)$ -ES | <b><math>0.893 \pm 0.006</math></b> | $0.736 \pm 0.008$                   |
| ARS                                  | $0.884 \pm 0.004$                   | <b><math>0.754 \pm 0.006</math></b> |
| Random search                        | $0.842 \pm 0.005$                   | $0.636 \pm 0.004$                   |
| MeZO (SPSA)                          | $0.826 \pm 0.008$                   | $0.472 \pm 0.010$                   |

Table 2: Test accuracy at the checkpoint from the final validation evaluation on **GSM8K** and **MATH500**, where the **MATH500** models are trained on **PRM800K**. Both experiments use **Qwen2.5-7B**, LoRA rank 16, population size 96, with per-candidate batch size 100 on **GSM8K** and 300 on **PRM800K**→**MATH500**. All methods operate on the same SFT-seeded singular-value parameterization. Results are reported as mean  $\pm$  standard deviation over five independent seeds. Full configurations and generation counts are reported in Table 9; optimizer convergence is reported in Appendix Table 10.

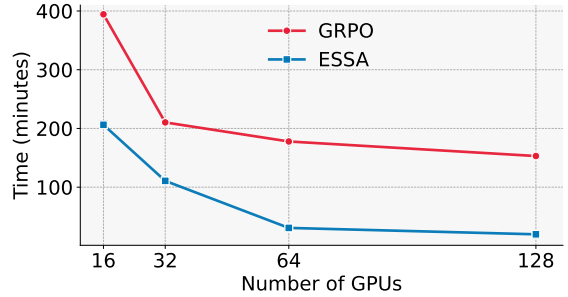


Figure 4: LoRA-GRPO and ESSA scaling on **PRM800K** with **Qwen2.5-32B**: wall-clock time to reach 0.835 MATH500 test accuracy as a function of GPU count. ESSA (red) uses LoRA rank 16, population size 128, batch size 256, and  $\alpha = 1.0$ . LoRA-GRPO (blue) uses LoRA rank 16, learning rate  $1 \times 10^{-5}$ , global prompt batch 512, actor mini-batch size 64, and 8 responses per prompt. Per-GPU-count wall-clock times and policy-generation/reward-query counts are also listed in Appendix Table 16.

counts are reported in Appendix Table 16.

This speedup is not a claim that ESSA always uses fewer policy generations. Time to accuracy depends jointly on the number of generated responses, generation throughput, and work performed outside policy generation.

**Measured step-time breakdown.** At the largest evaluated scale, we profile both methods on 128 NVIDIA H100-equivalent GPUs using the configurations from Figure 4, the same SFT checkpoint, and BF16 precision. For both ESSA and LoRA-GRPO, we first average each component time across all 128 ranks within each step. We then compute the reported mean and standard deviation across 20 pre-specified steady-state training steps.

| Step component                           | ESSA                                 |                   | LoRA-GRPO                            |                   |
|------------------------------------------|--------------------------------------|-------------------|--------------------------------------|-------------------|
|                                          | s/step                               | %                 | s/step                               | %                 |
| Candidate reconstruction                 | $5.38 \pm 0.12$                      | $2.28 \pm 0.05$   | –                                    | –                 |
| Adapter dispatch and loading             | $0.18 \pm 0.01$                      | $0.08 \pm 0.004$  | –                                    | –                 |
| Policy rollout generation                | $189.34 \pm 25.67$                   | $80.17 \pm 10.87$ | $187.82 \pm 23.75$                   | $51.91 \pm 3.02$  |
| Reward evaluation                        | $0.71 \pm 0.09$                      | $0.30 \pm 0.04$   | $1.71 \pm 0.27$                      | $0.47 \pm 0.04$   |
| Old-policy and reference forwards        | –                                    | –                 | $46.15 \pm 0.10$                     | $12.81 \pm 0.91$  |
| Actor forward                            | –                                    | –                 | $23.38 \pm 0.03$                     | $6.49 \pm 0.44$   |
| Backward compute                         | –                                    | –                 | $36.92 \pm 0.06$                     | $10.25 \pm 0.72$  |
| Exposed gradient all-reduce              | –                                    | –                 | $30.54 \pm 0.38$                     | $8.47 \pm 0.47$   |
| Optimizer step                           | –                                    | –                 | $0.339 \pm 0.002$                    | $0.094 \pm 0.007$ |
| Policy-to-rollout weight sync            | –                                    | –                 | $0.236 \pm 0.045$                    | $0.066 \pm 0.017$ |
| Search-distribution update               | $7.83 \pm 0.21$                      | $3.32 \pm 0.09$   | –                                    | –                 |
| ESSA residual orchestration and idle     | $32.74 \pm 0.78$                     | $13.86 \pm 0.33$  | –                                    | –                 |
| LoRA-GRPO barriers, idle, and stragglers | –                                    | –                 | $34.03 \pm 0.51$                     | $9.44 \pm 0.51$   |
| <b>Total step</b>                        | <b><math>236.18 \pm 25.68</math></b> | <b>100</b>        | <b><math>361.11 \pm 24.72</math></b> | <b>100</b>        |

Table 3: Step-time breakdown on 128 NVIDIA H100-equivalent GPUs for **Qwen2.5-32B** on **PRM800K**. ESSA uses LoRA rank 16, population size 128, and 256 prompts per candidate. LoRA-GRPO uses LoRA rank 16, a global prompt batch of 512, and eight responses per prompt. For each step and component, times are first averaged across all 128 ranks. The displayed mean and sample standard deviation are then computed across the same 20 pre-specified steady-state training steps. The training configurations are listed in Appendix Table 16.

**Why ESSA’s steps are faster.** The mean rollout-generation times are nearly identical for ESSA ( $189.34 \pm 25.67$  s) and LoRA-GRPO ( $187.82 \pm 23.75$  s), although an ESSA step generates 32,768 responses and a LoRA-GRPO step generates 4,096. ESSA achieves this eightfold response throughput because its 128 population members are independent and can be distributed across the cluster. LoRA-GRPO instead divides a fixed global batch of 512 prompts among the same workers, leaving little per-device work at 128 GPUs.

LoRA-GRPO is slower because each step also serializes old-policy and reference-model forwards ( $12.81 \pm 0.91\%$ ), the actor update ( $25.3\%$ ), and barriers, idle time, and stragglers ( $9.44 \pm 0.51\%$ ). ESSA performs none of the reference-model, backward, gradient-synchronization, optimizer, or policy-weight resynchronization stages; candidate reconstruction, adapter loading, and the search-distribution update together occupy less than 6% of its mean step time. Consequently, the mean complete ESSA step takes  $236.18 \pm 25.68$  s, compared with  $361.11 \pm 24.72$  s for LoRA-GRPO, corresponding to a  $1.53\times$  speedup based on the mean step times.

Gradient all-reduce accounts for  $8.47 \pm 0.47\%$  of the complete LoRA-GRPO step and is therefore not its sole bottleneck. The measured slowdown is explained by the combination of an underfilled fixed global batch, additional policy and actor com-

putations, and synchronization and idle time. The larger  $7.8\times$  time-to-accuracy advantage in Figure 4 combines this lower mean per-step cost with the number of optimization steps required to reach the target.

**Generation and reward-query accounting.** Appendix Table 16 reports training policy-generation and reward-query counts ( $G = Q$ ) for the main experiments. The counts vary in both directions: in the rank-8 GSM8K comparison, ESSA uses 201,600 generations versus 204,800 for LoRA-GRPO, whereas the main GSM8K comparison uses 489,600 versus 184,320 and IFEval uses 1,764,000 versus 122,880. These counts correspond to the displayed trajectories rather than a common accuracy threshold, so they do not constitute a uniform sample-efficiency comparison.

Each generated response is scored once, making reward-query and policy-generation counts equal. ESSA’s measured speed therefore comes from parallel generation and the absence of LoRA-GRPO’s additional forward, backward, optimizer, and synchronization stages, not from consistently using fewer samples. With expensive learned reward models, the additional reward evaluations may reduce or eliminate this advantage.

**Batch-size and theoretical scope.** We keep the LoRA-GRPO global prompt batch fixed at 512 and do not sweep batch size at 128 GPUs. Figure 4

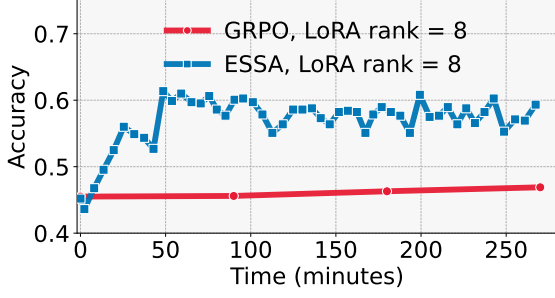


Figure 5: Validation strict accuracy over time on **IFEval** with **Llama-3.1-8B**. ESSA (blue) uses LoRA rank 8, population size 24, batch size 500, and  $\alpha = 1.0$ . LoRA-GRPO (red) uses LoRA rank 8, learning rate  $5 \times 10^{-5}$ , global batch size 512, and minibatch size 64. At the checkpoint from the final validation evaluation, final strict test accuracy over five independent seeds is  $0.555 \pm 0.009$  for ESSA and  $0.483 \pm 0.002$  for LoRA-GRPO. Full configurations and policy-generation and reward-query counts are reported in Appendix Table 16; complete run-level LoRA-GRPO learning-rate sweep results are reported in Appendix Table 18.

therefore characterizes this measured configuration, not the best possible LoRA-GRPO scaling over all batch sizes. Equation (1) describes an idealized full-parameter gradient pipeline rather than this LoRA-GRPO baseline. Accordingly, the scaling claim rests on the wall-clock measurements in Figure 4, and the explanation of the 128-GPU result rests on the measured decomposition in Table 3.

#### 4.4 Instruction following, preference, and safety

**Instruction following (IFEval).** Figure 5 compares ESSA and LoRA-GRPO on Llama-3.1-8B/IFEval using strict accuracy. ESSA reaches approximately 0.6 validation strict accuracy within 60 minutes and remains near this level, whereas LoRA-GRPO improves more gradually. At the checkpoint from the final validation evaluation, the final strict test accuracy is  $0.555 \pm 0.009$  for ESSA and  $0.483 \pm 0.002$  for LoRA-GRPO, reported over five independent seeds. Thus, ESSA’s advantage over LoRA-GRPO is not specific to verifiable-reward math and also holds for rule-based instruction following. Full configurations and policy-generation and reward-query counts are reported in Appendix Table 16; complete run-level LoRA-GRPO learning-rate sweep results are reported in Appendix Table 18.

**Preference-based assistant tuning (HelpSteer2).** On Llama-3.1-8B/HelpSteer2 with the ArmoRM

reward model, which provides a noisy learned reward signal, both ESSA and LoRA-GRPO improve over the shared SFT initialization (Appendix Figure 35). Additional diagnostics (Appendix H.3, Table 8) show that mean response length increases by 19% for ESSA and 15% for LoRA-GRPO. ESSA also remains substantially closer to the SFT policy in generation entropy: 0.735 for ESSA versus 1.973 for LoRA-GRPO, compared with 0.642 for the SFT checkpoint. Distinct-2 rises from 0.201 at SFT to 0.613 for ESSA and 0.733 for LoRA-GRPO; the relative parameter drift from the SFT LoRA update is 2.403 for ESSA and 4.233 for LoRA-GRPO. Full configurations and policy-generation and reward-query counts are reported in Appendix Table 16. Cross-domain experiments, in which the SFT initialization and the target alignment domain differ (IFEval  $\leftrightarrow$  HelpSteer2), are reported in Appendix C.

**Safety / helpfulness (HH-RLHF).** We further evaluate ESSA in the safety domain by training Mistral-7B on Anthropic HH-RLHF (Bai et al., 2022) with the OpenAssistant DeBERTa-v3 reward model. ESSA reaches a higher in-domain reward than LoRA-GRPO (Appendix Figure 36). On a held-out set of safety/helpfulness prompts, an independent GPT-4o side-by-side judge reports 12% preference for ESSA, 76% ties, and 12% preference for LoRA-GRPO, indicating parity under a judge that was not used to produce the training reward. Full configurations and policy-generation and reward-query counts are reported in Appendix Table 16.

#### 4.5 Scaling to 32B/72B on advanced math

Tables 4 and 5 report  $\text{avg}@k$  and  $\text{pass}@8$  on six advanced-math benchmarks with Qwen2.5-32B and Qwen2.5-72B. ESSA matches or exceeds LoRA-GRPO on the aggregate scores, including on AIME’24/’25 where backbone capacity matters most. At 72B we run ESSA under INT4 with one candidate per GPU; LoRA-GRPO requires tensor parallelism ( $TP = 4$ ) for the backward pass.

#### 4.6 Sensitivity and precision

**Hyperparameters.** A grid sweep over LoRA rank, population size, and  $\alpha$  for five (backbone, task) pairs (full plots in Appendix F; training-dynamics curves in Appendix G) reveals three consistent trends: (i) accuracy plateaus for  $r \geq 2$  in most settings; (ii) populations of 24–96

| Method    | MATH       | Min.       | Olym.      | AI24       | AI25       | AMC        | Avg        |
|-----------|------------|------------|------------|------------|------------|------------|------------|
| avg@k     |            |            |            |            |            |            |            |
| LoRA-GRPO | 81.8 ± 0.1 | 41.2 ± 0.1 | 45.7 ± 0.1 | 14.6 ± 0.1 | 10.0 ± 0.1 | 61.7 ± 0.1 | 42.5 ± 0.1 |
| ESSA      | 82.1 ± 0.1 | 41.8 ± 0.1 | 47.6 ± 0.1 | 17.3 ± 0.1 | 12.1 ± 0.1 | 63.3 ± 0.1 | 44.0 ± 0.1 |
| pass@8    |            |            |            |            |            |            |            |
| LoRA-GRPO | 93.6 ± 0.1 | 53.3 ± 0.1 | 65.3 ± 0.1 | 32.9 ± 0.1 | 29.2 ± 0.1 | 87.3 ± 0.1 | 60.2 ± 0.1 |
| ESSA      | 92.6 ± 0.1 | 52.2 ± 0.1 | 66.4 ± 0.1 | 35.3 ± 0.1 | 28.0 ± 0.1 | 86.9 ± 0.1 | 60.2 ± 0.1 |

Table 4: Results on advanced-math benchmarks with **Qwen2.5-32B**. ESSA uses LoRA rank 8; LoRA-GRPO uses LoRA rank 16. Results are reported as mean  $\pm$  standard deviation over five independent seeds. Rows are grouped by metric: avg@k and pass@8. For **AIME’24**, **AIME’25**, and **AMC’23**, avg@k is reported at  $k = 16$ ; for the other benchmarks,  $k = 8$ . The final **Avg** averages across all six benchmarks. Full configurations and policy-generation and reward-query counts are reported in Appendix Table 16.

| Method    | MATH       | Min.       | Olym.      | AI24       | AI25       | AMC        | Avg        |
|-----------|------------|------------|------------|------------|------------|------------|------------|
| avg@k     |            |            |            |            |            |            |            |
| LoRA-GRPO | 83.0 ± 0.1 | 43.0 ± 0.1 | 49.1 ± 0.1 | 18.8 ± 0.1 | 11.0 ± 0.5 | 62.2 ± 0.2 | 44.5 ± 0.2 |
| ESSA      | 82.8 ± 0.1 | 43.5 ± 0.5 | 48.4 ± 0.3 | 17.9 ± 0.3 | 13.3 ± 0.6 | 62.5 ± 0.3 | 44.7 ± 0.4 |
| pass@8    |            |            |            |            |            |            |            |
| LoRA-GRPO | 94.2 ± 0.2 | 51.8 ± 0.2 | 66.7 ± 0.3 | 37.5 ± 0.2 | 23.5 ± 0.2 | 85.6 ± 0.2 | 59.9 ± 0.2 |
| ESSA      | 93.0 ± 0.2 | 52.9 ± 0.4 | 66.8 ± 0.4 | 37.4 ± 0.3 | 26.8 ± 0.2 | 85.8 ± 0.1 | 60.5 ± 0.3 |

Table 5: Results on advanced-math benchmarks with **Qwen2.5-72B**. ESSA uses LoRA rank 4; LoRA-GRPO uses LoRA rank 16. Results are reported as mean  $\pm$  standard deviation over five independent seeds. Rows are grouped by metric: avg@k and pass@8. For **AIME’24**, **AIME’25**, and **AMC’23**, avg@k is reported at  $k = 16$ ; for the other benchmarks,  $k = 8$ . The final **Avg** averages across all six benchmarks. Full configurations and policy-generation and reward-query counts are reported in Appendix Table 16.

are sufficient; (iii) accuracy is stable for  $\alpha \geq 0.4$ . ESSA therefore avoids the multi-knob sensitivity (KL coefficient, clipping, learning rate, mini-batch size) of PPO/GRPO.

**Precision.** Because its optimization loop is inference-only, ESSA runs natively with INT8- or INT4-quantized weights. On Qwen2.5-32B/PRM800K, reducing precision from BF16 to INT4 decreases MATH500 test accuracy by 1.1 percentage points (Table 6). Full training curves under each precision are shown in Appendix I. INT4 execution also enables the 72B configuration to run with one candidate per NVIDIA H100-equivalent GPU. Full configurations and policy-generation and reward-query counts are reported in Appendix Table 16.

## 5 Discussion

ESSA shows that the modern post-SFT alignment stage admits a gradient-free realization that is competitive in quality, friendly to quantization, and exhibits stronger measured multi-GPU scaling than

| Weight precision | Test MATH500 accuracy |
|------------------|-----------------------|
| BF16             | 0.845 ± 0.004         |
| INT8             | 0.841 ± 0.006         |
| INT4             | 0.834 ± 0.006         |

Table 6: ESSA test accuracy at the checkpoint from the final validation evaluation as a function of **Qwen2.5-32B** weight precision on **MATH500** after training on **PRM800K**. All experiments use LoRA rank 8, population size 64, batch size 256, and  $\alpha = 1.0$ . Results are reported as mean  $\pm$  standard deviation over five independent seeds. Full configurations and policy-generation and reward-query counts are reported in Appendix Table 16.

the evaluated LoRA-GRPO configuration. The three design choices (restricting to LoRA, parameterizing by singular values, and optimizing with CMA-ES) are individually old, but combined they turn online alignment into a problem where the per-iteration cost is dominated by inference rather than by gradient-specific training and communication stages. Our comparison against Online DPO, PPO, LoRA-GRPO, full-parameter GRPO, and SVD-GRPO shows that ESSA achieves the strongest parameter-efficient result on verifiable-reward GSM8K. On instruction following, ESSA outperforms LoRA-GRPO and remains competitive with Online DPO and PPO; on the preference-based safety task, it reaches parity with LoRA-GRPO as judged by GPT-4o. The optimizer and subspace controls refine the interpretation of these results. Plain isotropic ES and ARS match or exceed CMA-ES at convergence. SVD-GRPO and MeZO remain substantially below the population-based methods on the harder MATH500 task, indicating that merely restricting a gradient method to the same coordinates does not reproduce the result. A random orthogonal basis also supports substantial improvement and comes within 1.5 percentage points of the SFT-SVD basis under a matched budget, although the SFT-SVD basis converges faster and reaches a higher plateau. Taken together, the results support a broader account: ESSA succeeds by combining a compact SFT-seeded parameter space with population-based gradient-free search. CMA-ES is a practical default because of its early progress and limited tuning requirements, rather than a uniquely necessary component.

## Limitations

**SFT initialization is required.** ESSA is positioned as a post-SFT alignment stage and relies on a strong SFT initialization; without any SFT, ESSA still improves over the base model but does not reach the SFT→ESSA accuracy (Appendix Figure 31). This dependence is shared by standard modern alignment pipelines (Ouyang et al., 2022b; Lambert et al., 2024; Grattafiori et al., 2024), but it does mean that ESSA does not replace SFT.

**Reward-evaluation cost.** ESSA can use substantially more reward evaluations than LoRA-GRPO, although the ratio varies across experiments (Appendix Table 16). With cheap rewards (rule-based verifiers or small reward models), this additional cost can be offset by the savings from removing the backward pass and associated training stages; with very expensive reward models, the higher evaluation count could erode the wall-clock advantage. The break-even point depends on the ratio between reward-model cost and policy-model cost and is a useful subject for future work.

**Low-rank assumption.** The singular-value parameterization assumes that useful alignment updates can be expressed in a low-dimensional space initialized from SFT. Our results across eight (backbone, task) pairs support this assumption in the evaluated settings, but do not establish that it holds universally. Tasks that genuinely require a high-rank update may not be reachable by ESSA at the LoRA ranks we explored.

**Mechanistic scope.** Our controls identify an effective optimization regime but do not establish a unique explanation for why population-based search performs well in this space. In particular, covariance adaptation is not necessary: plain isotropic ES and ARS match or exceed CMA-ES at convergence. The relative ranking of these optimizers may also change across tasks, reward-noise levels, and evaluation budgets. We therefore treat CMA-ES as a practical default rather than as an essential component of the method.

**Backbone diversity.** We covered Qwen2.5 (7B/Math-7B/32B/72B), Llama-3.1-8B, and Mistral-7B, but a broader study across additional families (Gemma, Phi, OLMo) would further strengthen the generality claim.

## Ethical Considerations

ESSA is a generic alignment method; it inherits the well-known dual-use considerations of any RLHF-style technique. Because ESSA reduces the engineering cost of online alignment, it may lower the barrier to both beneficial and harmful applications. Our HH-RLHF and HelpSteer2 experiments use only publicly available, license-compatible datasets and reward models, and our reward signals correspond to helpfulness/harmlessness preferences released by their original authors. We do not introduce any new dataset or release any new model weights with this work. We did not perform any human-subjects evaluation. The GPT-4o side-by-side comparison in Section 4.4 was performed via API access in compliance with OpenAI’s usage policies. All datasets (GSM8K, PRM800K, IFEval, HelpSteer2, HH-RLHF, AIME, AMC, MinervaMath, OlympiadBench), backbone models (Qwen2.5, Llama-3.1, Mistral-7B), reward models (ArmoRM, OpenAssistant DeBERTa-v3), and software (verl, vLLM, EvoTorch) are released under permissive licenses (MIT, Apache 2.0, CC-BY-4.0) or their respective model community licenses (the Llama 3 and Llama 3.1 community licenses), and our academic-research use complies with each artifact’s terms.

**Use of AI assistants.** We used AI assistants (general-purpose LLMs) for grammar checks and short-form text polishing throughout the manuscript, and as coding aides for plotting and table formatting. All scientific claims, experiments, analyses, and design decisions were authored by the human authors, and we take full responsibility for the content.

## References

- Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet Üstün, and Sara Hooker. 2024. [Back to basics: Revisiting reinforce style optimization for learning from human feedback in llms](#). *Preprint*, arXiv:2402.14740.
- Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, and 1 others. 2022. [Training a helpful and harmless assistant with reinforcement learning from human feedback](#). *Preprint*, arXiv:2204.05862.
- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman.

2021. [Training verifiers to solve math word problems](#). *Preprint*, arXiv:2110.14168.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. [The llama 3 herd of models](#). *Preprint*, arXiv:2407.21783.
- Nikolaus Hansen and Andreas Ostermeier. 2001. [Completely derandomized self-adaptation in evolution strategies](#). *Evolutionary Computation*, 9(2):159–195.
- Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, and 1 others. 2024. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. *arXiv preprint arXiv:2402.14008*.
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1026–1034.
- Jiwoo Hong, Noah Lee, and James Thorne. 2024. [Orpo: Monolithic preference optimization without reference model](#). *Preprint*, arXiv:2403.07691.
- Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. [Parameter-efficient transfer learning for nlp](#). *Preprint*, arXiv:1902.00751.
- Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. [Lora: Low-rank adaptation of large language models](#). *Preprint*, arXiv:2106.09685.
- Jian Hu. 2025. Reinforce++: A simple and efficient approach for aligning large language models. *arXiv preprint arXiv:2501.03262*.
- Chengsong Huang, Qian Liu, Bill Yuchen Lin, Tianyu Pang, Chao Du, and Min Lin. 2024. [Lorahub: Efficient cross-task generalization via dynamic lora composition](#). *Preprint*, arXiv:2307.13269.
- Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, and 1 others. 2023. [Mistral 7b](#). *Preprint*, arXiv:2310.06825.
- Feihu Jin, Yin Liu, and Ying Tan. 2024. [Derivative-free optimization for low-rank adaptation in large language models](#). *Preprint*, arXiv:2403.01754.
- Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.
- Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, and 1 others. 2024. Tulu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*.
- Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, and 1 others. 2022. Solving quantitative reasoning problems with language models. *Advances in neural information processing systems*, 35:3843–3857.
- Jia Li, Edward Beeching, Lewis Tunstall, Ben Lipkin, Roman Soletskyi, Shengyi Huang, Kashif Rasul, Longhui Yu, Albert Q Jiang, Ziju Shen, and 1 others. 2024. Numinamath: The largest public dataset in ai4maths with 860k pairs of competition math problems and solutions. *Hugging Face repository*, 13(9):9.
- Xiang Lisa Li and Percy Liang. 2021. [Prefix-tuning: Optimizing continuous prompts for generation](#). *Preprint*, arXiv:2101.00190.
- Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. [Let’s verify step by step](#). *Preprint*, arXiv:2305.20050.
- Vijay Lingam, Atula Bhaskar, Hong Yin, Aran Komatsuzaki, Pavlo Molchanov, and Hieu Phan. 2024. [SVFT: Parameter-efficient fine-tuning with singular vectors](#). In *Advances in Neural Information Processing Systems (NeurIPS)*.
- Niru Maheswaranathan, Luke Metz, George Tucker, Dami Choi, and Jascha Sohl-Dickstein. 2019. [Guided evolutionary strategies: Augmenting random search with surrogate gradients](#). In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 4264–4273. PMLR.
- Horia Mania, Aurelia Guy, and Benjamin Recht. 2018. [Simple random search provides a competitive approach to reinforcement learning](#). *Preprint*, arXiv:1803.07055.
- Yu Meng, Mengzhou Xia, and Danqi Chen. 2024. [Simpo: Simple preference optimization with a reference-free reward](#). *Preprint*, arXiv:2405.14734.
- Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022a.

- Training language models to follow instructions with human feedback. *Preprint*, arXiv:2203.02155.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. 2022b. [Training language models to follow instructions with human feedback](#). In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc.
- Xin Qiu, Yulu Gan, Conor F. Hayes, Qiyao Liang, Elliot Meyerson, Babak Hodjat, and Risto Miikkilainen. 2025. [Evolution strategies at scale: Llm fine-tuning beyond reinforcement learning](#). *Preprint*, arXiv:2509.24372.
- Qwen Team. 2024. [Qwen2.5 technical report](#). *Preprint*, arXiv:2412.15115.
- Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2023. [Direct preference optimization: Your language model is secretly a reward model](#). *Preprint*, arXiv:2305.18290.
- Tim Salimans, Jonathan Ho, Xi Chen, Szymon Sidor, and Ilya Sutskever. 2017. [Evolution strategies as a scalable alternative to reinforcement learning](#). *Preprint*, arXiv:1703.03864.
- Bidipta Sarkar, Mattie Fellows, Juan Agustin Duque, Alistair Letcher, Antonio León Villares, Anya Sims, Dylan Cope, Jarek Liesen, Lukas Seier, Theo Wolf, Uljad Berdica, Alexander David Goldie, Aaron Courville, Karin Sevegnani, Shimon Whiteson, and Jakob Nicolaus Foerster. 2025. [Evolution strategies at the hyperscale](#). *Preprint*, arXiv:2511.16652.
- John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. [Proximal policy optimization algorithms](#). *Preprint*, arXiv:1707.06347.
- Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. [Deepseekmath: Pushing the limits of mathematical reasoning in open language models](#). *Preprint*, arXiv:2402.03300.
- Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. 2024. [HybridFlow: A flexible and efficient RLHF framework](#). *arXiv preprint arXiv:2409.19256*.
- Qi Sun, Edoardo Cetin, and Yujin Tang. 2025. [Transformer-squared: Self-adaptive llms](#). *Preprint*, arXiv:2501.06252.
- Yanpeng Sun, Qiang Chen, Xiangyu He, Jian Wang, Haocheng Feng, Junyu Han, and 1 others. 2022. Singular value fine-tuning: Few-shot segmentation requires few-parameters fine-tuning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pages 37484–37496.
- Yiren Sun, Zerui Li, Zihan Liu, and Bo Chen. 2024. [SVFit: Parameter-efficient fine-tuning of large pre-trained models using singular values](#). *Preprint*, arXiv:2409.05926.
- Yunhao Tang, Daniel Zhaoan Guo, Zeyu Zheng, Daniele Calandriello, Yuan Cao, Eugene Tarassov, Rémi Munos, Bernardo Ávila Pires, Michal Valko, Yong Cheng, and Will Dabney. 2024. [Understanding the performance gap between online and offline alignment algorithms](#). *Preprint*, arXiv:2405.08448.
- Haoxiang Wang, Wei Xiong, Tengyang Xie, Han Zhao, and Tong Zhang. 2024a. [Interpretable preferences via multi-objective reward modeling and mixture-of-experts](#). In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 10582–10592, Miami, Florida, USA. Association for Computational Linguistics.
- Zhilin Wang, Yi Dong, Olivier Delalleau, Jiaqi Zeng, Gerald Shen, Daniel Egert, Jimmy J. Zhang, Makesh Narsimhan Sreedhar, and Oleksii Kuchaiev. 2024b. [HelpSteer2: Open-source dataset for training top-performing reward models](#). In *Advances in Neural Information Processing Systems*, volume 37, pages 1474–1501.
- Daan Wierstra, Tom Schaul, Tobias Glasmachers, Yi Sun, and Jürgen Schmidhuber. 2011. [Natural evolution strategies](#). *Preprint*, arXiv:1106.4487.
- Ronald J. Williams. 1992. [Simple statistical gradient-following algorithms for connectionist reinforcement learning](#). *Machine Learning*, 8(3–4):229–256.
- Shusheng Xu, Wei Fu, Jiaxuan Gao, Wenjie Ye, Weilin Liu, Zhiyu Mei, Guangju Wang, Chao Yu, and Yi Wu. 2024a. [Is dpo superior to ppo for llm alignment? a comprehensive study](#). *Preprint*, arXiv:2404.10719.
- Zhangchen Xu, Fengqing Jiang, Luyao Niu, Yuntian Deng, Radha Poovendran, Yejin Choi, and Bill Yuchen Lin. 2024b. [Magpie: Alignment data synthesis from scratch by prompting aligned llms with nothing](#). *Preprint*, arXiv:2406.08464.
- An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, Keming Lu, Mingfeng Xue, Runji Lin, Tianyu Liu, Xingzhang Ren, and Zhenru Zhang. 2024. [Qwen2.5-math technical report: Toward mathematical expert model via self-improvement](#). *Preprint*, arXiv:2409.12122.
- Qiyang Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan

- Tong, Chi Zhang, Mofan Zhang, Wang Zhang, and 16 others. 2025. [Dapo: An open-source llm reinforcement learning system at scale](#). *Preprint*, arXiv:2503.14476.
- Yifan Zhang and Team Math-AI. 2024. American invitational mathematics examination (aime) 2024.
- Yifan Zhang and Team Math-AI. 2025. American invitational mathematics examination (aime) 2025.
- Yihua Zhang, Pingzhi Li, Junyuan Hong, Jiayang Li, Yimeng Zhang, Wenqing Zheng, Pin-Yu Chen, Jason D. Lee, Wotao Yin, Mingyi Hong, Zhangyang Wang, Sijia Liu, and Tianlong Chen. 2024. [Revisiting zeroth-order optimization for memory-efficient llm fine-tuning: A benchmark](#). *Preprint*, arXiv:2402.11592.
- Yiqun Zhang, Peng Ye, Xiaocui Yang, Shi Feng, Shufei Zhang, Lei Bai, Wanli Ouyang, and Shuyue Hu. 2025. [Nature-inspired population-based evolution of large language models](#). *Preprint*, arXiv:2503.01155.
- Rui Zheng, Shihan Dou, Songyang Gao, Yuan Hua, Wei Shen, Binghai Wang, Yan Liu, Senjie Jin, Qin Liu, Yuhao Zhou, Limao Xiong, Lu Chen, Zhiheng Xi, Nuo Xu, Wenbin Lai, Minghao Zhu, Cheng Chang, Zhangyue Yin, Rongxiang Weng, and 8 others. 2023. [Secrets of rlhf in large language models part i: Ppo](#). *Preprint*, arXiv:2307.04964.
- Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. 2023. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*.

---

**Algorithm 1** ESSA: Distributed Evolutionary Search over LoRA Singular Values

---

**Require:** Training set  $\mathcal{D}$ ; SFT-initialized LoRA factors  $\{A_i, B_i\}$ ; fraction  $\alpha$ ; population  $P$ ; GPUs  $N$ ; generations  $E$ .

- 1: **SVD init:** for each  $A_i, B_i$  compute  $A_i = U_{A_i} \Sigma_{A_i} V_{A_i}^\top$ ,  $B_i = U_{B_i} \Sigma_{B_i} V_{B_i}^\top$ ; freeze  $U, V$ ; concatenate top- $\alpha$  singular values into  $\sigma \in \mathbb{R}^d$ .
  - 2: Initialize  $N$  workers with identical seeds and a common CMA-ES state  $(m_0, \sigma_0, C_0)$ .
  - 3: **for**  $t = 0, \dots, E - 1$  **do**
  - 4:   **for** worker  $k = 1, \dots, N$  **in parallel do**
  - 5:     Sample candidate offsets  $\{\delta^{(k,j)}\}$  from shared seeds.
  - 6:     **for** each local candidate  $j$  **do**
  - 7:       Reconstruct  $A'_i = U_{A_i} \text{diag}(\sigma_{A_i} + \delta_{A_i}) V_{A_i}^\top$  and similarly  $B'_i$ ; set  $W'_i = W_i + B'_i A'_i$ .
  - 8:       Sample mini-batch  $\mathcal{D}_{\text{mini}} \subset \mathcal{D}$  and compute reward  $F$ .
  - 9:     **end for**
  - 10:   **end for**
  - 11:   All workers exchange  $\langle \text{seed}, F \rangle$  pairs.
  - 12:   Each worker reconstructs the same population and performs one CMA-ES update of  $(m, \sigma, C)$ .
  - 13: **end for**
  - 14: Return aligned model with final singular values.
- 

## A ESSA Algorithm

Algorithm 1 formalises the full ESSA training loop sketched in Section 3: SVD initialisation of the LoRA factors from the SFT checkpoint, parallel candidate sampling across  $N$  workers, fitness evaluation by a single forward pass per candidate, and a CMA-ES update per generation. Workers exchange only  $\langle \text{seed}, F \rangle$  pairs, keeping per-step communication constant in model size.

## B Detailed Iteration-Time Analysis

This appendix expands the iteration-latency analysis sketched in Section 4.3, comparing one optimization step of an idealized full-parameter gradient-based RLHF pipeline against one ESSA step. The gradient pipeline incurs an all-reduce of a model-sized tensor, whereas ESSA exchanges only small per-candidate records, such as random seeds or identifiers and scalar rewards. This analysis does not model the LoRA-GRPO baseline measured in

Section 4.3. The proposition below isolates a sufficient condition under which ESSA is strictly faster per iteration under the idealized timing model.

**Proposition 1** (ESSA iteration-time advantage). *Assume  $m_{\text{fb}} \geq 1$ ,  $m_{\text{gen}} \leq G$ , and that the total ESSA synchronization payload satisfies  $M_{\text{essa}} \leq M_{\text{params}}$ . If the ESSA population size  $N_{\text{pop}}$  and per-candidate batch size  $B^{\text{essa}}$  satisfy*

$$N_{\text{pop}} B^{\text{essa}} < B^{\text{grad}} \left( 1 + \frac{\eta_{\text{fb}}}{G \eta_{\text{gen}}} \right) + \frac{(G-1) M_{\text{params}}}{\text{peak\_bw } G \eta_{\text{gen}}}, \quad (2)$$

*then the modeled ESSA iteration time is strictly smaller than that of the idealized full-parameter gradient pipeline:  $T^{\text{essa}} < T^{\text{grad}}$ .*

To prove the proposition, we (i) introduce notation, (ii) bound per-iteration computation and communication, (iii) optimize the device split between training and generation, and (iv) derive a conservative population-size bound under which ESSA is guaranteed to be faster within the timing model.

**Notation.** Let  $B^{\text{grad}}$  be the global batch size used in gradient methods,  $B^{\text{essa}}$  the batch size processed by a single population instance,  $b_{\text{fb}}, b_{\text{gen}}$  the microbatch sizes, and  $m_{\text{fb}}, m_{\text{gen}}$  the numbers of GPUs per model instance for training and generation. Let  $G$  be the total number of GPUs, with  $G_{\text{fb}}$  and  $G_{\text{gen}}$  allocated to training and generation, respectively. Let  $\tau_{\text{fb}}(b)$  and  $\tau_{\text{gen}}(b)$  be the forward-backward and generation microbatch times, and define  $\eta_{\text{fb}} = \tau_{\text{fb}}(b_{\text{fb}})/b_{\text{fb}}$  and  $\eta_{\text{gen}} = \tau_{\text{gen}}(b_{\text{gen}})/b_{\text{gen}}$  as the corresponding per-sample times. Finally, let  $k_{\text{fb}}^{\text{par}} = G_{\text{fb}}/m_{\text{fb}}$  and  $k_{\text{gen}}^{\text{par}} = G_{\text{gen}}/m_{\text{gen}}$  be the numbers of microbatches processed in parallel.

Then  $T^{\text{grad}} = T_{\text{fb-gen}}^{\text{grad}} + T_{\text{comm}}^{\text{grad}}$  and  $T^{\text{essa}} = T_{\text{gen}}^{\text{essa}} + T_{\text{comm}}^{\text{essa}}$ . Under ideal asynchronous training and generation,

$$T_{\text{fb-gen}}^{\text{grad}} = \max(T_{\text{fb}}^{\text{grad}}, T_{\text{gen}}^{\text{grad}}).$$

**Computation.** Processing  $B^{\text{grad}}$  samples requires  $B^{\text{grad}}/b_{\text{fb}}$  microbatches, of which  $k_{\text{fb}}^{\text{par}}$  run in parallel:

$$T_{\text{fb}}^{\text{grad}} = \frac{B^{\text{grad}} m_{\text{fb}}}{G_{\text{fb}}} \eta_{\text{fb}},$$

$$T_{\text{gen}}^{\text{grad}} = \frac{B^{\text{grad}} m_{\text{gen}}}{G_{\text{gen}}} \eta_{\text{gen}}.$$

For ESSA, the entire population is evaluated by generation only:

$$T_{\text{gen}}^{\text{essa}} = \frac{N_{\text{pop}} B^{\text{essa}} m_{\text{gen}}}{G} \eta_{\text{gen}}.$$

**Communication.** Let  $M_{\text{params}}$  be the model-parameter size in bytes. For the idealized full-parameter gradient pipeline, a ring all-reduce consists of a reduce-scatter and an all-gather:

$$T_{\text{comm}}^{\text{grad}} = 2 \cdot \frac{M_{\text{params}}(G-1)}{G \text{ peak\_bw}}.$$

Let  $M_{\text{essa}}$  denote the total payload synchronized by ESSA in one iteration, including the scalar rewards and any random seeds or candidate identifiers required to reconstruct the population. This payload grows with the population size but remains negligible relative to a model-sized tensor for the configurations considered here. Under the same bandwidth model,

$$T_{\text{comm}}^{\text{essa}} = \frac{M_{\text{essa}}(G-1)}{G \text{ peak\_bw}}.$$

**Optimal device split.** Let  $\theta \in (0, 1)$  be the fraction of devices used for training, so that  $G_{\text{fb}} = \theta G$  and  $G_{\text{gen}} = (1 - \theta)G$ . Then

$$T_{\text{fb-gen}}^{\text{grad}}(\theta) = \frac{B^{\text{grad}}}{G} \max\left(\frac{m_{\text{fb}}\eta_{\text{fb}}}{\theta}, \frac{m_{\text{gen}}\eta_{\text{gen}}}{1-\theta}\right).$$

**Lemma 2** (Optimal split). *The minimum of  $T_{\text{fb-gen}}^{\text{grad}}(\theta)$  over  $\theta \in (0, 1)$  is attained at*

$$\theta^* = \frac{m_{\text{fb}}\eta_{\text{fb}}}{m_{\text{fb}}\eta_{\text{fb}} + m_{\text{gen}}\eta_{\text{gen}}},$$

$$T_{\text{fb-gen}}^{\text{grad}}(\theta^*) = \frac{B^{\text{grad}}}{G} (m_{\text{fb}}\eta_{\text{fb}} + m_{\text{gen}}\eta_{\text{gen}}).$$

*Proof.* The function  $\max(a/\theta, b/(1-\theta))$  is minimized when its two arguments are equal:  $a/\theta = b/(1-\theta)$ , where  $a = m_{\text{fb}}\eta_{\text{fb}}$  and  $b = m_{\text{gen}}\eta_{\text{gen}}$ .  $\square$

Define the ideal gradient iteration time under perfect scheduling:

$$T_{\star}^{\text{grad}} = \frac{B^{\text{grad}}}{G} (m_{\text{fb}}\eta_{\text{fb}} + m_{\text{gen}}\eta_{\text{gen}}) + 2 \cdot \frac{M_{\text{params}}(G-1)}{G \text{ peak\_bw}}.$$

Under the timing model,  $T^{\text{grad}} \geq T_{\star}^{\text{grad}}$ .

**Theorem 3** (ESSA iteration is faster under a conservative bound). *Assume  $m_{\text{fb}} \geq 1$ ,  $m_{\text{gen}} \leq G$ , and  $M_{\text{essa}} \leq M_{\text{params}}$ . If Equation 2 holds, then*

$$T^{\text{essa}} < T_{\star}^{\text{grad}} \leq T^{\text{grad}}$$

*under the timing model defined above.*

*Proof.* Substituting the computation and communication terms into  $T^{\text{essa}} < T_{\star}^{\text{grad}}$  gives

$$N_{\text{pop}} B^{\text{essa}} < B^{\text{grad}} \left(1 + \frac{m_{\text{fb}}\eta_{\text{fb}}}{m_{\text{gen}}\eta_{\text{gen}}}\right) + \frac{(2M_{\text{params}} - M_{\text{essa}})(G-1)}{\text{peak\_bw } m_{\text{gen}}\eta_{\text{gen}}}.$$
(3)

Because  $m_{\text{fb}} \geq 1$  and  $m_{\text{gen}} \leq G$ ,

$$\frac{m_{\text{fb}}}{m_{\text{gen}}} \geq \frac{1}{G}.$$

Moreover,  $M_{\text{essa}} \leq M_{\text{params}}$  implies  $2M_{\text{params}} - M_{\text{essa}} \geq M_{\text{params}}$ , and  $m_{\text{gen}} \leq G$  implies  $1/m_{\text{gen}} \geq 1/G$ . Therefore, Equation 2 is a sufficient condition for Equation 3, and hence for  $T^{\text{essa}} < T_{\star}^{\text{grad}}$ . The optimal-split construction gives  $T_{\star}^{\text{grad}} \leq T^{\text{grad}}$ .  $\square$

**Discussion.** Theorem 3 gives a conservative threshold on the ESSA population size below which its modeled iteration time is smaller than that of the idealized full-parameter gradient pipeline. The right-hand side of Equation 2 grows linearly with  $M_{\text{params}}$  when the other quantities are fixed. Its dependence on  $G$  occurs through  $1/G$  and  $(G-1)/G$  and therefore saturates as the cluster size increases. Accordingly, the bound supports the model-size argument, while the empirical cluster-scaling behavior is established separately by the wall-clock measurements in Section 4.3.

## C Cross-Domain Robustness

This section extends the IFEval and HelpSteer2 experiments in Section 4.4 and the LoRA-budget results in Section 4.1 by evaluating ESSA when the SFT initialization and the target alignment task come from different domains.

To assess robustness under distribution shift, we conduct additional cross-domain experiments using two datasets with substantially different reward structures: HelpSteer2 and IFEval. HelpSteer2 encourages general-purpose response quality through

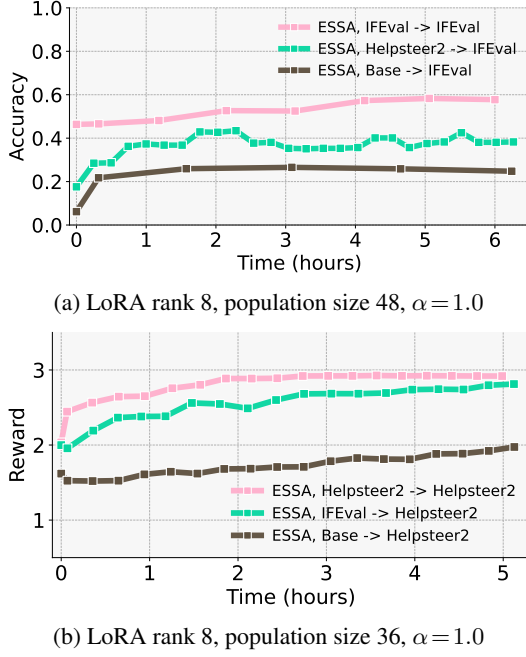


Figure 6: Validation metrics over time on **IFEval** (a) and **HelpSteer2** (b) with **Llama-3.1-8B**. (a) IFEval → IFEval (pink), HelpSteer2 → IFEval (green), base model → IFEval (brown) experiments, batch size 500. (b): HelpSteer2 → HelpSteer2 (pink), IFEval → HelpSteer2 (green), base model → HelpSteer2 (brown) experiments, batch size 100.

a learned reward model, whereas IFEval evaluates strict constraint satisfaction using rule-based criteria. Moving between these domains therefore tests whether ESSA remains effective when the SFT warm-start and the target alignment domain differ. The HelpSteer2 experiments use the same RLHFlow/ArmoRM-Llama3-8B-v0.1 reward model as the corresponding main-text experiment. The IFEval experiments train on the same IFEval-like dataset used in the main text.

For each target dataset  $B$ , we evaluate three configurations. The first performs SFT on dataset  $A$  and subsequently applies ESSA to dataset  $B$  ( $A \rightarrow B$ ). The second performs both SFT and ESSA on dataset  $B$  ( $B \rightarrow B$ ). The third applies ESSA directly to dataset  $B$  without an SFT warm-start (Base model  $\rightarrow B$ ); LoRA factor  $A$  is initialized using Kaiming-uniform initialization (He et al., 2015), while factor  $B$  is initialized to zero.

In both evaluated directions (Figure 6), ESSA improves performance on the target dataset even when the SFT checkpoint originates from the other domain. In the IFEval-to-HelpSteer2 condition, the gap between cross-domain and in-domain SFT is modest, indicating that the initialization learned

from the IFEval-like dataset provides useful directions for subsequent optimization on HelpSteer2. In the opposite direction, the gap is larger, consistent with HelpSteer2 not explicitly teaching the strict formatting and constraint-following behavior required by IFEval. Nevertheless, ESSA improves over the Base model  $\rightarrow$  IFEval configuration initialized with random LoRA parameters.

In both directions, the SFT-initialized configurations outperform optimization from the base model, indicating that an out-of-domain SFT checkpoint can still provide useful structure for ESSA’s evolutionary search. These results show that ESSA remains effective under the two evaluated distribution shifts, although in-domain SFT remains the strongest initialization.

## D Toy Example on MNIST

This section provides a minimal illustration of the singular-value-only parameterization introduced in Section 3 and complements the subspace controls in Section 4.2.

A natural concern with a singular-value-only parameterization is that it may be too restrictive in a simple linear or logistic-regression setting. When the singular directions of a fixed low-rank update are held constant, only their magnitudes can change, which may appear to provide insufficient flexibility for meaningful adaptation.

ESSA’s effectiveness for LLM alignment relies on the structured representations already present in the pretrained backbone and refined by the downstream SFT initialization. To illustrate this distinction, we design an MNIST experiment that mirrors the essential initialization and parameterization stages of the LLM experiments within a minimal linear model. The MNIST experiment proceeds as follows:

1. At the pretrain stage (analog of LLM pre-training) we train a logistic regression model on MNIST digits 0-5 (train size 5000) using SGD optimizer. We obtain a "base model"  $W = W'$ , where  $W'$  is the single matrix of size 784 by 10, that knows something about the data distribution, similar to LLM pretraining.
2. At the "SFT" stage (analogue of LoRA SFT) we take 400 samples of digits 6-9, initialize a matrix update  $\Delta W$  with Kaiming init, and train only  $\Delta W$  using SGD:  $W = W' + \Delta W$ ,

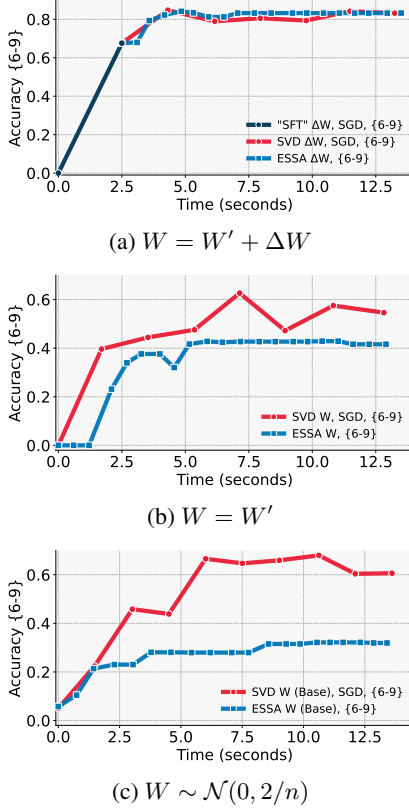


Figure 7: Validation accuracy over time on **MNIST**, evaluated on the test set containing digits 6–9, using a single matrix  $W \in \mathbb{R}^{n \times m}$  with  $n = 784$  and  $m = 10$ . Panels (a)–(c) correspond to the setups  $W = W' + \Delta W$ ,  $W = W'$ , and  $W \sim \mathcal{N}(0, 2/n)$ , respectively, where  $W'$  denotes the pretrained base weights. The training set contains 400 examples. ESSA (blue): solution length 10, population size 24, and batch size 64. SGD (red): learning rate 0.1 and batch size 64.

where  $\Delta W$  is the single matrix of size 784 by 10.

- At the last stage we decomposed  $\Delta W = BSA$  and train only the singular values  $S$  using ES or SGD optimizer. ESSA optimizes singular values directly using train accuracy, while the SGD baseline optimized them through the standard cross-entropy loss.

In Figure 7a, ESSA follows the behavior of SGD on the MNIST task with high fidelity. This demonstrates that once the singular vectors encode useful directions, modulating only the associated singular amplitudes is already sufficient to achieve effective adaptation. In Figure 7b, we remove the additive update entirely. We decompose the pretrained weights  $W = W' = B'S'A'$ , and train only the singular values  $S'$  using ESSA and SGD. Even in this more constrained setup, ESSA achieved perfor-

mance comparable to SGD. The pretrained matrix contained enough structure that rebalancing its singular directions remained a viable strategy for improving accuracy. Finally, in Figure 7c, we examine the case without pretraining or SFT, where the model is initialized with Kaiming-normal weights  $W \sim \mathcal{N}(0, 2/n)$  and the singular vectors carry no semantic information. In this setup ESSA improves only slowly, while SGD makes substantially faster progress. Without a structured subspace, changing only singular values is insufficient to emulate full learning dynamics.

These experiments demonstrate that the singular-value parameterization is effective precisely in the settings where LLMs operate: pretrained models with downstream SFT produce meaningful low-rank directions, and ESSA leverages this structure by adjusting their relative scaling. This behavior is consistent with the results observed in our large-scale LLM experiments.

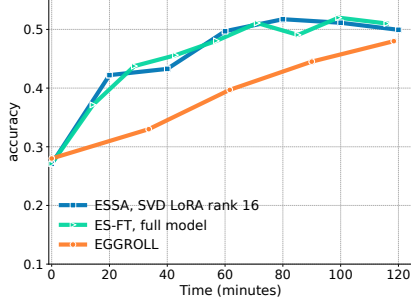


Figure 8: Validation accuracy over time on **Countdown** with **Qwen2.5-7B** using eight GPUs. ESSA (blue): population size 48,  $\alpha = 1.0$ , and batch size 200. ES-FT (green):  $\sigma = 1 \times 10^{-3}$ ,  $1r = 1 \times 10^{-4}$ , population size 48, and batch size 200. EGGROLL (orange):  $\sigma = 1 \times 10^{-3}$ , rank 1, and batch size 200.

| Method                        | Search dim.           | Tasks                                                  |
|-------------------------------|-----------------------|--------------------------------------------------------|
| ESSA (ours)                   | $\mathcal{O}(r)$      | GSM8K, PRM800K, IFEval, HelpSteer2, HH-RLHF, Countdown |
| ES-FT (Qiu et al., 2025)      | $\mathcal{O}(nm)$     | Countdown, MATH, ARC-AGI, Sudoku                       |
| EGGROLL (Sarkar et al., 2025) | $\mathcal{O}(r(n+m))$ | Countdown, GSM8K, MATH500, AIME, OlympiadBench, AMC    |

Table 7: ES-based methods for LLM fine-tuning. ESSA’s search dimension is  $\mathcal{O}(r)$  per LoRA factor, substantially smaller than that of full-parameter ES (ES-FT,  $\mathcal{O}(M_{\text{params}})$ ) and LoRA-coefficient ES (EGGROLL,  $\mathcal{O}(r(m+n))$  per matrix). For Qwen2.5-7B with 28 layers, four adapted attention matrices per layer, and  $r = 8$ , ESSA exposes  $28 \times 4 \times 2r \approx 1.8\text{K}$  singular-value coordinates: the two LoRA factors  $A$  and  $B$  each contribute  $r$  coordinates per adapted matrix. ES-FT searches the full  $\sim 7\text{B}$  model parameters, whereas EGGROLL searches  $28 \times 4 \times r(m+n) \approx 5\text{M}$  LoRA coefficients.

## E Other ES Methods on Countdown

This section provides a cross-task extension of the alternative gradient-free optimizer comparison in Section 4.2. Whereas the main-text experiments compare optimizers on GSM8K and PRM800K  $\rightarrow$  MATH500, the following experiment compares ESSA with two existing ES-based LLM fine-tuning methods on Countdown.

Figure 8 compares ESSA, ES-FT, and EGGROLL for optimizing Qwen2.5-7B on Countdown using eight GPUs. Table 7 summarizes their search dimensions and task coverage. ESSA substantially outperforms EGGROLL while matching the convergence of ES-FT, which fine-tunes the full model.

## F Hyperparameter Sensitivity

This section provides the complete results supporting the hyperparameter-sensitivity analysis in Section 4.6. The GSM8K experiments additionally complement the LoRA-budget results in Section 4.1; the PRM800K experiments complement the optimizer and subspace analysis in Section 4.2; and the IFEval experiments complement the instruction-following results in Section 4.4.

Figures 9–13 present five sensitivity studies: Qwen2.5-Math-7B and Qwen2.5-7B on GSM8K, Qwen2.5-Math-7B and Qwen2.5-7B on PRM800K, and Llama-3.1-8B on IFEval.

The five sweeps support three trends summarized in the main text: (i) accuracy peaks at low LoRA ranks ( $r \in \{2, 4, 8\}$ ) in four of the five settings, while Qwen2.5-Math-7B/GSM8K favors a larger rank; (ii) populations of 24–96 are generally sufficient, with diminishing returns beyond 96; and (iii) accuracy is broadly stable for  $\alpha \geq 0.4$ , indicating limited sensitivity to the fraction of singular-value coordinates optimized.

### F.0.1 Qwen2.5-Math-7B on GSM8K

This experiment supports Section 4.6 and extends the GSM8K results in Section 4.1 to Qwen2.5-Math-7B. Figure 9 shows that accuracy peaks at 0.896 for LoRA rank 32 and population size 96. Moderate ranks also perform strongly: ranks 4, 8, and 16 reach accuracies of 0.886, 0.886, and 0.882, respectively, at population size 48. Population size 8 generally underperforms, whereas performance remains relatively stable across population sizes 24–192.

When the population size is fixed to the best value for each rank, ESSA maintains high accuracy across nearly all evaluated values of  $\alpha$ . Rank 32 reaches 0.899 at  $\alpha = 0.1$ , rank 16 reaches 0.891 at  $\alpha = 0.6$ , and rank 8 reaches 0.886 at  $\alpha = 1.0$ . Smaller fractions can therefore match or outperform  $\alpha = 1.0$ , indicating that optimizing all singular-value coordinates is not required for strong performance.

### F.0.2 Qwen2.5-7B on GSM8K

This experiment provides the sensitivity analysis for the Qwen2.5-7B/GSM8K setting used in Section 4.1. Figure 10 shows stable performance across a broad range of configurations, with accuracy peaking at 0.893 for LoRA rank 2 and population size 96. LoRA ranks 2–4 achieve the strongest

performance, while larger ranks provide diminishing or slightly degraded results.

With population size fixed, ESSA maintains accuracy of approximately 0.87–0.89 across the evaluated values of  $\alpha$ , indicating robustness to the fraction of singular-value coordinates optimized.

### F.0.3 Qwen2.5-Math-7B on PRM800K

This experiment supports Section 4.6 and complements the PRM800K→MATH500 analysis in Section 4.2. Figure 11 shows that accuracy peaks at 0.784 for LoRA rank 2 and population size 24. Higher ranks or very large populations gradually reduce performance, indicating that increasing the search dimension does not improve the result.

With population size fixed, the highest score in the corresponding  $\alpha$  sweep is 0.616 at LoRA rank 8 and  $\alpha = 1.0$ , while intermediate values of  $\alpha$  remain competitive.

### F.0.4 Qwen2.5-7B on PRM800K

This experiment provides additional sensitivity results for the Qwen2.5-7B/PRM800K setting analyzed in Section 4.2. Figure 12 shows that accuracy peaks at 0.748 for LoRA rank 2 and population size 48. Performance decreases for larger ranks or oversized populations, indicating diminishing returns beyond a compact search space.

With population size fixed, intermediate values  $\alpha \approx 0.4$ –0.8 perform best, reaching up to 0.696 accuracy at LoRA rank 8, while very small fractions and  $\alpha = 1.0$  perform slightly worse.

### F.0.5 Llama-3.1-8B on IFEval

This experiment provides the sensitivity analysis for the Llama-3.1-8B/IFEval setting used in Sections 4.1 and 4.4. Figure 13 shows stable performance across configurations, with the highest accuracy of 0.614 achieved at LoRA rank 8 and population size 24. Ranks 4–16 yield comparable results, while rank 32 performs slightly worse, indicating that low or moderate ranks and population sizes are sufficient in this setting.

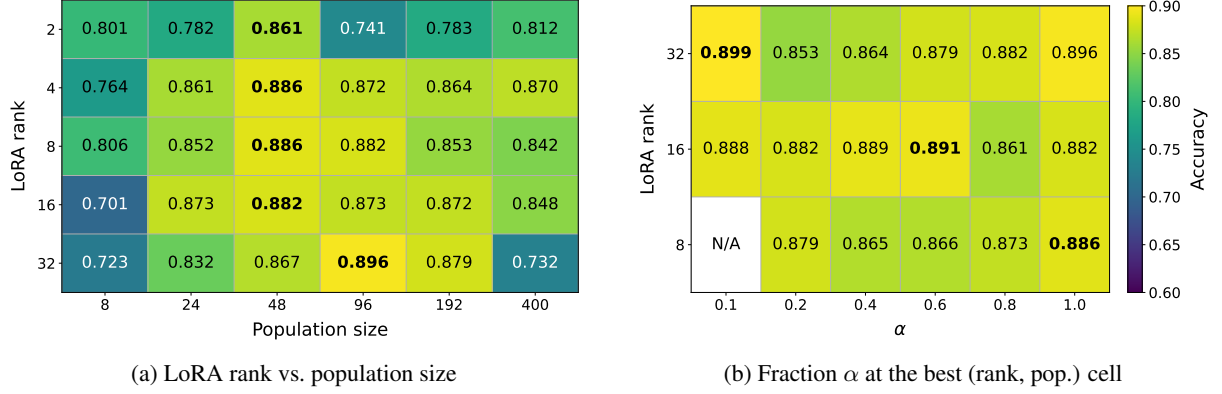


Figure 9: Hyperparameter sensitivity of ESSA on **Qwen2.5-Math-7B** for **GSM8K**. Batch size 100. (a) Accuracy when varying LoRA rank and population size. (b) For each LoRA rank, the population size is fixed to the best value found in (a), while the percentage  $\alpha$  of trainable singular values is varied. This illustrates how ESSA performance depends jointly on adapter rank and the fraction of singular values optimized. The single white cell occurs because for LoRA rank 8 and  $\alpha=0.1$ , rounding down yields zero trainable singular values, so no valid accuracy is reported.

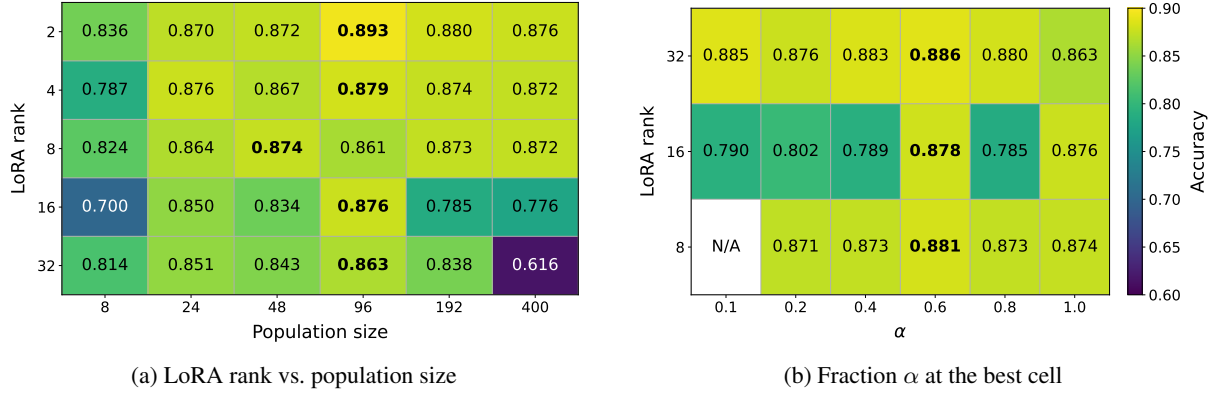


Figure 10: Hyperparameter sensitivity of ESSA on **Qwen2.5-7B** for **GSM8K**. Batch size 100. (a) Accuracy when varying LoRA rank and population size. (b) For each LoRA rank, the population size is fixed to the best value found in (a), while the percentage  $\alpha$  of trainable singular values is varied. This illustrates how ESSA performance depends jointly on adapter rank and the fraction of singular values optimized. The single white cell occurs because for LoRA rank 8 and  $\alpha=0.1$ , rounding down yields zero trainable singular values, so no valid accuracy is reported.

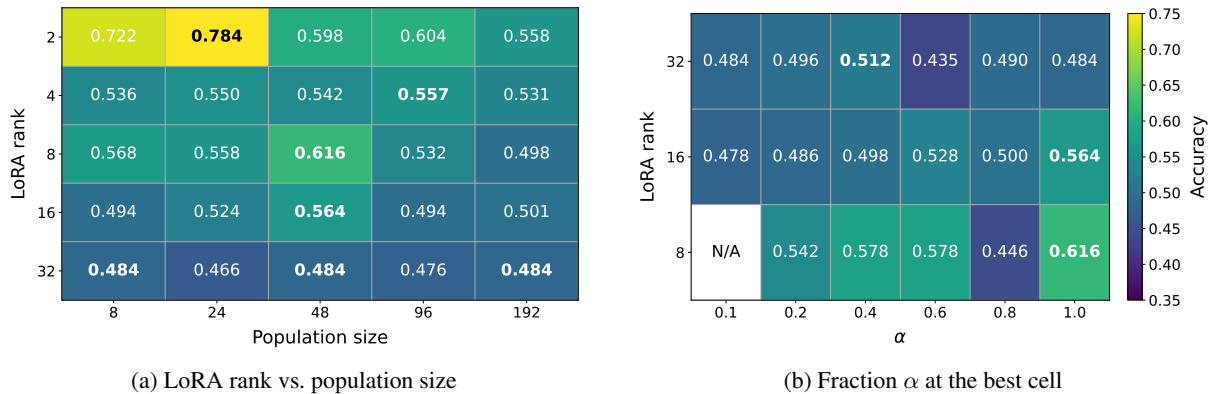
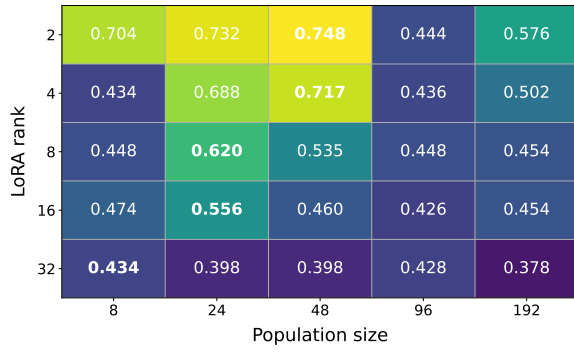
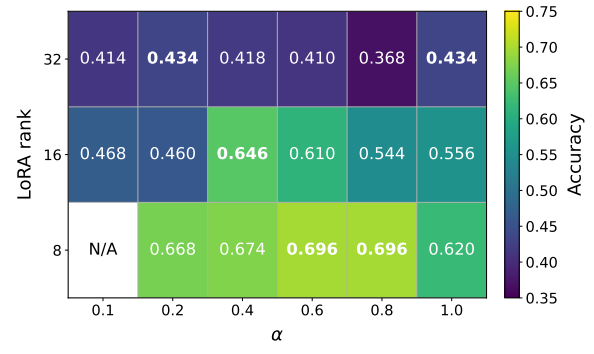


Figure 11: Hyperparameter sensitivity of ESSA on **Qwen2.5-Math-7B** for **PRM800K**. Batch size 300. (a) Accuracy when varying LoRA rank and population size. (b) For each LoRA rank, the population size is fixed to the best value found in (a), while the percentage  $\alpha$  of trainable singular values is varied. This illustrates how ESSA performance depends jointly on adapter rank and the fraction of singular values optimized. The single white cell occurs because for LoRA rank 8 and  $\alpha=0.1$ , rounding down yields zero trainable singular values, so no valid accuracy is reported.



(a) LoRA rank vs. population size



(b) Fraction  $\alpha$  at the best cell

Figure 12: Hyperparameter sensitivity of ESSA on **Qwen2.5-7B** for **PRM800K**. Batch size 300. **(a)** Accuracy when varying LoRA rank and population size. **(b)** For each LoRA rank, the population size is fixed to the best value found in (a), while the percentage  $\alpha$  of trainable singular values is varied. This illustrates how ESSA performance depends jointly on adapter rank and the fraction of singular values optimized. The single white cell occurs because for LoRA rank 8 and  $\alpha = 0.1$ , rounding down yields zero trainable singular values, so no valid accuracy is reported.

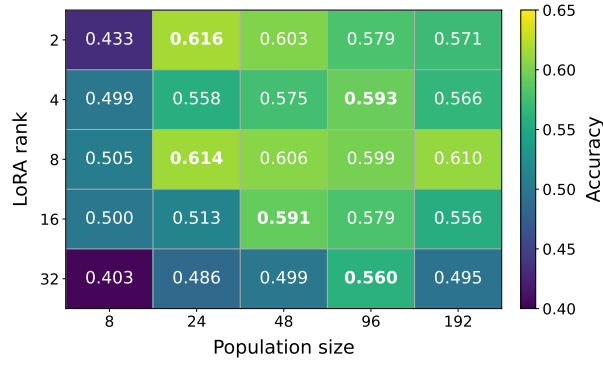


Figure 13: Hyperparameter sensitivity of ESSA on **Llama-3.1-8B** for **IFEval**. Accuracy when varying LoRA rank and population size.

## G Training Dynamics

This section provides the full training-dynamics results supporting the hyperparameter-sensitivity analysis in Section 4.6. The GSM8K curves additionally complement the LoRA-budget comparison in Section 4.1; the PRM800K curves complement the optimizer and subspace analysis in Section 4.2; and the IFEval curves complement the instruction-following results in Section 4.4.

Figures 14–27 show how ESSA validation accuracy evolves when varying the population size at fixed LoRA ranks, the LoRA rank at fixed population sizes, and, where applicable, the fraction  $\alpha$  of singular-value coordinates optimized. These curves complement the static sensitivity heatmaps reported in Appendix F.

### G.0.1 Qwen2.5-Math-7B on GSM8K

These experiments support the sensitivity analysis in Section 4.6 and extend the GSM8K results in Section 4.1 to Qwen2.5-Math-7B. Figures 14, 15, and 16 present the complete training-dynamics study corresponding to the static results in Figure 9. Figure 14 varies the population size at fixed LoRA ranks, Figure 15 varies the LoRA rank at fixed population sizes, and Figure 16 varies  $\alpha$ .

### G.0.2 Qwen2.5-7B on GSM8K

These experiments provide the full sensitivity results for the Qwen2.5-7B/GSM8K setting used in Section 4.1 and support the analysis in Section 4.6. Figures 17, 18, and 19 correspond to the static results in Figure 10. Figure 17 varies the population size at fixed LoRA ranks, Figure 18 varies the LoRA rank at fixed population sizes, and Figure 19 varies  $\alpha$ .

### G.0.3 Qwen2.5-Math-7B on PRM800K

These experiments support the sensitivity analysis in Section 4.6 and complement the PRM800K→MATH500 analysis in Section 4.2. Figures 20, 21, and 22 present the complete training-dynamics study corresponding to the static results in Figure 11. Figure 20 varies the population size at fixed LoRA ranks, Figure 21 varies the LoRA rank at fixed population sizes, and Figure 22 varies  $\alpha$ .

### G.0.4 Qwen2.5-7B on PRM800K

These experiments provide additional training dynamics for the Qwen2.5-7B/PRM800K setting analyzed in Section 4.2 and support the sensitivity

analysis in Section 4.6. Figures 23, 24, and 25 correspond to the static results in Figure 12. Figure 23 varies the population size at fixed LoRA ranks, Figure 24 varies the LoRA rank at fixed population sizes, and Figure 25 varies  $\alpha$ .

### G.0.5 Llama-3.1-8B on IFEval

These experiments provide the full sensitivity results for the Llama-3.1-8B/IFEval setting used in the LoRA-budget comparison in Section 4.1 and the instruction-following analysis in Section 4.4. They also support the hyperparameter analysis in Section 4.6. Figures 26 and 27 correspond to the static results in Figure 13. Figure 26 varies the population size at fixed LoRA ranks, while Figure 27 varies the LoRA rank at fixed population sizes.

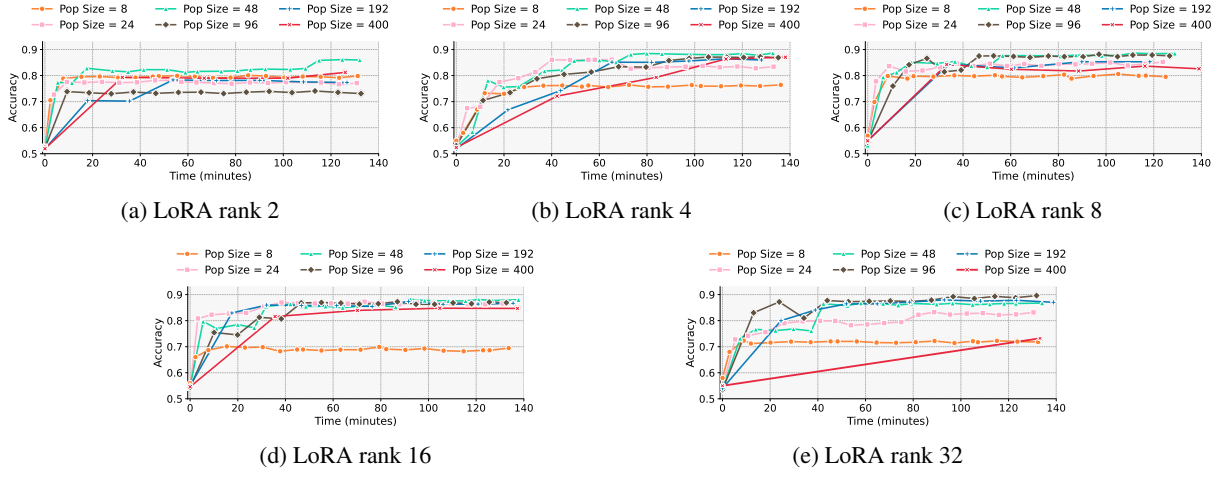


Figure 14: Validation accuracy over time on **GSM8K** with **Qwen2.5-Math-7B** when varying the ESSA population size (8, 24, 48, 96, 192, 400) at fixed LoRA ranks (2, 4, 8, 16, 32). Batch size 100.

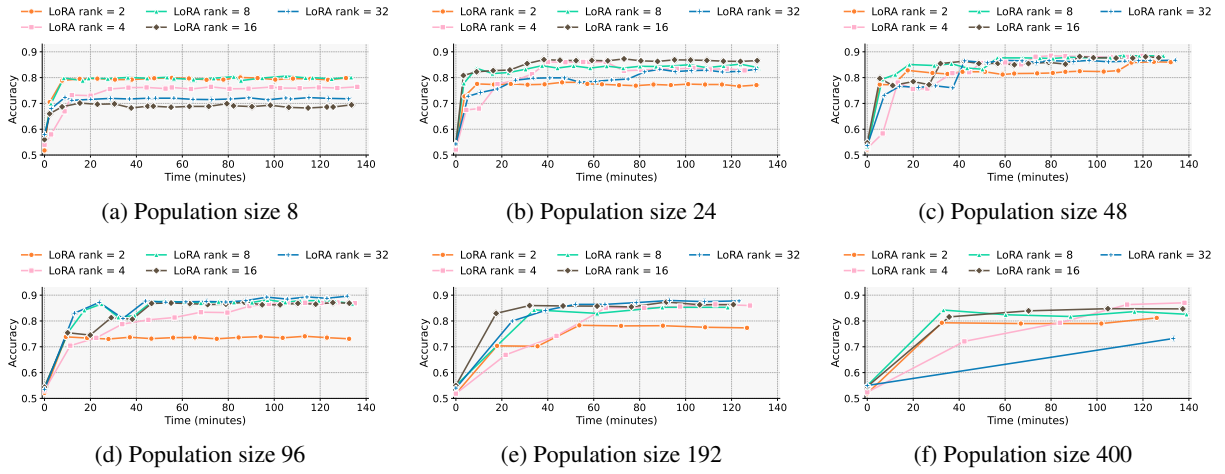


Figure 15: Validation accuracy over time on **GSM8K** with **Qwen2.5-Math-7B** when varying the LoRA rank (2, 4, 8, 16, 32) at fixed population sizes (8, 24, 48, 96, 192, 400). Batch size 100.

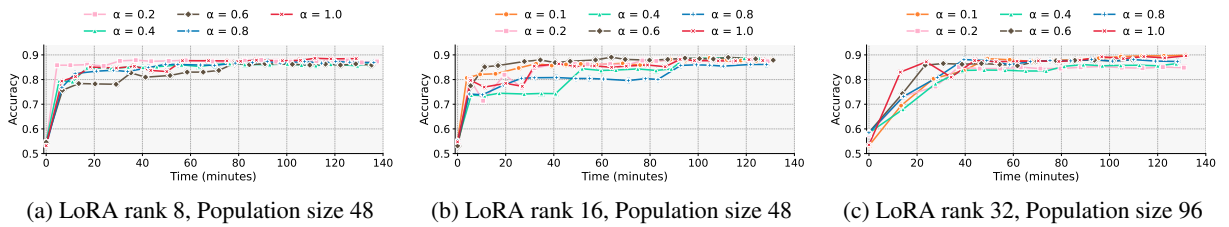


Figure 16: Validation accuracy on **GSM8K** with **Qwen2.5-Math-7B** while varying  $\alpha$ . LoRA rank and population size are fixed to the optimal choices from Figure 9. Batch size 100.

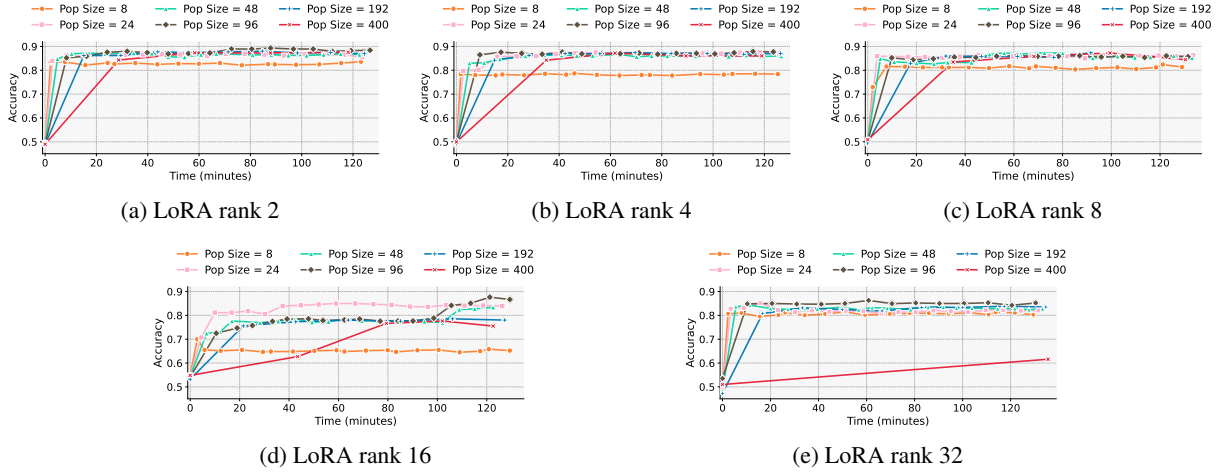


Figure 17: Validation accuracy over time on **GSM8K** with **Qwen2.5-7B** when varying the ESSA population size (8, 24, 48, 96, 192, 400) at fixed LoRA ranks (2, 4, 8, 16, 32). Batch size 100.

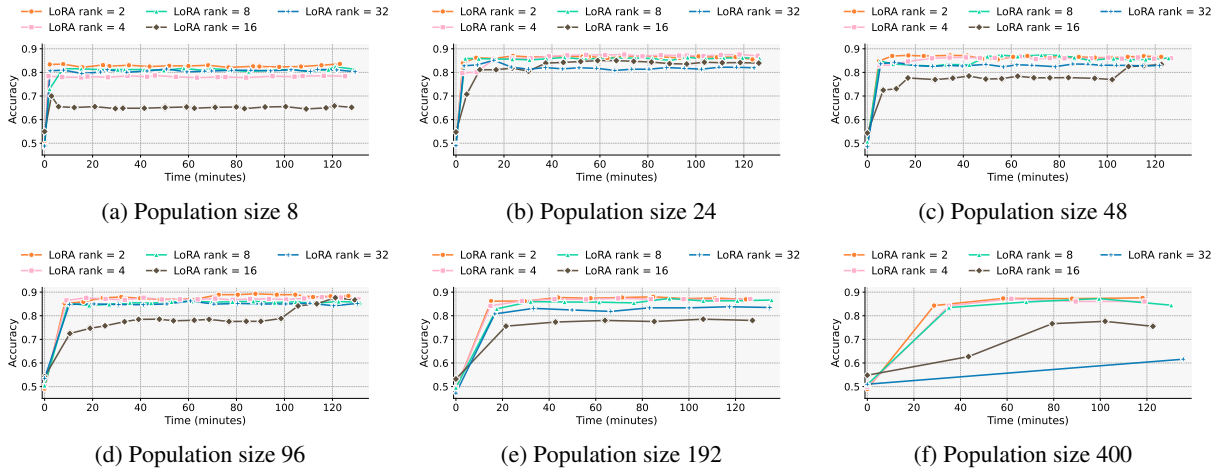


Figure 18: Validation accuracy over time on **GSM8K** with **Qwen2.5-7B** when varying the LoRA rank (2, 4, 8, 16, 32) at fixed population sizes (8, 24, 48, 96, 192, 400). Batch size 100.

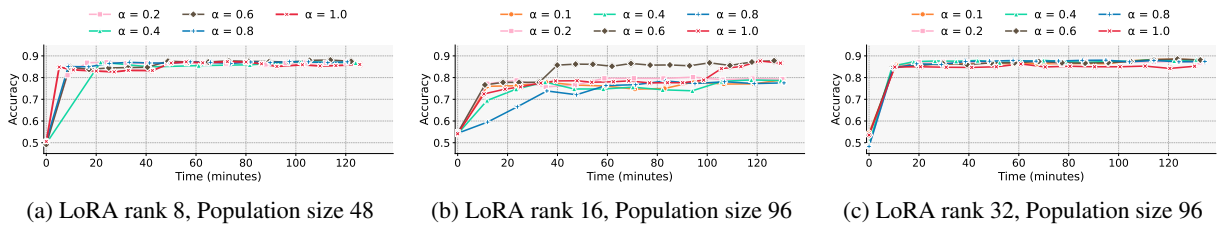


Figure 19: Validation accuracy on **GSM8K** with **Qwen2.5-7B** while varying  $\alpha$ . LoRA rank and population size are fixed to the optimal choices from Figure 10. Batch size 100.

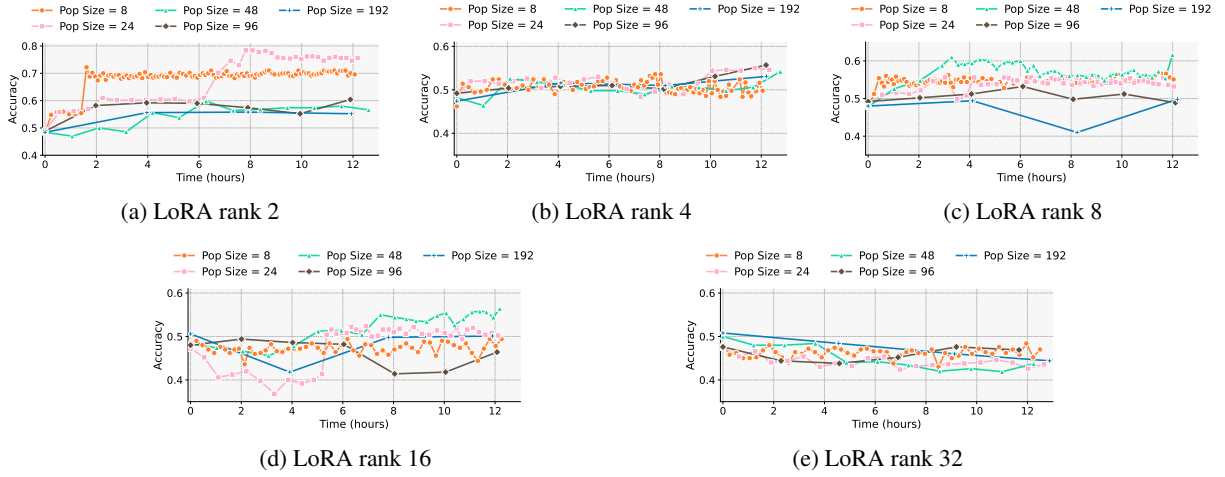


Figure 20: Validation accuracy over time on **PRM800K** with **Qwen2.5-Math-7B** when varying the ESSA population size (8, 24, 48, 96, 192) at fixed LoRA ranks (2, 4, 8, 16, 32). Batch size 300.

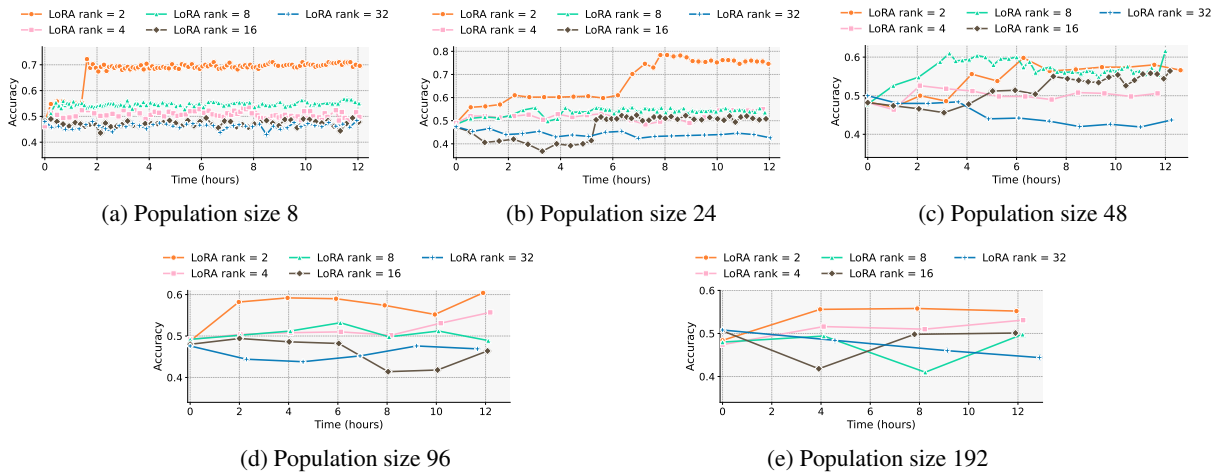


Figure 21: Validation accuracy over time on **PRM800K** with **Qwen2.5-Math-7B** when varying the LoRA rank (2, 4, 8, 16, 32) at fixed population sizes (8, 24, 48, 96, 192). Batch size 300.

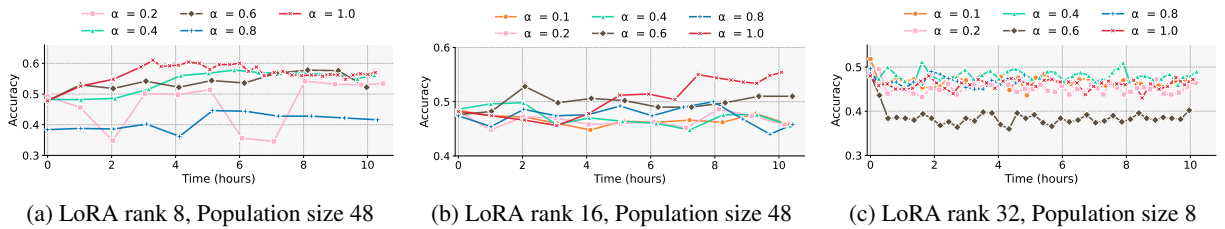


Figure 22: Validation accuracy over time on **PRM800K** with **Qwen2.5-Math-7B** when varying  $\alpha$ . LoRA rank and population size are fixed to the optimal choices from Figure 11. Batch size 300.

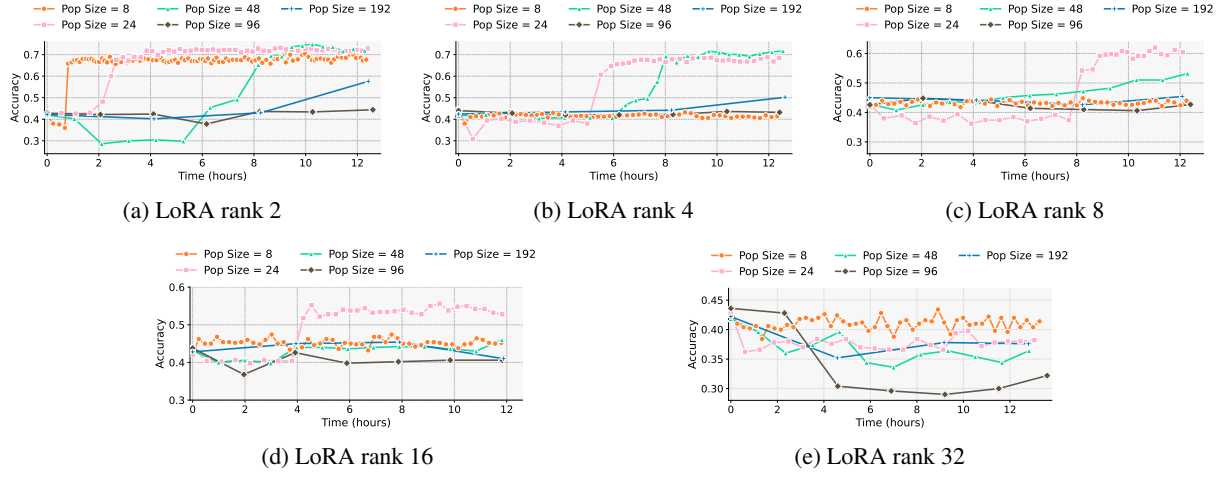


Figure 23: Validation accuracy over time on **PRM800K** with **Qwen2.5-7B** when varying the ESSA population size (8, 24, 48, 96, 192) at fixed LoRA ranks (2, 4, 8, 16, 32). Batch size 300.

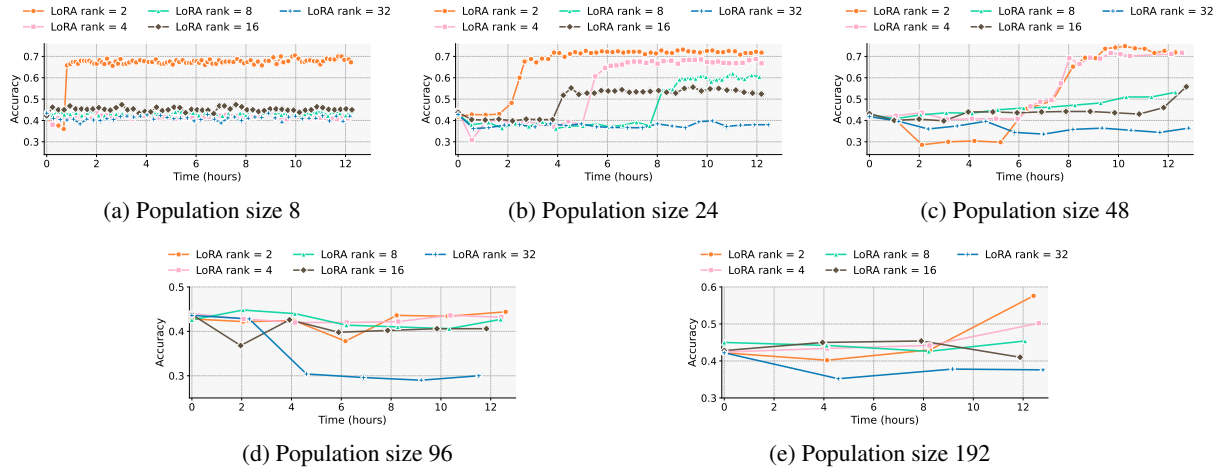


Figure 24: Validation accuracy over time on **PRM800K** with **Qwen2.5-7B** when varying the LoRA rank (2, 4, 8, 16, 32) at fixed population sizes (8, 24, 48, 96, 192). Batch size 300.

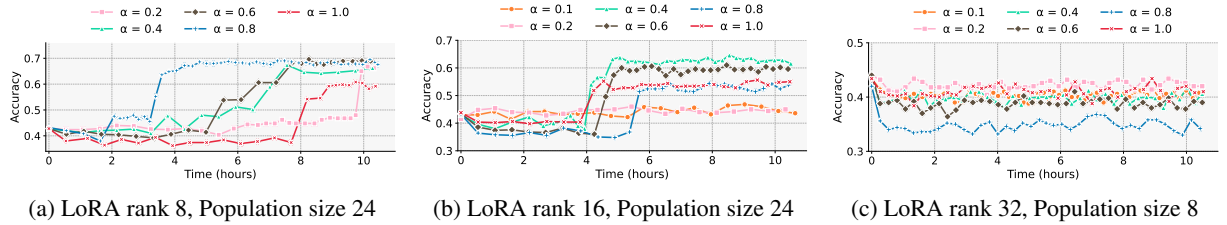


Figure 25: Validation accuracy over time on **PRM800K** with **Qwen2.5-7B** while varying  $\alpha$ . LoRA rank and population size are fixed to the optimal choices from Figure 12. Batch size 300.

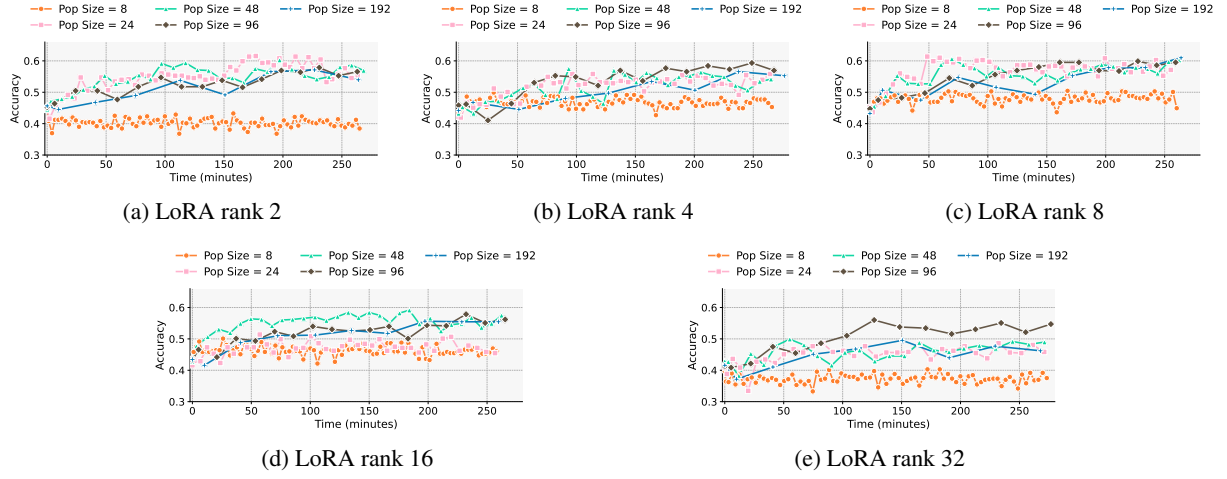


Figure 26: Validation accuracy over time on **IFEval** with **Llama-3.1-8B** when varying the ESSA population size (8, 24, 48, 96, 192) at fixed LoRA ranks (2, 4, 8, 16, 32). Batch size 500,  $\alpha = 1.0$ .

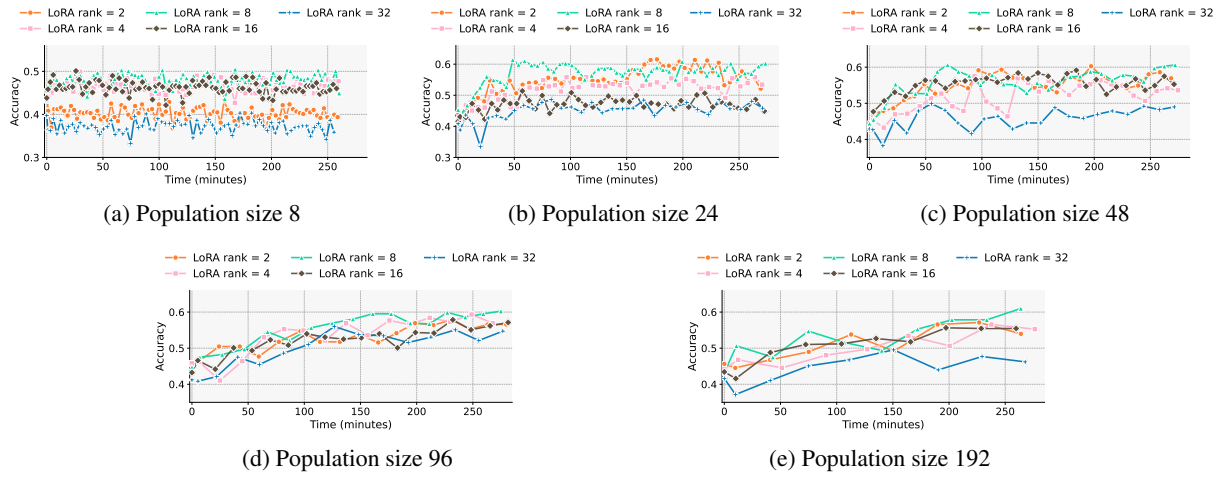


Figure 27: Validation accuracy over time on **IFEval** with **Llama-3.1-8B** when varying the LoRA rank (2, 4, 8, 16, 32) at fixed population sizes (8, 24, 48, 96, 192). Batch size 500,  $\alpha = 1.0$ .

## H Comparison to LoRA-GRPO

This section provides additional ESSA–LoRA-GRPO comparisons supporting the main-text results in Sections 4.1, 4.2, 4.4, and 4.5.

### H.1 School Math

This subsection extends the GSM8K comparison in Section 4.1. Figures 29 and 30 present the results for Qwen2.5-Math-7B and Qwen2.5-7B, respectively. Figure 32 presents the results for Qwen2.5-3B on one GPU. This experiment shows that ESSA’s benefits are not restricted to multi-GPU setups but persist when only one device is available. Figure 31 presents the results for Qwen2.5-7B and Qwen2.5-Math-7B on GSM8K without an SFT warm-start. LoRA matrix  $A$  is initialized using Kaiming initialization, while matrix  $B$  is initialized to zero. In these experiments, runs starting from randomly initialized LoRA adapters achieve lower absolute accuracy.

### H.2 Advanced Math

This subsection complements the PRM800K→MATH500 analysis in Section 4.2 and the large-model advanced-math results in Section 4.5. Figures 33 and 34 present the full results for Qwen2.5-Math-7B and Qwen2.5-7B, respectively. ESSA climbs to high accuracy more rapidly than LoRA-GRPO and often reaches a higher plateau, particularly when the LoRA rank is small or moderate.

### H.3 HelpSteer2 dynamics

This subsection provides the full training dynamics and endpoint diagnostics for the HelpSteer2 experiment summarized in Section 4.4. Figure 35 reports the HelpSteer2 validation reward over time at three LoRA ranks for Llama-3.1-8B. Both ESSA and LoRA-GRPO improve over the SFT initialization. We additionally examine response length, Distinct-2, generation entropy, and relative parameter drift at the endpoints of the training trajectories.

| Diagnostic               | SFT   | ESSA  | LoRA-GRPO |
|--------------------------|-------|-------|-----------|
| Mean response length     | 626   | 745   | 722       |
| Distinct-2               | 0.201 | 0.613 | 0.733     |
| Generation entropy       | 0.642 | 0.735 | 1.973     |
| Relative parameter drift | 0.000 | 2.403 | 4.233     |

Table 8: Endpoint diagnostics on **HelpSteer2** with **Llama-3.1-8B** at LoRA rank 16. ESSA uses population size 42,  $\alpha = 1.0$ , and batch size 100. LoRA-GRPO uses learning rate  $1 \times 10^{-5}$ , global batch size 512, and minibatch size 64. Mean response length increases moderately for both methods (+19% for ESSA and +15% for LoRA-GRPO), providing no indication of disproportionate length growth. Relative parameter drift is  $\|\Delta W_{\text{final}} - \Delta W_{\text{SFT}}\|_F / \|\Delta W_{\text{SFT}}\|_F$  (zero at the SFT checkpoint). ESSA remains closer to the SFT policy in both generation entropy and relative parameter drift, while both methods increase Distinct-2.

### H.4 HH-RLHF dynamics

This subsection provides the full training dynamics for the safety/helpfulness experiment summarized in Section 4.4. Figure 36 shows the in-domain reward over time for Mistral-7B trained on Anthropic HH-RLHF with the OpenAssistant DeBERTa-v3 reward model. ESSA crosses LoRA-GRPO within the first hour and reaches a higher plateau by the end of training.

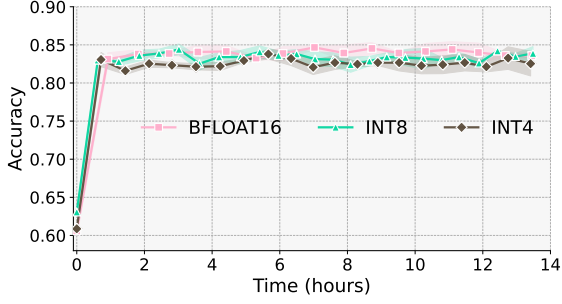


Figure 28: Validation accuracy over training time for **Qwen2.5-32B** on **PRM800K** under different weight precisions (BFLOAT16, INT8, INT4). Settings: LoRA rank 8, pop. 64, batch size 256,  $\alpha = 1.0$ .

## I Precision Analysis

This section provides the full training dynamics for the precision study summarized in Section 4.6.

Figure 28 shows the full training curves of ESSA on PRM800K with Qwen2.5-32B, LoRA rank 8, population size 64,  $\alpha = 1.0$ , and per-candidate batch size 256 under three numerical precisions: BF16, INT8, and INT4. Shaded areas indicate one standard deviation over five independent seeds.

Across all three precisions, accuracy rises quickly within the first hour and then remains stable. Reducing precision from BF16 to INT8 or INT4 produces only a small reduction in final accuracy, consistent with the main-text results in Table 6. In particular, the difference between BF16 and INT4 is 1.1 test-accuracy percentage points. These results show that ESSA can use aggressive weight quantization while retaining similar final performance.

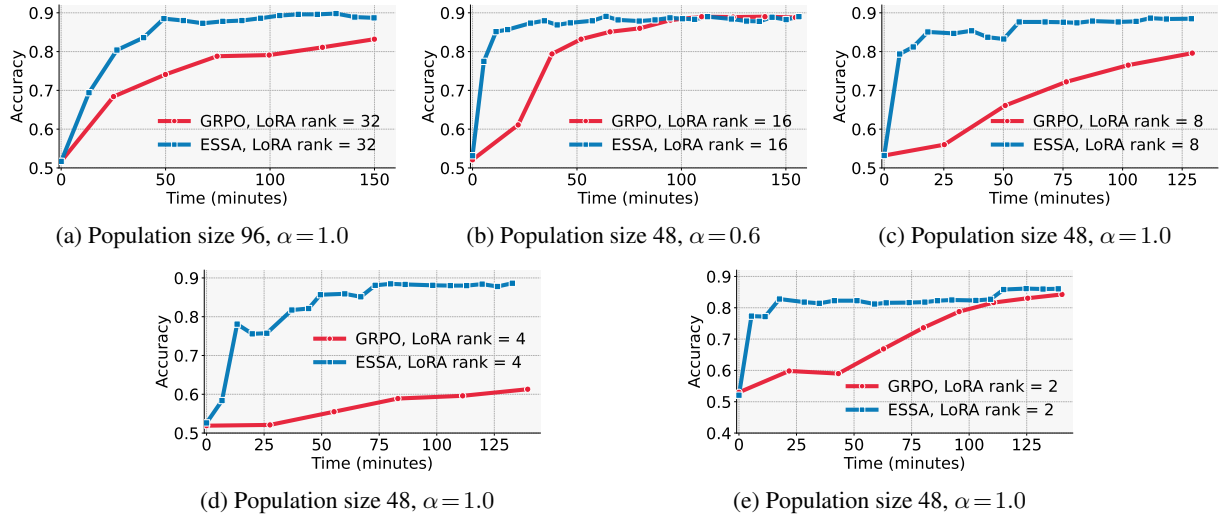


Figure 29: Validation accuracy over time on **GSM8K** with **Qwen2.5-Math-7B**. Panels (a)-(e) correspond to LoRA ranks 32, 16, 8, 4, and 2, respectively. ESSA (blue): batch size 100. LoRA-GRPO (red):  $\text{lr } 1 \times 10^{-5}$ , global batch 512, mini batch 64.

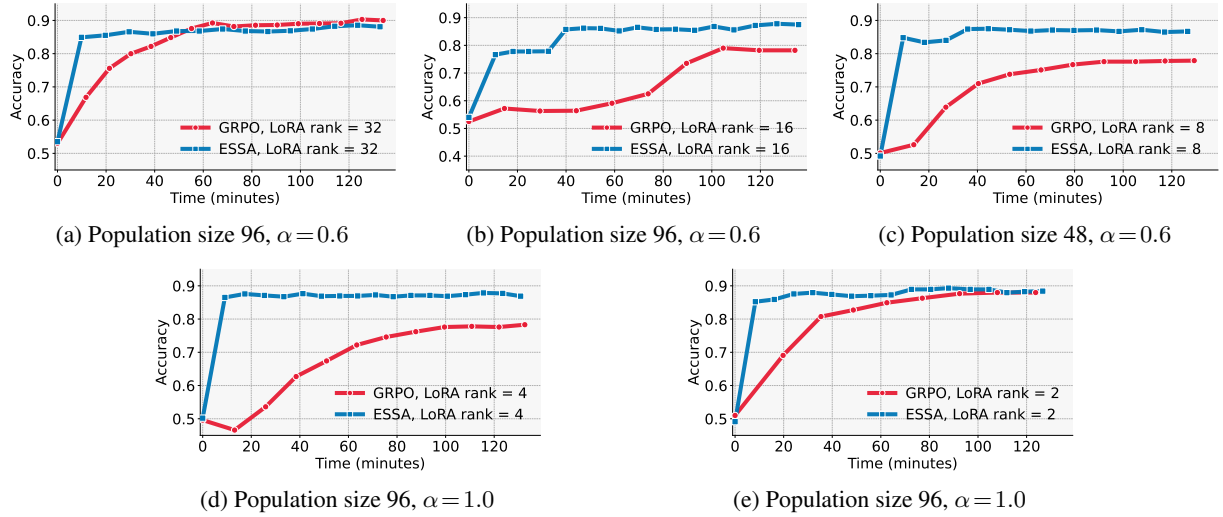
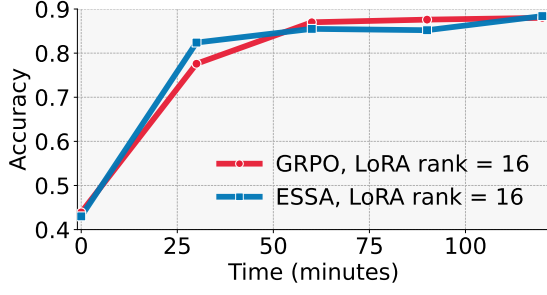
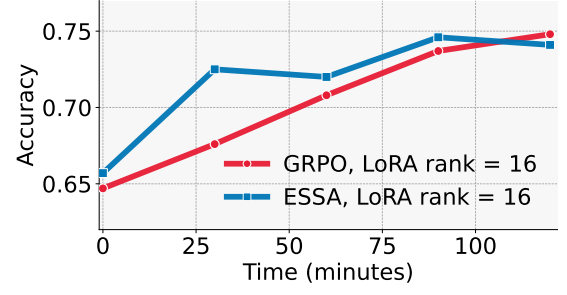


Figure 30: Validation accuracy over time on **GSM8K** with **Qwen2.5-7B**. Panels (a)-(e) correspond to LoRA ranks 32, 16, 8, 4, and 2, respectively. ESSA (blue): batch size 100. LoRA-GRPO (red):  $\text{lr } 1 \times 10^{-5}$ , global batch 512, mini batch 64.



(a) Population size 96,  $\alpha = 1.0$



(b) Population size 96,  $\alpha = 1.0$

Figure 31: Validation accuracy over time on **GSM8K** with **Qwen2.5-7B** (a) and **Qwen2.5-Math-7B** (b) without SFT warm-start. ESSA (blue): batch size 100. LoRA-GRPO (red):  $\text{lr } 1 \times 10^{-5}$ , global batch 512, mini batch 64.

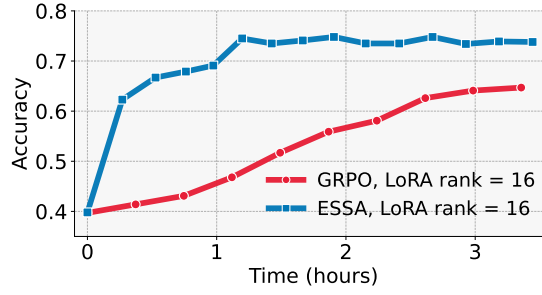
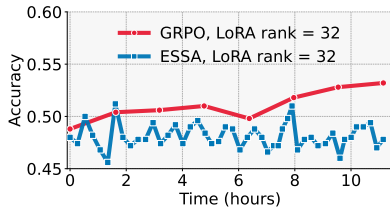
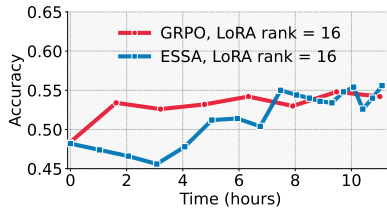


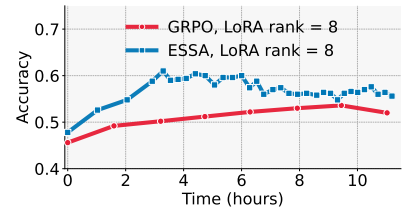
Figure 32: Validation accuracy over time on **GSM8K** with **Qwen2.5-3B** on one GPU. ESSA (blue): population size 48,  $\alpha = 1.0$ , batch size 100. LoRA-GRPO (red):  $\text{lr } 1 \times 10^{-5}$ , global batch 512, mini batch 64.



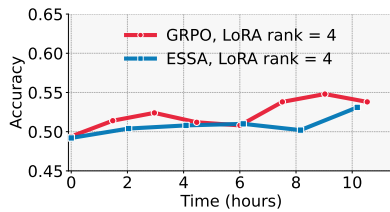
(a) Population size 8,  $\alpha = 0.4$



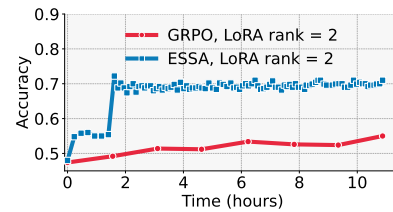
(b) Population size 48,  $\alpha = 1.0$



(c) Population size 48,  $\alpha = 1.0$



(d) Population size 96,  $\alpha = 1.0$



(e) Population size 8,  $\alpha = 1.0$

Figure 33: Validation accuracy over time on **PRM800K** with **Qwen2.5-Math-7B**. Panels (a)-(e) correspond to LoRA ranks 32, 16, 8, 4, and 2, respectively. ESSA (blue): batch size 300. LoRA-GRPO (red):  $\text{lr } 1 \times 10^{-5}$ , global batch 512, mini batch 64.

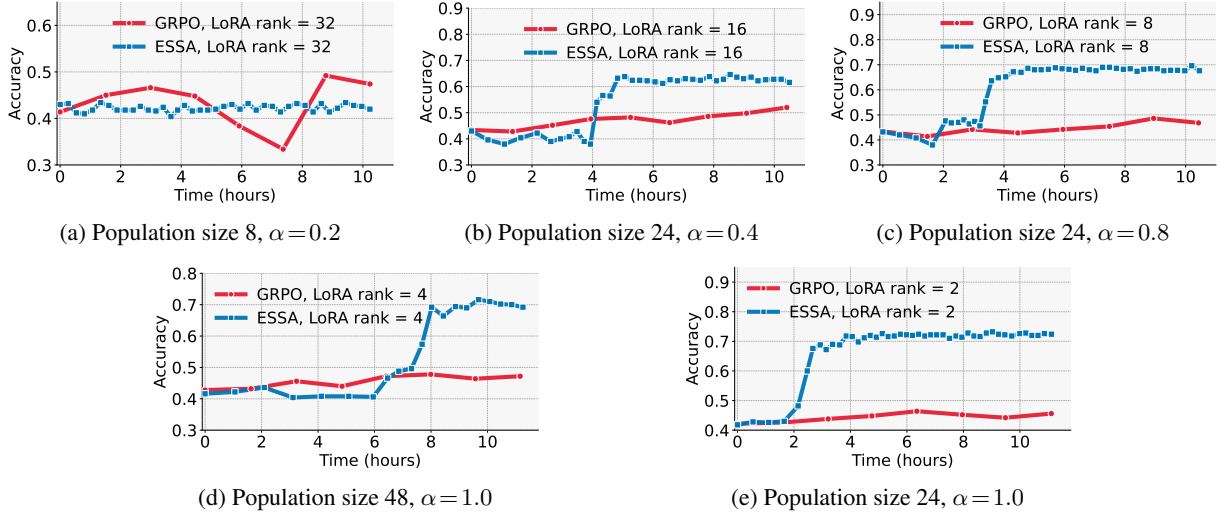


Figure 34: Validation accuracy over time on **PRM800K** with **Qwen2.5-7B**. Panels (a)-(e) correspond to LoRA ranks 32, 16, 8, 4, and 2, respectively. ESSA (blue): batch size 300. LoRA-GRPO (red):  $\text{lr } 1 \times 10^{-5}$ , global batch 512, mini batch 64.

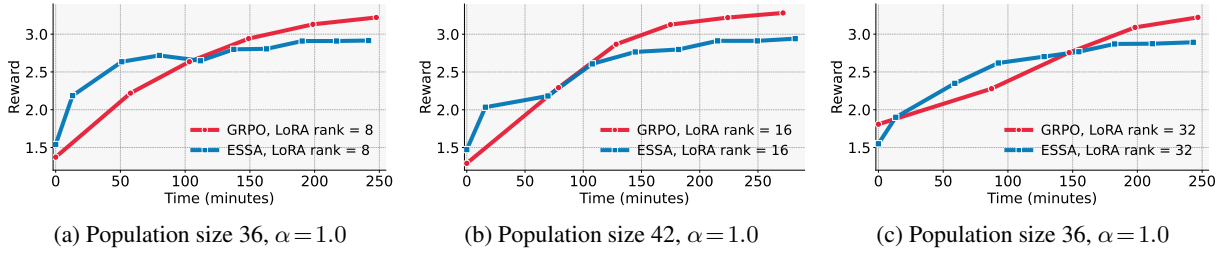


Figure 35: Validation reward over time on **HelpSteer2** with **Llama-3.1-8B**. Panels (a)-(c) correspond to LoRA ranks 8, 16, 32 respectively. ESSA (blue): batch size 100. LoRA-GRPO (red):  $\text{lr } 1 \times 10^{-5}$ , global batch 512, mini batch 64.

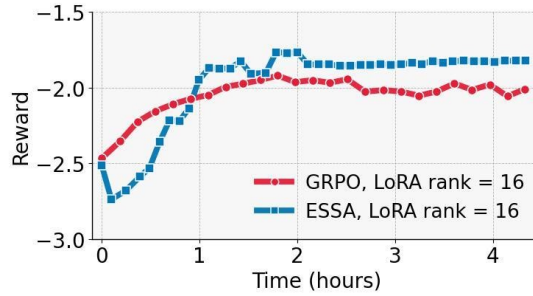


Figure 36: Validation reward over time on **HH-RLHF** with **Mistral-7B**, LoRA rank 16. ESSA (blue): population size 32,  $\alpha = 1.0$ , batch size 100. LoRA-GRPO (red):  $\text{lr } 1 \times 10^{-5}$ , global batch 512, mini batch 64.

| Task            | Optimizer                  | SFT init. | Per-step candidates | Wall-clock | $G = Q$   | Final test        |
|-----------------|----------------------------|-----------|---------------------|------------|-----------|-------------------|
| GSM8K           | CMA-ES (ESSA)              | 0.780     | 96                  | 2 h        | 441,600   | $0.867 \pm 0.010$ |
|                 | Plain $(\mu, \lambda)$ -ES | 0.780     | 96                  | 2 h        | 441,600   | $0.893 \pm 0.006$ |
|                 | ARS                        | 0.780     | 96                  | 2 h        | 441,600   | $0.884 \pm 0.004$ |
|                 | Random search              | 0.780     | 96                  | 2 h        | 441,600   | $0.842 \pm 0.005$ |
|                 | MeZO (SPSA)                | 0.780     | 96                  | 2 h        | 441,600   | $0.826 \pm 0.008$ |
| PRM800K→MATH500 | CMA-ES (ESSA)              | 0.420     | 96                  | 8 h        | 4,608,000 | $0.706 \pm 0.009$ |
|                 | Plain $(\mu, \lambda)$ -ES | 0.420     | 96                  | 8 h        | 4,608,000 | $0.736 \pm 0.008$ |
|                 | ARS                        | 0.420     | 96                  | 8 h        | 4,608,000 | $0.754 \pm 0.006$ |
|                 | Random search              | 0.420     | 96                  | 8 h        | 4,608,000 | $0.636 \pm 0.004$ |
|                 | MeZO (SPSA)                | 0.420     | 96                  | 8 h        | 4,608,000 | $0.472 \pm 0.010$ |

Table 9: Alternative optimizers on Qwen2.5-7B with LoRA rank 16. GSM8K uses population size 96 and batch size 100; PRM800K→MATH500 uses population size 96 and batch size 300.

## J Optimizer and Subspace Controls

This section provides additional results for the optimizer and subspace controls presented in Section 4.2.

### J.1 Optimizer convergence

Table 10 compares validation accuracy at selected optimization steps. CMA-ES improves fastest initially, while ARS achieves the highest late-stage accuracy.

### J.2 Random-basis control

We isolate the contribution of the SFT-derived singular vectors by replacing them with random orthogonal bases. The resulting search space is  $W_{\text{SFT}} + U_{\text{rand}} \text{diag}(\delta) V_{\text{rand}}^T$ , so  $\delta = 0$  exactly recovers the same SFT model. The search dimension, CMA-ES configuration, and 37-step budget are held fixed.

The random basis still improves test accuracy from 0.777 to 0.823 (Table 11), showing that low-dimensional search around a strong SFT policy already provides useful directions. The SFT-SVD basis nevertheless converges faster and reaches higher validation and test accuracy. Thus, the learned basis is beneficial but is not the sole explanation for ESSA’s performance.

### J.3 Sensitivity to SFT strength

ESSA reaches similar final accuracy across all five SFT fractions (Table 12), with no monotonic relationship between SFT fraction and final performance. The method therefore does not require a precisely optimized SFT checkpoint, although it still assumes a warm start containing learned structure.

| Optimizer                  | 21    | 54    | 107   | 160   | 214   |
|----------------------------|-------|-------|-------|-------|-------|
| CMA-ES                     | 0.732 | 0.724 | 0.734 | 0.706 | 0.708 |
| Plain $(\mu, \lambda)$ -ES | 0.702 | 0.732 | 0.714 | 0.694 | 0.696 |
| ARS                        | 0.516 | 0.682 | 0.726 | 0.728 | 0.744 |
| Random search              | 0.556 | 0.572 | 0.602 | 0.652 | 0.646 |
| MeZO (SPSA)                | 0.490 | 0.484 | 0.514 | 0.508 | 0.502 |

Table 10: PRM800K validation accuracy for Qwen2.5-7B with LoRA rank 16, population size 96, and batch size 300.

| Basis             | SFT test | Best val.    | Test @ best val. |
|-------------------|----------|--------------|------------------|
| SFT-SVD           | 0.777    | <b>0.916</b> | <b>0.838</b>     |
| Random orthogonal | 0.777    | 0.884        | 0.823            |

Table 11: Random-basis control on Qwen2.5-7B/GSM8K with LoRA rank 8, population size 96, and a matched 37-step budget.

| SFT fraction | SFT only | SFT→ESSA                            |
|--------------|----------|-------------------------------------|
| 5%           | 0.501    | $0.710 \pm 0.016$                   |
| 25%          | 0.447    | $0.699 \pm 0.023$                   |
| 50%          | 0.395    | <b><math>0.728 \pm 0.024</math></b> |
| 75%          | 0.475    | $0.710 \pm 0.014$                   |
| 100%         | 0.458    | $0.721 \pm 0.023$                   |

Table 12: MATH500 accuracy as a function of the fraction of PRM800K SFT data used to construct the warm start.

## K Experimental Configuration and Implementation Details

**Data splits.** Table 13 summarizes the training, selection-validation, and final-test splits.

**Main experimental configurations.** Table 16 reports the configurations and compute accounting for the reported experiments. The run-level learning-rate sweeps for the main-text LoRA-GRPO comparisons on GSM8K and IFEval are reported in Tables 17 and 18, respectively. The SVD-GRPO sweeps for the optimizer ablation on PRM800K→MATH500 are reported in Table 19. These experiments use the nine-value learning-rate set

$$\mathcal{L} = \{10^{-6}, 5 \times 10^{-6}, 10^{-5}, 5 \times 10^{-5}, 10^{-4}, 5 \times 10^{-4}, 10^{-3}, 5 \times 10^{-3}, 10^{-2}\}.$$

Each generated response is scored once, so the policy-generation count  $G$  equals the reward-query count  $Q$  and is reported jointly as  $G = Q$ . For the SFT-seeded methods, wall-clock budgets exclude the shared SFT stage, whose cost is reported separately in Table 14.

**SFT cost.** SFT is shared by ESSA and the post-SFT LoRA baselines. Full-parameter GRPO in Table 1 is initialized without SFT and is therefore excluded from this shared-cost statement. Table 14 reports the SFT GPU-hours and the initial held-out metric observed before online optimization.

### System prompts.

- **Math reasoning:** Please reason step by step and place your final answer in `\boxed{\}`.
- **Countdown:** You are a helpful assistant.
- **Instruction following:** You are a helpful, honest, and concise assistant.

**ESSA hyperparameter grid.** We evaluate ESSA using task-specific subsets of the hyperparameter values reported in Table 15; not every Cartesian combination is evaluated. The CMA-ES initial step size is fixed at  $\sigma_0 = 0.32$  in all experiments.

**Software.** We train the gradient baselines—LoRA-GRPO, Online DPO, PPO, full-parameter GRPO, and SVD-GRPO—using the `verl` library (Sheng et al., 2024). ESSA uses `vLLM` (Kwon et al.,

2023) for policy inference and `EvoTorch` for CMA-ES. Unless otherwise noted, the experiments use NVIDIA H100-equivalent GPUs.

| Dataset              | SFT train | Online train | Selection val         | Final test         |
|----------------------|-----------|--------------|-----------------------|--------------------|
| GSM8K                | 1.5K      | 5.5K         | 500 from GSM8K train  | GSM8K test (1,319) |
| PRM800K → MATH500    | 3.6K      | 7.4K         | 500 from PRM800K      | MATH500 (500)      |
| IFEval-like → IFEval | 1.7K      | 2.9K         | 500 from source train | IFEval (541)       |
| HelpSteer2           | 19K       | 1.3K         | 500                   | 500                |
| HH-RLHF              | 5K        | 10K          | 1.0K                  | held-out judge set |
| Countdown            | —         | 1.0K         | 1.0K                  | 1.0K               |

Table 13: Training, selection, and final-evaluation splits. Online-train counts exclude selection-validation examples. For GSM8K and IFEval, the official benchmark sets are never used for model or hyperparameter selection. For PRM800K→MATH500, selection is performed on 500 held-out PRM800K problems and MATH500 is used only for final reporting.

| Backbone (task)            | LoRA rank | Reported in               | SFT epochs | SFT GPU-hrs |
|----------------------------|-----------|---------------------------|------------|-------------|
| Qwen2.5-7B (GSM8K)         | 2         | Sec. 4.1; Apps. F and H.1 | 1          | 0.10        |
|                            | 4         |                           | 1          | 0.15        |
|                            | 8         |                           | 1          | 0.16        |
|                            | 16        |                           | 1          | 0.18        |
|                            | 32        |                           | 1          | 0.21        |
| Qwen2.5-Math-7B (GSM8K)    | 2         | Apps. F and H.1           | 1          | 0.10        |
|                            | 4         |                           | 1          | 0.12        |
|                            | 8         |                           | 1          | 0.15        |
|                            | 16        |                           | 1          | 0.16        |
|                            | 32        |                           | 1          | 0.19        |
| Qwen2.5-7B (PRM800K)       | 2         | Sec. 4.2; Apps. F and H.2 | 1          | 0.31        |
|                            | 4         |                           | 1          | 0.35        |
|                            | 8         |                           | 1          | 0.38        |
|                            | 16        |                           | 1          | 0.43        |
|                            | 32        |                           | 1          | 0.48        |
| Qwen2.5-Math-7B (PRM800K)  | 2         | Apps. F and H.2           | 1          | 0.28        |
|                            | 4         |                           | 1          | 0.32        |
|                            | 8         |                           | 1          | 0.35        |
|                            | 16        |                           | 1          | 0.40        |
|                            | 32        |                           | 1          | 0.44        |
| Qwen2.5-32B (PRM800K)      | 8         | Secs. 4.5 and 4.6         | 3          | 9.80        |
|                            | 16        | Sec. 4.3                  | 3          | 11.24       |
| Qwen2.5-72B (PRM800K)      | 4         | Sec. 4.5                  | 3          | 51.97       |
| Llama-3.1-8B (IFEval-like) | 2         | Secs. 4.1 and 4.4; App. F | 1          | 0.35        |
|                            | 4         |                           | 1          | 0.41        |
|                            | 8         |                           | 1          | 0.45        |
|                            | 16        |                           | 1          | 0.50        |
|                            | 32        |                           | 1          | 0.56        |
| Llama-3.1-8B (HelpSteer2)  | 8         | Sec. 4.4; App. H.3        | 3          | 3.38        |
|                            | 16        |                           | 3          | 3.74        |
|                            | 32        |                           | 3          | 4.01        |
| Mistral-7B (HH-RLHF)       | 16        | Sec. 4.4; App. H.4        | 3          | 3.24        |

Table 14: SFT cost for the checkpoints used in the reported experiments. GPU-hours are measured on NVIDIA H100 80GB GPUs and reported separately for each LoRA rank. For every referenced SFT-seeded training curve, the first plotted point is the validation performance of the corresponding SFT checkpoint.

| Parameter                               | Values evaluated                             | Default |
|-----------------------------------------|----------------------------------------------|---------|
| LoRA rank ( $r$ )                       | 2, 4, 8, 16, 32                              | 8       |
| Population size ( $P$ )                 | 8, 24, 32, 36, 42, 48, 64, 96, 128, 192, 400 | 96      |
| Retained fraction ( $\alpha$ )          | 0.1, 0.2, 0.4, 0.6, 0.8, 1.0                 | 1.0     |
| Per-candidate batch size ( $B_E$ )      | 100, 200, 256, 300, 500                      | 100     |
| Weight precision                        | BF16, INT8, INT4                             | BF16    |
| CMA-ES initial step size ( $\sigma_0$ ) | 0.32                                         | 0.32    |

Table 15: ESSA hyperparameter values used or evaluated across experiments. The listed values form the union of task-specific configurations; not every Cartesian combination was evaluated. Experiments use up to 128 NVIDIA H100 80GB GPUs.

| Source     | Model / task                                | Method              | GPUs | Configuration                            | SFT init.     | Wall-clock | $G = Q$   | Reported result       | Prec. |
|------------|---------------------------------------------|---------------------|------|------------------------------------------|---------------|------------|-----------|-----------------------|-------|
| Tab. 1     | Qwen2.5-7B / GSM8K                          | ESSA                | 8    | 16/96/100/1.0                            | 0.723         | 2 h        | 489,600   | 0.867 $\pm$ 0.010     | BF16  |
|            |                                             | LoRA-GRPO           | 8    | 16/1 $\times$ 10 <sup>-4</sup> /512/64/8 | 0.723         | 2 h        | 184,320   | 0.823 $\pm$ 0.012     | BF16  |
|            |                                             | Online DPO          | 8    | 16/5 $\times$ 10 <sup>-5</sup> /512/64/8 | 0.723         | 2 h        | 401,408   | 0.818 $\pm$ 0.001     | BF16  |
|            |                                             | PPO                 | 8    | 16/1 $\times$ 10 <sup>-4</sup> /512/64/1 | 0.723         | 2 h        | 15,360    | 0.791 $\pm$ 0.010     | BF16  |
|            |                                             | Full-parameter GRPO | 8    | -/1 $\times$ 10 <sup>-5</sup> /512/64/8  | -             | 2 h        | 94,208    | 0.870 $\pm$ 0.009     | BF16  |
| Tab. 1     | Llama-3.1-8B / IFEval                       | ESSA                | 8    | 8/24/500/1.0                             | 0.451         | 4 h        | 1,764,000 | 0.555 $\pm$ 0.009     | BF16  |
|            |                                             | LoRA-GRPO           | 8    | 8/5 $\times$ 10 <sup>-5</sup> /512/64/8  | 0.451         | 4 h        | 122,880   | 0.483 $\pm$ 0.002     | BF16  |
|            |                                             | Online DPO          | 8    | 8/1 $\times$ 10 <sup>-5</sup> /512/64/8  | 0.451         | 4 h        | 602,112   | 0.559 $\pm$ 0.008     | BF16  |
|            |                                             | PPO                 | 8    | 8/1 $\times$ 10 <sup>-4</sup> /512/64/1  | 0.451         | 4 h        | 51,200    | 0.559 $\pm$ 0.007     | BF16  |
|            |                                             | Full-parameter GRPO | 8    | -/1 $\times$ 10 <sup>-5</sup> /512/64/8  | -             | 4 h        | 143,360   | 0.692 $\pm$ 0.008     | BF16  |
| Fig. 2     | Qwen2.5-7B / GSM8K                          | ESSA                | 8    | 8/48/100/0.8                             | 0.724         | 2 h        | 201,600   | 0.853 $\pm$ 0.008     | BF16  |
|            |                                             | LoRA-GRPO           | 8    | 8/1 $\times$ 10 <sup>-4</sup> /512/64/8  | 0.724         | 2 h        | 204,800   | 0.847 $\pm$ 0.014     | BF16  |
| Fig. 3     | Qwen2.5-7B / PRM800K $\rightarrow$ MATH500  | ESSA                | 8    | 2/24/300/1.0                             | 0.420         | 8 h        | 1,080,000 | 0.723 $\pm$ 0.005     | BF16  |
|            |                                             | SVD-GRPO            | 8    | 2/1 $\times$ 10 <sup>-2</sup> /512/64/8  | 0.420         | 8 h        | 204,800   | 0.422 $\pm$ 0.008     | BF16  |
|            |                                             | SVD-GRPO            | 8    | 4/1 $\times$ 10 <sup>-2</sup> /512/64/8  | 0.420         | 8 h        | 245,760   | 0.322 $\pm$ 0.012     | BF16  |
|            |                                             | SVD-GRPO            | 8    | 8/1 $\times$ 10 <sup>-2</sup> /512/64/8  | 0.420         | 8 h        | 245,760   | 0.488 $\pm$ 0.001     | BF16  |
|            |                                             | SVD-GRPO            | 8    | 16/1 $\times$ 10 <sup>-2</sup> /512/64/8 | 0.420         | 8 h        | 245,760   | 0.510 $\pm$ 0.002     | BF16  |
| Fig. 4     | Qwen2.5-32B / PRM800K $\rightarrow$ MATH500 | ESSA                | 16   | 16/128/256/1.0                           | 0.631         | 206.17 min | 622,592   | 0.835 target          | BF16  |
|            |                                             | LoRA-GRPO           | 16   | 16/1 $\times$ 10 <sup>-5</sup> /512/64/8 | 0.631         | 394.36 min | 204,800   | 0.835 target          | BF16  |
| Fig. 4     | Qwen2.5-32B / PRM800K $\rightarrow$ MATH500 | ESSA                | 32   | 16/128/256/1.0                           | 0.631         | 110.71 min | 622,592   | 0.835 target          | BF16  |
|            |                                             | LoRA-GRPO           | 32   | 16/1 $\times$ 10 <sup>-5</sup> /512/64/8 | 0.631         | 210.29 min | 204,800   | 0.835 target          | BF16  |
| Fig. 4     | Qwen2.5-32B / PRM800K $\rightarrow$ MATH500 | ESSA                | 64   | 16/128/256/1.0                           | 0.631         | 30.62 min  | 622,592   | 0.835 target          | BF16  |
|            |                                             | LoRA-GRPO           | 64   | 16/1 $\times$ 10 <sup>-5</sup> /512/64/8 | 0.631         | 177.82 min | 204,800   | 0.835 target          | BF16  |
| Fig. 4     | Qwen2.5-32B / PRM800K $\rightarrow$ MATH500 | ESSA                | 128  | 16/128/256/1.0                           | 0.631         | 19.72 min  | 622,592   | 0.835 target          | BF16  |
|            |                                             | LoRA-GRPO           | 128  | 16/1 $\times$ 10 <sup>-5</sup> /512/64/8 | 0.631         | 153.04 min | 204,800   | 0.835 target          | BF16  |
| Fig. 5     | Llama-3.1-8B / IFEval                       | ESSA                | 8    | 8/24/500/1.0                             | 0.451         | 4 h        | 1,764,000 | 0.555 $\pm$ 0.009     | BF16  |
|            |                                             | LoRA-GRPO           | 8    | 8/5 $\times$ 10 <sup>-5</sup> /512/64/8  | 0.451         | 4 h        | 122,880   | 0.483 $\pm$ 0.002     | BF16  |
| Tab. 4     | Qwen2.5-32B / advanced math                 | ESSA                | 8    | 8/48/256/1.0                             | 0.631         | 8 h        | 614,400   | 0.440                 | BF16  |
|            |                                             | LoRA-GRPO           | 8    | 16/1 $\times$ 10 <sup>-5</sup> /512/64/8 | 0.631         | 8 h        | 204,800   | 0.425                 | BF16  |
| Tab. 5     | Qwen2.5-72B / advanced math                 | ESSA                | 8    | 4/64/256/1.0                             | 0.678         | 20 h       | 327,680   | 0.447                 | INT4  |
|            |                                             | LoRA-GRPO           | 8    | 16/1 $\times$ 10 <sup>-5</sup> /512/64/8 | 0.678         | 20 h       | 61,440    | 0.445                 | BF16  |
| Tab. 6     | Qwen2.5-32B / MATH500 precision             | ESSA (BF16)         | 8    | 8/64/256/1.0                             | 0.631         | 8 h        | 147,456   | 0.845 $\pm$ 0.004     | BF16  |
|            |                                             | ESSA (INT8)         | 8    | 8/64/256/1.0                             | 0.631         | 8 h        | 229,376   | 0.841 $\pm$ 0.006     | INT8  |
|            |                                             | ESSA (INT4)         | 8    | 8/64/256/1.0                             | 0.631         | 8 h        | 196,608   | 0.834 $\pm$ 0.006     | INT4  |
| Fig. 35(a) | Llama-3.1-8B / HelpSteer2                   | ESSA                | 8    | 8/36/100/1.0                             | 1.815 reward  | 4 h        | 144,000   | 2.909 reward          | BF16  |
|            |                                             | LoRA-GRPO           | 8    | 8/1 $\times$ 10 <sup>-5</sup> /512/64/8  | 1.815 reward  | 4 h        | 163,840   | 3.130 reward          | BF16  |
| Fig. 35(b) | Llama-3.1-8B / HelpSteer2                   | ESSA                | 8    | 16/42/100/1.0                            | 1.815 reward  | 4 h        | 126,000   | 2.913 reward          | BF16  |
|            |                                             | LoRA-GRPO           | 8    | 16/1 $\times$ 10 <sup>-5</sup> /512/64/8 | 1.815 reward  | 4 h        | 163,840   | 3.220 reward          | BF16  |
| Fig. 35(c) | Llama-3.1-8B / HelpSteer2                   | ESSA                | 8    | 32/36/100/1.0                            | 1.815 reward  | 4 h        | 126,000   | 2.874 reward          | BF16  |
|            |                                             | LoRA-GRPO           | 8    | 32/1 $\times$ 10 <sup>-5</sup> /512/64/8 | 1.815 reward  | 4 h        | 122,880   | 3.089 reward          | BF16  |
| Fig. 36    | Mistral-7B / HH-RLHF                        | ESSA                | 8    | 16/32/100/1.0                            | -2.500 reward | 4 h        | 729,600   | 12/76/12              | BF16  |
|            |                                             | LoRA-GRPO           | 8    | 16/1 $\times$ 10 <sup>-5</sup> /512/64/8 | -2.500 reward | 4 h        | 294,912   | pairwise result above | BF16  |

Table 16: Configurations and accounting for the reported main experiments. Within each experiment, ESSA is listed first, followed by the corresponding baselines. ESSA configurations are reported as  $r/P/B_E/\alpha$ , where  $r$  is the LoRA rank,  $P$  is the population size,  $B_E$  is the per-candidate batch size, and  $\alpha$  is the fraction of singular-value coordinates optimized. Gradient-based configurations are reported as  $r/\eta/B_G/b/n$ , where  $\eta$  is the learning rate,  $B_G$  is the global batch size,  $b$  is the minibatch size, and  $n$  is the number of responses sampled per prompt. The displayed  $\eta$  is the learning rate of the reported run; all nine sweep values are listed in the run-level tables.  $G$  denotes the number of training policy generations and  $Q$  the number of reward queries; because each generated response is scored once,  $G = Q$ . The GPUs column reports the number of GPUs used: all non-scaling experiments use eight GPUs, while the Figure 4 rows report 16, 32, 64, and 128 GPUs. SFT init. denotes the result of the shared supervised-fine-tuning initialization, and Prec. denotes the numerical precision. For HelpSteer2, the reported result is the validation reward at the last checkpoint within the four-hour cutoff, and  $G = Q$  is truncated at the same checkpoint. For Mistral-7B/HH-RLHF, the reported result is the GPT-4o pairwise outcome in the order ESSA win / tie / LoRA-GRPO win.

| Experiment         | Method / rank       | LR                 | SFT initial test | Wall-clock | Final test        |
|--------------------|---------------------|--------------------|------------------|------------|-------------------|
| Qwen2.5-7B / GSM8K | LoRA-GRPO, $r = 8$  | $10^{-6}$          | 0.724            | 2 h        | $0.746 \pm 0.004$ |
|                    |                     | $5 \times 10^{-6}$ | 0.724            | 2 h        | $0.755 \pm 0.003$ |
|                    |                     | $10^{-5}$          | 0.724            | 2 h        | $0.832 \pm 0.010$ |
|                    |                     | $5 \times 10^{-5}$ | 0.724            | 2 h        | $0.839 \pm 0.008$ |
|                    |                     | $10^{-4}$          | 0.724            | 2 h        | $0.847 \pm 0.014$ |
|                    |                     | $5 \times 10^{-4}$ | 0.724            | 2 h        | $0.844 \pm 0.012$ |
|                    |                     | $10^{-3}$          | 0.724            | 2 h        | $0.842 \pm 0.010$ |
|                    |                     | $5 \times 10^{-3}$ | 0.724            | 2 h        | $0.000 \pm 0.000$ |
| Qwen2.5-7B / GSM8K | LoRA-GRPO, $r = 16$ | $10^{-6}$          | 0.723            | 2 h        | $0.728 \pm 0.006$ |
|                    |                     | $5 \times 10^{-6}$ | 0.723            | 2 h        | $0.737 \pm 0.012$ |
|                    |                     | $10^{-5}$          | 0.723            | 2 h        | $0.789 \pm 0.005$ |
|                    |                     | $5 \times 10^{-5}$ | 0.723            | 2 h        | $0.801 \pm 0.006$ |
|                    |                     | $10^{-4}$          | 0.723            | 2 h        | $0.823 \pm 0.012$ |
|                    |                     | $5 \times 10^{-4}$ | 0.723            | 2 h        | $0.805 \pm 0.015$ |
|                    |                     | $10^{-3}$          | 0.723            | 2 h        | $0.814 \pm 0.007$ |
|                    |                     | $5 \times 10^{-3}$ | 0.723            | 2 h        | $0.000 \pm 0.000$ |
|                    |                     | $10^{-2}$          | 0.723            | 2 h        | $0.000 \pm 0.000$ |

Table 17: LoRA-GRPO learning-rate sweep results for Qwen2.5-7B on GSM8K. For each run, we report the test-set performance of the checkpoint from the final validation evaluation within the stated wall-clock budget.

| Experiment            | Method / rank      | LR                 | SFT initial test | Wall-clock | Final test        |
|-----------------------|--------------------|--------------------|------------------|------------|-------------------|
| Llama-3.1-8B / IFEval | LoRA-GRPO, $r = 8$ | $10^{-6}$          | 0.451            | 4 h        | $0.433 \pm 0.011$ |
|                       |                    | $5 \times 10^{-6}$ | 0.451            | 4 h        | $0.446 \pm 0.009$ |
|                       |                    | $10^{-5}$          | 0.451            | 4 h        | $0.476 \pm 0.002$ |
|                       |                    | $5 \times 10^{-5}$ | 0.451            | 4 h        | $0.483 \pm 0.002$ |
|                       |                    | $10^{-4}$          | 0.451            | 4 h        | $0.447 \pm 0.012$ |
|                       |                    | $5 \times 10^{-4}$ | 0.451            | 4 h        | $0.458 \pm 0.008$ |
|                       |                    | $10^{-3}$          | 0.451            | 4 h        | $0.421 \pm 0.015$ |
|                       |                    | $5 \times 10^{-3}$ | 0.451            | 4 h        | $0.000 \pm 0.000$ |
|                       |                    | $10^{-2}$          | 0.451            | 4 h        | $0.000 \pm 0.000$ |

Table 18: LoRA-GRPO learning-rate sweep results for Llama-3.1-8B on IFEval. For each run, we report the test-set performance of the checkpoint from the final validation evaluation within the stated wall-clock budget.

| Experiment                   | Method / rank      | LR                 | SFT initial test | Wall-clock | Final test        |
|------------------------------|--------------------|--------------------|------------------|------------|-------------------|
| Qwen2.5-7B / PRM800K→MATH500 | SVD-GRPO, $r = 2$  | $10^{-6}$          | 0.398            | 8 h        | $0.000 \pm 0.000$ |
|                              |                    | $5 \times 10^{-6}$ | 0.398            | 8 h        | $0.000 \pm 0.000$ |
|                              |                    | $10^{-5}$          | 0.398            | 8 h        | $0.000 \pm 0.000$ |
|                              |                    | $5 \times 10^{-5}$ | 0.398            | 8 h        | $0.308 \pm 0.016$ |
|                              |                    | $10^{-4}$          | 0.398            | 8 h        | $0.239 \pm 0.012$ |
|                              |                    | $5 \times 10^{-4}$ | 0.398            | 8 h        | $0.413 \pm 0.005$ |
|                              |                    | $10^{-3}$          | 0.398            | 8 h        | $0.411 \pm 0.004$ |
|                              |                    | $5 \times 10^{-3}$ | 0.398            | 8 h        | $0.418 \pm 0.004$ |
|                              |                    | $10^{-2}$          | 0.398            | 8 h        | $0.422 \pm 0.008$ |
| Qwen2.5-7B / PRM800K→MATH500 | SVD-GRPO, $r = 4$  | $10^{-6}$          | 0.405            | 8 h        | $0.000 \pm 0.000$ |
|                              |                    | $5 \times 10^{-6}$ | 0.405            | 8 h        | $0.000 \pm 0.000$ |
|                              |                    | $10^{-5}$          | 0.405            | 8 h        | $0.000 \pm 0.000$ |
|                              |                    | $5 \times 10^{-5}$ | 0.405            | 8 h        | $0.000 \pm 0.000$ |
|                              |                    | $10^{-4}$          | 0.405            | 8 h        | $0.000 \pm 0.000$ |
|                              |                    | $5 \times 10^{-4}$ | 0.405            | 8 h        | $0.000 \pm 0.000$ |
|                              |                    | $10^{-3}$          | 0.405            | 8 h        | $0.300 \pm 0.014$ |
|                              |                    | $5 \times 10^{-3}$ | 0.405            | 8 h        | $0.278 \pm 0.018$ |
|                              |                    | $10^{-2}$          | 0.405            | 8 h        | $0.322 \pm 0.012$ |
| Qwen2.5-7B / PRM800K→MATH500 | SVD-GRPO, $r = 8$  | $10^{-6}$          | 0.401            | 8 h        | $0.444 \pm 0.002$ |
|                              |                    | $5 \times 10^{-6}$ | 0.401            | 8 h        | $0.406 \pm 0.004$ |
|                              |                    | $10^{-5}$          | 0.401            | 8 h        | $0.235 \pm 0.018$ |
|                              |                    | $5 \times 10^{-5}$ | 0.401            | 8 h        | $0.432 \pm 0.005$ |
|                              |                    | $10^{-4}$          | 0.401            | 8 h        | $0.346 \pm 0.009$ |
|                              |                    | $5 \times 10^{-4}$ | 0.401            | 8 h        | $0.427 \pm 0.002$ |
|                              |                    | $10^{-3}$          | 0.401            | 8 h        | $0.452 \pm 0.002$ |
|                              |                    | $5 \times 10^{-3}$ | 0.401            | 8 h        | $0.402 \pm 0.006$ |
|                              |                    | $10^{-2}$          | 0.401            | 8 h        | $0.488 \pm 0.001$ |
| Qwen2.5-7B / PRM800K→MATH500 | SVD-GRPO, $r = 16$ | $10^{-6}$          | 0.422            | 8 h        | $0.472 \pm 0.002$ |
|                              |                    | $5 \times 10^{-6}$ | 0.422            | 8 h        | $0.472 \pm 0.004$ |
|                              |                    | $10^{-5}$          | 0.422            | 8 h        | $0.484 \pm 0.001$ |
|                              |                    | $5 \times 10^{-5}$ | 0.422            | 8 h        | $0.492 \pm 0.003$ |
|                              |                    | $10^{-4}$          | 0.422            | 8 h        | $0.480 \pm 0.002$ |
|                              |                    | $5 \times 10^{-4}$ | 0.422            | 8 h        | $0.480 \pm 0.014$ |
|                              |                    | $10^{-3}$          | 0.422            | 8 h        | $0.492 \pm 0.001$ |
|                              |                    | $5 \times 10^{-3}$ | 0.422            | 8 h        | $0.484 \pm 0.006$ |
|                              |                    | $10^{-2}$          | 0.422            | 8 h        | $0.510 \pm 0.002$ |

Table 19: SVD-GRPO learning-rate sweep results for Qwen2.5-7B on PRM800K→MATH500. Each LoRA rank is evaluated using all nine learning rates. For each run, we report the test-set performance of the checkpoint from the final validation evaluation within the stated wall-clock budget.