CFA: Mitigating Forgetting in RL Post-Training with Content-Free Anchors

Anonymous authors
Paper under double-blind review

Abstract

RL post-training is deployed under the assumption that it forgets little of what it was not trained on. We show this is incomplete: across Qwen3-VL at 2B, 4B, and 8B scale, single-domain RL post-training silently collapses held-out capability, with worst-decile drops up to −67% hidden behind suite-wide averages that stay flat only because unrelated benchmarks improve on the same run. We introduce the *worst-k curve* to expose this tail. Every existing defense, parameter-importance penalties, replay, KL toward a reference, assumes the anchor set used to constrain the model must represent the real capabilities being protected. Protecting a handful of anticipated domains this way is expensive; protecting *general* capability this way is infeasible, since it requires anchoring on essentially the model's entire pretraining distribution. We find this assumption is not load-bearing: anchors drawn from pure random noise, with no connection to any capability, match and at modest budgets outperform a curated real coreset spanning five capability dimensions. We instantiate this as CFA, a Fisher-weighted Maximum Mean Discrepancy penalty between current and reference next-token distributions, which we derive as a Gauss-Newton generalization of Elastic Weight Consolidation whose finite-sample guarantees hold regardless of anchor content. Across all nine domain-scale combinations tested, CFA holds worst-decile forgetting to −0.3% to −9.1%, against −8.1% to −67.4% for the best baseline, while matching or exceeding in-domain gain, using no real, licensed, or curated data at all.

1 Introduction

RL post-training such as RLHF, RLVR, and their variants is the default recipe for adapting large language and vision-language models to new capabilities, deployed under an optimistic assumption that finetuned models forget little of what they were not trained on. This is supported by a growing body of work arguing that on-policy RL stays close to the base policy in KL divergence (Shenfeld et al., 2026), that reward variance acts as an implicit regularizer (Lai et al., 2025), and that RL's mode-seeking dynamics preserve more of the prior task than SFT's mode-covering dynamics (Chen et al., 2026). Taken together, this literature supports a rule-of-thumb that one could RL-tune a pre-trained model on a new domain, and ship it without worrying about overwriting knowledge, or catastrophic forgetting. Recent works such as Luo et al. (2026) complicate this picture, arguing that the benchmarks behind the optimistic literature are ones the pretraining corpus likely already covers. They build a benchmark of recent, diverse tasks (MRCL) and show RL post-training *does* forget across a *sequence* of tasks. Their proposed fix, CPO, is a replay-free parameter-movement regularizer that penalizes the magnitude of parameter change without accounting for curvature or Fisher information. In this work, we revisit the most common deployment pattern involving post-training on a single new domain, which prior literature suggests should show little to no forgetting in terms of suite-wide averages. We RL-tune Qwen3-VL on three domains from MRCL (Luo et al., 2026) and evaluate on a common multimodal suite, and find the opposite: grounding and document-understanding accuracy collapses. For instance, CountBenchQA falls by up to 67% relative to base, with localized drops up to 80% while STEM

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

and reasoning benchmarks *improve on the same runs*. A suite-wide average cancels the two and suggests minimal forgetting, but this is an incomplete picture (Section 5.2).

Motivation. Every existing defense against this kind of forgetting including parameter-importance penalties (EWC; Kirkpatrick et al., 2017), replay buffers (Rolnick et al., 2019b), KL regularization against reference prompts, and their variants share an assumption about the anchor set: the reference inputs used to constrain model behavior must represent the capabilities being protected, whether as a stored buffer of prior-task examples, a set of inducing points, or domain-relevant prompts. This assumption is easy to state and hard to satisfy, for two reasons of increasing severity. The shallow problem is cost: sourcing, licensing, and maintaining real data for even a handful of known capabilities is expensive, and gets more so as the number of protected domains grows. The deeper problem is scope. We do not want to protect a handful of domains we happened to anticipate, but the *general* capability, and the full breadth of what the base model can do, because an RL run optimized for a single narrow reward can degrade competencies far outside its training distribution, and there is no way to know *a priori* which of a model's many pretrained capabilities a given run will damage. Once the goal is general capability protection rather than protection of a few chosen domains, a real anchor set stops being merely expensive and becomes infeasible by construction: the only anchor set that can be guaranteed to cover "whatever capability turns out to be at risk" is one that spans essentially the model's entire pretraining distribution. No practitioner curates, licenses, and replays pretraining-scale data on every RL fine-tuning run, and the natural workaround is anchoring on the evaluation suite itself is unusable, since it invites contamination and reward hacking. What is needed is a way to constrain a model's outputs relative to its own prior behavior at the scale that general capability protection demands, without requiring real data at that scale.

Overview. We test whether curating real anchors is load-bearing, and find it is not: content-free anchors built from random token sequences with no informative structure or connection to any capability lag a curated real coreset at small budgets, and match or *outperform* it once given a modest anchor budget (Section 5). We instantiate this as Content-Free Anchoring (CFA): a frozen reference policy predicts behavior on a fixed anchor set, and the current policy is penalized for diverging from it via a Fisher-weighted Maximum Mean Discrepancy computed in the model's own token-embedding space, rather than discrete KL over the raw vocabulary. We hypothesize that KL might be a sub-optimal choice for anchoring behavior since KL treats every token as an equidistant, unrelated class and penalizes a benign paraphrase as heavily as semantic corruption. To expose the tail that suite-wide averages hide, we report the *worst-k curve*: mean relative change over the k worst-affected out-of-domain benchmarks. Across all nine domain-scale combinations tested, CFA keeps worst-decile change under -10%, using data-free synthetic anchors that require no real data or license and can be generated on the fly. Figure 1 previews the resulting trade-off on Navigation at 2B: CFA attains both higher in-domain accuracy and higher retention than every baseline family we compare against (including model-souping, parameter importance penalties, and traditional KL regularization). Before scaling to Qwen3-VL, we validate both design questions, whether functional anchoring is necessary at all, and whether anchor content matters, in a minimal, fully enumerable testbed (Section 4).

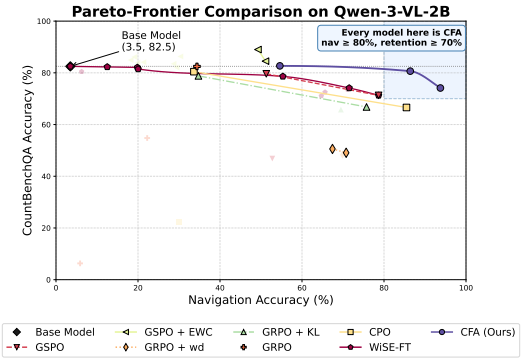

Figure 1: **Pareto frontiers, in-domain vs. OOD retention.** Navigation accuracy vs. CountBenchQA accuracy, Qwen3-VL-2B. CFA reaches $\geq 80\%$ Navigation while retaining $\geq 70\%$ CountBenchQA, a region unavailable to other methods.

Contributions. In summary, we make the following contributions in this work:

- **A hidden failure mode in the default RL post-training recipe.** Single-domain RL post-training silently collapses held-out capability while suite-wide av-

erages hide it. We propose studying the *worst-k curve*, an empirical analogue of Conditional Value-at-Risk over out-of-domain benchmarks, as a replacement metric (Section 3.2).

- **A first-principles connection to EWC, and where it stops holding.** We unify EWC and CFA under a single Gauss-Newton template, establish finite-sample concentration guarantees for CFA's curvature estimates regardless of anchor content, give a matching deterministic guarantee for curated coresets, and show the two together predict the observed budget-dependent crossover, including an explicit trust radius for when EWC's second-order truncation remains valid (Section 3.5).
- **Evidence that curated real anchor sets are unnecessary, not essential.** Content-free random-token anchors match or outperform a curated real coreset once given a modest budget. We thus propose a data-free regularizer that substantially outperforms GRPO and GRPO+KL (Shao et al., 2024), GSPO (Zheng et al., 2025), EWC (Kirkpatrick et al., 2017), and CPO (Luo et al., 2026) on worst-k forgetting at equal or better in-domain gain, holding worst-case forgetting under -10% in every domain and scale tested. (Section 5).

2 Related Work

Forgetting in RL post-training. Catastrophic forgetting is a long-standing continual-learning concern (McCloskey & Cohen, 1989; French, 1999; Kirkpatrick et al., 2017) that has recently reappeared in LLM post-training (Luo et al., 2023; Li et al., 2024; Scialom et al., 2022). A separate thread reports RL as more conservative than SFT at matched target-task performance, attributing this to low-KL on-policy optimization (Shenfeld et al., 2026), on-policy data (Chen et al., 2026), or RL's mode-seeking dynamics (Lai et al., 2025) making average-case claims over target tasks plausibly already seen during pretraining. We ask whether that average conceals selective, worst-case collapse when the training domain is genuinely unseen, in the spirit of Luo et al. (2026)'s critique. The closest comparison is Luo et al. (2026), which studies *inter-task* forgetting across a sequence of multimodal tasks and proposes CPO, a replay-free parameter-movement regularizer. Our setting is *domain-to-general*: a single RL run with no task sequence, so a capability can be damaged with no evident connection to the training domain. CPO constrains parameter movement as a proxy for behavioral drift; CFA instead evaluates the current and reference policies directly on anchor inputs, targeting functional equivalence on held-out anchors while learning the new task. Related mitigations include proactive pre-hardening against anticipated shifts (FRPO; Sabbaghi et al., 2026) and sequential preference regularization requiring stored historical query buffers (COPR; Zhang et al., 2025). Two further paradigms are widely deployed in practice: mixing in pretraining or instruction-token loss during policy gradient updates (PPO-ptx; Ouyang et al., 2022), which is effective but requires complex data streaming pipelines; and post-hoc weight interpolation between the finetuned and reference parameters (Wortsman et al., 2022; Ilharco et al., 2023; Yadav et al., 2023; Yu et al., 2024), which is training-free but assumes linear weight-space paths preserve non-linear representation manifolds, often trading away target-task gains to recover general skills. Most of these, like EWC rely on a second-order parameter-space truncation and we show that requiring domain-specific anchor data is an artifact of that truncation, since exact Gauss-Newton functional evaluation concentrates independently of anchor semantics.

Classical continual learning and data-free distillation. Parameter-importance penalties (Kirkpatrick et al., 2017; Zenke et al., 2017), gradient-projection methods (Lopez-Paz & Ranzato, 2017; Chaudhry et al., 2018; Farajtabar et al., 2020; Saha et al., 2021), and replay of stored or generated examples (Rolnick et al., 2019a; Buzzega et al., 2020; Scialom et al., 2022) all protect prior behavior in some form; dark-experience replay (Buzzega et al., 2020), which stores soft targets from an earlier reference model, is closest in spirit to our use of reference outputs. Data-free distillation similarly transfers a teacher's behavior without original training data, but by *optimizing* synthetic inputs to be informative matching activation statistics (Lopes et al., 2017), targeting class-conditional outputs (Nayak et al., 2019), or inverting a classifier (Yin et al., 2020). All of these assume the input side carries information about the protected distribution, an assumption that breaks down for a de-

ployed model whose prior capabilities are broad, proprietary, and impossible to enumerate in advance. CFA's anchors are neither drawn from nor optimized toward any such distribution: an anchor supplies only a *location* at which to evaluate, and it is the frozen reference policy's output at that location that carries the behavior signal (Section 5, Appendix A.3).

3 Content-Free Anchoring: Method and Theory

3.1 Setup and Notation

We consider an autoregressive vision-language policy π_θ over a vocabulary of size V, with token-embedding matrix $E \in \mathbb{R}^{V \times d}$. Given a context $c = (I, x_{<t})$, $\pi_\theta(\cdot \mid c) \in \Delta^{V-1}$ denotes the next-token distribution. We write θ^* for the frozen reference checkpoint (the model before RL post-training) and $\pi^*(\cdot \mid c) := \pi_{\theta^*}(\cdot \mid c)$ for its induced distribution. RL post-training optimizes θ from θ^* via a policy-gradient objective $\mathcal{L}_{\mathrm{RL}}$ (GRPO or GSPO) on a single target domain. Section 3.5 shows that forgetting tracks π_θ's induced behavior rather than the magnitude of parameter movement; we therefore constrain the output distribution directly rather than through a proxy on θ. Doing so requires (i) a reference distribution to compare against and (ii) a set of inputs on which to make that comparison. The reference is simply the frozen pretrained policy π^*, already resident in memory for standard KL-regularized RL; Section 3.3 addresses the harder question of what the input set should be. We formalize the resulting mechanism as Content-Free Anchoring (CFA): a fixed anchor set is evaluated once under π^*, and π_θ is penalized for diverging from the cached reference output.

3.2 Measuring Forgetting: The Worst-k Curve

Let $b \in \mathcal{B}$ index out-of-domain benchmarks and $\Delta_b = (s_b^{\text{tuned}} - s_b^{\text{base}})/s_b^{\text{base}}$ be relative change in score. The standard reported quantity is the suite-wide mean $\bar{\Delta} = \frac{1}{|\mathcal{B}|}\sum_b \Delta_b$. We instead report the *worst-k curve*

$$\Delta_k \;=\; \frac{1}{k}\sum_{b \in \mathcal{B}_k} \Delta_b \;=\; \min_{\mathcal{S} \subseteq \mathcal{B},\, |\mathcal{S}|=k} \frac{1}{k}\sum_{b \in \mathcal{S}} \Delta_b, \qquad \mathcal{B}_k = \text{the } k \text{ benchmarks with smallest } \Delta_b, \tag{1}$$

swept over $k = 1, \ldots, |\mathcal{B}|$, so $\Delta_{|\mathcal{B}|} = \bar{\Delta}$ recovers the usual average at the right endpoint. Because $\mathcal{B}_k$ dynamically selects the k most degraded benchmarks per method, the worst-k curve is an empirical extreme-value risk measure directly analogous to Conditional Value-at-Risk (CVaR): the left endpoint ($k = 1$) diagnoses worst-case localized collapse, while the curvature and slope measure how concentrated that collapse is across capabilities. The selected sets are nested, $\mathcal{B}_1 \subset \mathcal{B}_2 \subset \cdots \subset \mathcal{B}_{|\mathcal{B}|} = \mathcal{B}$, so Δ_k is non-decreasing in k by construction. Additional information about our metric of study and the motivation behind the same is provided in Appendix B.

3.3 Anchor Set Construction

We construct two anchor sets of fixed cardinality N: a curated set of real image-context pairs, and a content-free set with no curation at all.

Real anchors. The candidate pool is assembled exclusively from *training* splits of public datasets spanning five capability dimensions, so no evaluation benchmark can leak into the regularizer. A size-N subset is selected by greedy k-center traversal over a frozen visual encoder, giving a 2-approximation to the optimal N-point covering radius (Gonzalez, 1985); full pool and algorithm details are in Appendix F. Because the traversal is greedy, the sets used across budgets form a single nested sequence with non-increasing covering radius rather than independent draws. This is a *non*-i.i.d. construction, which is why the real anchor set requires its own guarantee (Proposition 7) rather than the i.i.d. argument covering content-free anchors.

Content-free anchors. These consist of synthetic token sequences drawn i.i.d. and uniformly from the base vocabulary, at sequence length $T = 2048$, with no prompt engineering

(full generation details in Appendix F). Because this is an exact i.i.d. draw from a fixed distribution, the concentration guarantees of Section 3.5 apply directly.

3.4 CFA Objective

Let $\phi(v) := e_v/\sqrt{d}$ be the feature map induced by the model's own token-embedding matrix, and $K := \frac{1}{d}EE^\top$. Given an anchor set $\{c^{(i)}\}_{i=1}^N$ of fixed contexts specified at construction time, we evaluate the frozen reference policy in a single teacher-forced pass once before training. Rather than caching dense categorical distributions, we cache only the d-dimensional expected feature vectors $\mu_t^{*(i)} := \frac{1}{\sqrt{d}}E^\top\pi^*(\cdot \mid c_{<t}^{(i)})$ and scalar weights $\tilde{w}_t^{(i)}$, requiring only $\mathcal{O}(NTd)$ storage. During training we penalize the active policy for diverging from the cached reference embeddings:

$$\mathcal{L}_{\text{anchor}} = \frac{\lambda}{BT}\sum_{i=1}^{B}\sum_{t=1}^{T}\tilde{w}_t^{(i)} \cdot \text{MMD}_\phi^2\Big(\pi_\theta(\cdot \mid c_{<t}^{(i)}),\, \pi^*(\cdot \mid c_{<t}^{(i)})\Big), \tag{2}$$

where $\text{MMD}_\phi^2(\pi_\theta, \pi^*) := \|\mathbb{E}_{v\sim\pi_\theta}[\phi(v)] - \mu_t^*\|_2^2 = (\pi_\theta - \pi^*)^\top K(\pi_\theta - \pi^*)$ is computable in closed form, $\mathcal{O}(Vd)$ per position, with no sampling and B is a mini-batch size we sample from the total anchor pool. The weight $\tilde{w}_t^{(i)}$ is normalized reference varentropy, $w_t = \text{Var}_{v\sim\pi^*}[\log\pi^*(v \mid c_{<t})]$, which is maximal at intermediate entropy where the reference distributes mass over a small set of live candidate tokens (decision forks) and vanishes at both near-uniform and near-deterministic extremes. The total training objective is $\mathcal{L} = \mathcal{L}_{\text{RL}} + \mathcal{L}_{\text{anchor}}$, adding only $\sim 15\%$ wall-clock overhead per step. CFA requires one frozen reference copy and a one-time forward pass over the anchor which we could do in the very first step, and reuse those cached references during training. We need one more teacher-forced forward pass on the policy model during training to measure drift against pre-trained behavior (see Algorithm 2 for exact details).

Naming convention. We designate CFA (Content-Free Anchoring) for Eq. 2 with Fisher-weighted MMD_ϕ^2 on Gaussian synthetic anchors, and denote the curated real-data anchor baseline as RDA (Real-Data Anchoring). Ablations that change the divergence or weighting (Fisher-KL, Unweighted-KL) are discussed in Section 5.3 and Appendix C.

3.5 Theoretical Grounding: CFA as a Generalized Fisher Penalty

Every defense in Section 1 including EWC, KL, replay shares an assumption about the anchor set that traces to a shared mathematical template. Near the reference policy, any twice-differentiable divergence $D(\pi^*\|\pi)$ with $D(\pi^*\|\pi^*) = 0$ reduces to a quadratic form $\frac{1}{2}\Delta\pi^\top M_D\Delta\pi$ on the simplex tangent space (Proposition 1): discrete KL and CFA's embedding MMD are the same template, differing only in the metric M_D. Discrete KL induces the Fisher-Rao metric $M_{\text{KL}} = \text{diag}(1/\pi^*)$ (Proposition 2); CFA's MMD is already exactly quadratic, with $M_{\text{MMD}} = 2K$ everywhere (Proposition 3). Pulling back through the parameter Jacobian J_π makes this precise: KL regularization toward the reference is, to leading order, exactly EWC's Fisher-information penalty (Corollary 1), while CFA is the same Gauss-Newton template with the Fisher-Rao metric replaced by K and an adaptive varentropy reweighting (Corollary 2). EWC and CFA are thus the same recipe with a different metric. The template itself is only a local approximation, valid near θ^*, with no built-in signal for when that stops holding. We make this checkable rather than qualitative via an explicit trust radius ρ^* bounding the quadratic form's relative error (Proposition 4); tracking $\|\Delta\theta_t\|_2$ along real RL training shows it escapes ρ^* within a few hundred gradient steps (Appendix D.3), so a curvature matrix computed once and frozen, as in EWC, goes stale for most of a run. CFA sidesteps this rather than managing it: Eq. 2 evaluates π_θ exactly at every step and never forms or truncates a curvature matrix.

Removing the Taylor truncation avoids the staleness problem, but invites an obvious objection: without a frozen curvature matrix to blame, surely the estimate is only trustworthy if the *particular* finite batch of anchors drawn happens to be representative of the capabilities being protected. Does the same requirement simply resurface one level down, in the

sampling of anchors rather than in the approximation itself? We note that this is not true. The curvature CFA estimates from a finite anchor set concentrates around the same fixed target regardless of what the anchors contain: both the scalar anchor loss and the curvature matrix $G_{\text{emb}}(\nu)$ (Eq. 10) converge to their population values at an $N^{-1/2}$ rate, for *any* i.i.d. anchor distribution ν (Propositions 5, 6). Crucially, the constants governing that rate do not depend on whether the anchors are semantically meaningful (Corollary 3): random noise concentrates exactly as fast as a carefully curated anchor set. An anchor set therefore needs to supply coverage density, not relevance. Curated real anchors buy coverage differently: our greedy k-center traversal is deterministic, not i.i.d., so its error is instead bounded by covering radius (Proposition 7), with no $N^{-1/2}$ rate. Real anchors drive this error down fast but plateau at a bias floor set by what the curated pool can represent; content-free anchors pay a slower $N^{-1/2}$ cost but are unconfined to any manifold, so their bias floor can sit lower. Whenever it does, a finite crossover budget $N_\times$ must exist beyond which content-free anchors reach strictly lower error (Corollary 4). Full statements and proofs are in Appendix D.

4 Validating the Approach in a Controlled Testbed

Before scaling to Qwen3-VL, we resolve two design questions in a minimal, fully enumerable testbed: is functional anchoring against a reference necessary at all, and must the anchor set contain authentic prior-task data? We pretrain a 3-layer MLP on FashionMNIST (Task A), then post-train with GRPO on ParityMNIST (Task B); unmitigated GRPO reaches near-ceiling Task-B accuracy while Task-A collapses from 88.7% to 16.7%, reproducing the single-domain collapse studied at scale.

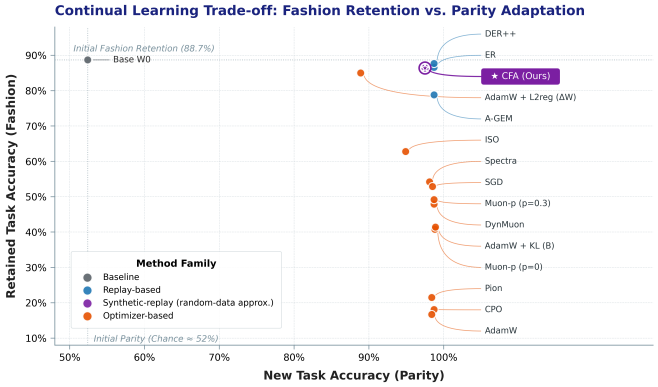

Figure 2: Fashion retention vs. Parity acquisition.

We sweep twelve optimizer- and parameter-space interventions spanning preconditioning (AdamW (Loshchilov & Hutter, 2017), SGD), spectral shaping (Muon (Jordan et al., 2024; Liu et al., 2025) and its variants (Muon^p Dong & Sawin, 2026, DynMuon Wu et al., 2026), SPECTRA (Jiang et al., 2026), Pion (Fan et al., 2026), ISO (Zhu et al., 2026)), and movement penalties (L_2, CPO (Luo et al., 2026)). None resolve forgetting: the best retains only 47.9-54.2% Fashion accuracy against an 88.7% base, since constraining *how* an update moves through weight space does not constrain the induced output distribution, which is a highly non-linear function of the parameters. Against these, real-data replay and content-free Gaussian anchoring (matching CFA's mechanism, Eq. 2) both retain Task-A accuracy near base (86.6% and 86.4% respectively), while learning the new task to near perfection. What governs retention is whether the objective evaluates functional behavior against a reference, not whether real data is cached, or restricting parameter movement. See Appendix A for the full experimental setup and further discussion.

5 Experiments and Results

5.1 Experimental Setup

Models. We post-train Qwen3-VL-Instruct at 2B, 4B and 8B scale on three domains drawn from MRCL's (Luo et al., 2026) task pool: Navigation, MedBookVQA, and Puzzle. Each run independently fine-tunes one base checkpoint on a single domain for 600 steps sweeping over both the learning rate, and regularizer weight (if present).

Evaluation suite. We use the common multimodal suite from Luo et al. (2026): STEM (MMMU-Pro, MathVista, MathVision, MathVerse), general VQA (RealWorldQA, MM-Star), chart/document understanding (DocVQA, CharXiv-DQ, CharXiv-RQ), and grounding (CountBenchQA). For evaluation, we sample trajectories stochastically with temperature $\tau = 0.7$, nucleus sampling with top-$p = 0.8$, top-$k = 20$, and max tokens 16384, and compare against our own reproduced base-model eval under identical settings.

Weight decay. Luo et al. (2026) use weight decay throughout, which independently pulls weights toward the origin and toward forgetting. Removing it improves retention modestly; we ablate both configurations and report the no-weight-decay setting as our default.

Baselines. We compare plain policy-gradient training with and without weight decay (GRPO wd$\in \{0, 0.1\}$, GSPO wd$= 0$); a domain-prompt KL penalty (GRPO, $\beta \in \{0.01, 0.005, 0.001\}$); parameter-importance regularization (GSPO + EWC with real curated anchor set; Kirkpatrick et al., 2017); full-parameter SFT and LoRA on the same domain; and CPO (Luo et al., 2026), adapted to a single-domain run by partitioning each training trajectory into k=3 equal segments and applying CPO's parameter-movement penalty between consecutive segments. Against these we report RDA and CFA at the anchor budgets of Section 3.3, and, on Navigation specifically, the full Fisher-KL/Unweighted-KL ablation grid across budgets $N \in \{320, 640, 1280, 2560\}$.

Reporting criterion. LoRA, SFT and CPO do not always learn the target domain to a comparable degree as RL-based methods, so comparing worst-k forgetting across methods that differ this much in whether they learned the task conflates two failure modes. We therefore report, for each domain, in-domain gain alongside the worst-k curve, and exclude from the headline forgetting comparison any method whose in-domain gain falls below a certain threshold achieved by RL baselines on that domain. Complete metrics with standard deviations appear in Appendix C.

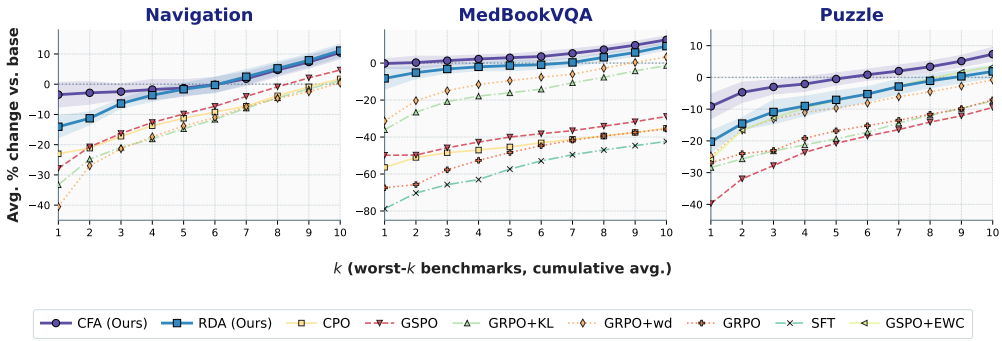

Figure 3: **Qwen3-VL-2B.** Worst-k forgetting curves on Navigation, MedBookVQA, and Puzzle. Each panel plots the cumulative average % change relative to the base model over the k most-affected out-of-domain benchmarks, alongside RDA and CFA at $N = 1280$. The right endpoint, $k = 10$, is the suite-wide average that is conventionally reported on its own; the gap between it and the left endpoint is the forgetting the average conceals. CFA substantially reduces tail degradation across all qualifying baselines in all three domains.

5.2 Results

We report worst-k forgetting curves across all domains and scales. On Navigation (Figure 3), CFA cuts worst-decile change from -40.5% (GRPO, wd$= 0.1$) and -23.0% (CPO, the strongest qualifying baseline) to -3.5% (CFA) and -14.1% (RDA), while also posting the highest in-domain gain of any method compared. WiSE-FT (Appendix C.1) mitigates forgetting relative to unregularized RL but remains suboptimal, trading away target-task plasticity while still conceding a larger tail than CFA. On MedBookVQA, unmitigated GRPO (wd$= 0$) records the most severe collapse observed in this study: -67.4% worst-decile change despite 99.8% in-domain accuracy; CFA holds this to -0.3%. On Puzzle, CFA again reduces worst-k forgetting relative to every baseline that learns the domain comparably well, cutting worst-decile change from -24.8% (GRPO, wd$= 0.1$) to -9.1%. CPO's

apparent CountBenchQA retention comes with only +39.0 points of in-domain gain (final 62.8%), well below the > 80% reached by qualifying RL methods (see Table 4).

At 4B (Figure 4), the ordering is unchanged: RDA and CFA outperform every qualifying baseline, with CFA reaching −0.1%/−3.7%/−5.6% on Navigation/MedBookVQA/Puzzle. GSPO + EWC fails badly here (−94.1% on MedBookVQA, DocVQA collapsing 64.4 → 3.8), evidence that a curvature matrix frozen at θ^* can actively mislead once training leaves its valid region. At 8B (Figure 5), the ordering holds: CFA posts the smallest worst-decile change in all three domains (−3.1%/−1.8%/−4.4%), RDA second-best throughout (−12.1%/−3.5%/−7.1%), against the strongest qualifying baseline (GRPO+KL: −14.2%/−8.3%/−11.8%); CPO degrades sharply at this scale (worst-1 accuracy of −64.8% MedBookVQA, and −44.6% Puzzle), performing worse than plain GRPO on both. The Real-vs-Gaussian gap narrows slightly from 2B to 8B on Navigation (10.5 → 9.0 points), consistent with content-free anchors needing less relative budget as capacity grows.

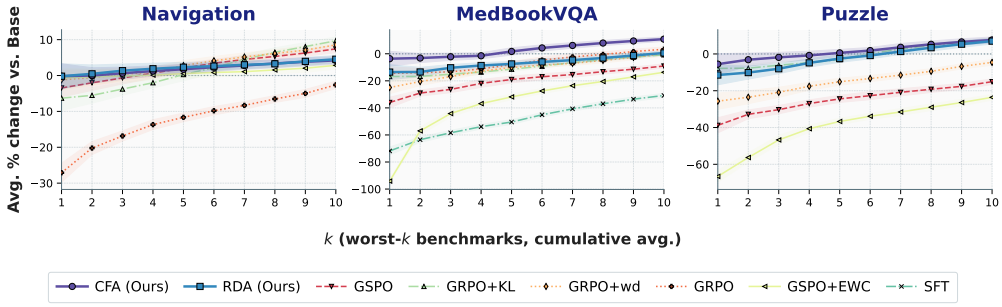

Figure 4: **Qwen3-VL-4B.** Worst-k forgetting curves on the same three domains and under the same reporting criterion as Figure 3. MedBookVQA and Puzzle reproduce the 2B ordering, with CFA and RDA again separating cleanly from every baseline. Navigation shows an atypical pattern at this scale specifically, discussed in Appendix C.2.

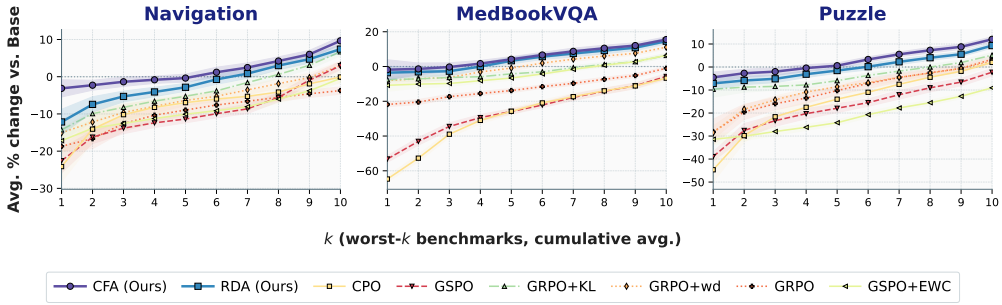

Figure 5: **Qwen3-VL-8B.** Worst-k forgetting curves on the same three domains as Figures 3-4, at the largest scale tested. RDA and CFA remain the two best-retaining methods across all three domains, and the gap to the strongest baseline does not shrink with scale.

Across all nine domain-scale combinations, CFA holds worst-decile change under −10% (range −0.3% to −9.1%), versus −8.1% to −67.4% for the best unregularized baseline per cell. The failure is qualitative as well as quantitative: on GRPO's most severe case (MedBookVQA, −67.4%), post-RL models retain fluent, correctly structured reasoning while the correspondence between that trace and the actual image content degrades into three recurring failure modes: hallucinated instances, shortcut answers, and loss of verification. Full examples of each are in Appendix H.

5.3 Anchor Content, Budget, and Divergence-Measure Ablations

Figure 6 shows a clear difference in budget efficiency between real and Gaussian anchors. Real anchors reach essentially their full benefit by N=320 and then plateau, with larger budgets providing little or slightly negative improvement. Gaussian anchors require more samples: they are weaker at N=320, match real anchors by N=640, and plateaus by N=2560. This crossover is consistent with the bias-variance analysis in Appendix D.5: real anchors

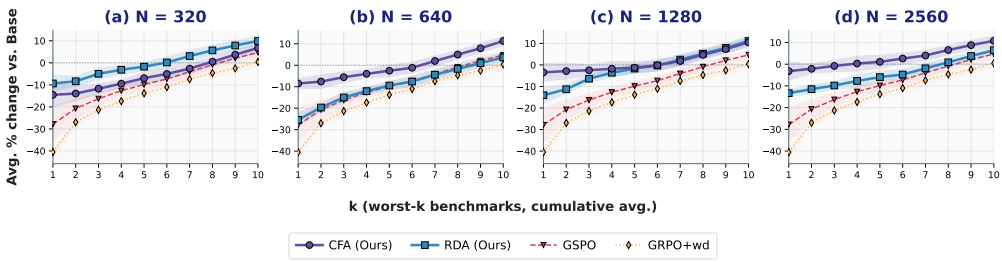

Figure 6: **Anchor budget and content.** Worst-k forgetting curves on Navigation (2B) for RDA, CFA, GSPO, and GRPO (wd=0.1), at four anchor budgets $N \in \{320, 640, 1280, 2560\}$. Real anchors are budget-efficient and essentially flat past N=320; Gaussian anchors are budget-hungry, matching real anchors by N=640 and exceeding them on the worst-1 metric, while real anchors degrade slightly relative to their own N=320 peak. This is the pattern Corollary 4 predicts.

reduce estimation error quickly but are limited by the representational bias of the curated pool, whereas content-free anchors trade higher sampling variance for a potentially lower bias floor. Corollary 4 formalizes this crossover behavior.

Figure 7 isolates the effect of the divergence and weighting choices at the deployed anchor budget (N=1280). Within real-anchor content, the predicted ordering holds cleanly at every k: RDA outperforms Fisher-KL (Real), which in turn outperforms Unweighted-KL (Real). This isolates the value of the embedding metric K over the discrete Fisher-Rao metric, and, within the Fisher-Rao metric, the added value of varentropy weighting over uniform weighting. Within Gaussian-anchor content, the same ordering holds for the most part. In both families, Unweighted-KL is the weakest configuration throughout, confirming that varentropy weighting provides a consistent benefit independent of anchor content.

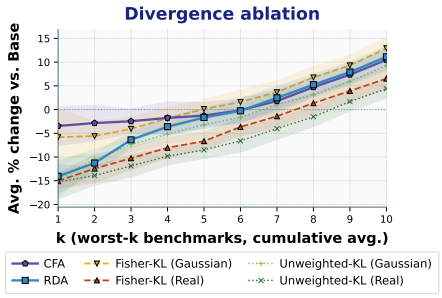

Figure 7: **Divergence ablation.** Worst-k forgetting, Navigation (2B), N=1280. Full results in Table 6.

6 Conclusion

Single-domain RL post-training is deployed on the assumption that it forgets little of what it never touches. We began by showing this is false: across Qwen3-VL at 2B, 4B, and 8B scale, GRPO and GSPO silently collapse fine-grained grounding and document perception, with relative drops reaching −67% on benchmarks the training domain never touches, while conventional suite-wide averages hide the damage behind unrelated gains on the same run. Every existing fix, EWC, replay, KL toward a reference, responds to this by assuming the anchor set used to hold the model in check must represent the capabilities being protected, exactly the assumption we argued, at the outset, becomes infeasible once the goal is general capability rather than a handful of anticipated domains. We have shown it is also unnecessary. Recasting anchoring as exact Gauss-Newton functional regularization reveals why: the curvature an anchor set estimates concentrates at a rate independent of what the anchors contain, so an anchor's job is to supply coverage, not relevance. Content-free anchors, pure noise with no connection to any capability, exploit exactly this gap: CFA matches a curated five-dimension coreset at a modest budget ($N = 1280$), and holds worst-decile degradation under −10% in every domain and scale tested, with no real, licensed, or curated data anywhere in the loop. The result reframes what it means to protect a deployed model's capabilities: not curating the right data for the capabilities you can anticipate, but discarding the need to anticipate them at all, probing the representation manifold in directions task gradients never reach.

AI use statement

In this work, we used generative AI tools exclusively for assistance with grammar checking, text formatting, and LaTeX typesetting. We have not used generative AI tools for scientific data generation, model implementation, or empirical experiment execution. We take full responsibility for the final content of this work, including all text, mathematical derivations, empirical claims, and conclusions.

Reproducibility Statement

Our implementation makes minimal modifications to the open-source Continual Policy Optimization (CPO) codebase of Luo et al. (2026)[1]. From this base we inherit the GRPO/GSPO trainer, dataset loaders, and evaluation harness; CFA is implemented as a trainer subclass on top of it, with the core mechanism given as Algorithm 2 in Appendix F. The full hyperparameter search space swept for every method reported in the paper (anchor budget N, regularizer strength λ, divergence family, learning rate, KL coefficient β, and number of optimization steps) is given in Appendix I.

Datasets. Continual post-training and in-domain evaluation use the five MRCL benchmark tasks (MedBookVQA, Navigation, We-Math2, Puzzle, and FinMME), released at `https://huggingface.co/datasets/MaolinLuo/MRCL`. General-capability (out-of-domain) retention is evaluated on nine external benchmarks spanning STEM reasoning, general VQA, chart/document understanding, and grounding: MMMU-Pro, MathVerse, MathVision, MathVista, RealWorldQA, MMStar, DocVQA, CharXiv, and CountBenchQA.

Compute. All experiments were run on nodes of 8 NVIDIA H100 GPUs and 8 NVIDIA B40 GPUs. We use DeepSpeed ZeRO Stage 2 for Qwen3-VL-2B, and Qwen3-VL-4B and ZeRO Stage 3 with CPU offload for Qwen3-VL-8B.

Ethics statement

This work studies a failure mode of RL post-training and a mitigation for it. To the best of our knowledge, we do not foresee a plausible dual-use concern specific to either the diagnosis or the method. We show that CFA's anchor sets can be built entirely from synthetic noise, removing a data-provenance and licensing burden that otherwise falls on practitioners trying to protect a model's prior capabilities. We view this as a net-positive access effect rather than a risk. Forgetting mitigation is not safety alignment: CFA constrains behavioral drift on capabilities present in the reference policy, and does not itself constrain or evaluate whether those reference behaviors were safe or desirable to begin with. We do not view this as unique to our method as every regularizer discussed in Section 2 that constrains drift toward a reference has the same property.

References

Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: a strong, simple baseline. *Advances in neural information processing systems*, 33:15920–15930, 2020.

Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with a-gem. *arXiv preprint arXiv:1812.00420*, 2018.

Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet K. Dokania, Philip H. S. Torr, and Marc'Aurelio Ranzato. Continual learning with tiny episodic memories. In *Workshop on Multi-Task and Lifelong Reinforcement Learning*, 2019.

[1] `https://github.com/MaolinLuo/CPO`

Howard Chen, Noam Razin, Karthik R Narasimhan, and Danqi Chen. Retaining by doing: The role of on-policy data in mitigating forgetting. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=ODTM64azGa.

Yihe Dong and Will Sawin. Muon $\hat{p}$: Muon with fractional spectral powers. *arXiv preprint arXiv:2606.13867*, 2026.

Chongyu Fan, Gaowen Liu, Mingyi Hong, Ramana Rao Kompella, and Sijia Liu. Rethinking muon beyond pretraining: Spectral failures and high-pass remedies for vla and rlvr. *arXiv preprint arXiv:2605.19282*, 2026.

Mehrdad Farajtabar, Navid Azizan, Alex Mott, and Ang Li. Orthogonal gradient descent for continual learning. In *International conference on artificial intelligence and statistics*, pp. 3762–3773. PMLR, 2020.

Robert M French. Catastrophic forgetting in connectionist networks. *Trends in cognitive sciences*, 3(4):128–135, 1999.

Teofilo F Gonzalez. Clustering to minimize the maximum intercluster distance. *Theoretical computer science*, 38:293–306, 1985.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *International Conference on Learning Representations*, 2023.

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. *Advances in neural information processing systems*, 31, 2018.

Xiaowen Jiang, Andrei Semenov, and Sebastian U Stich. Enhancing llm training via spectral clipping. *arXiv preprint arXiv:2603.14315*, 2026.

Keller Jordan, Yuchen Jin, Vlado Boza, Jiacheng You, Franz Cesista, Laker Newhouse, and Jeremy Bernstein. Muon: An optimizer for hidden layers in neural networks, 2024. *URL https://kellerjordan. github. io/posts/muon*, 6(3):4, 2024.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

Song Lai, Haohan Zhao, Rong Feng, Changyi Ma, Wenzhuo Liu, Hongbo Zhao, Xi Lin, Dong Yi, Qingfu Zhang, Hongbin Liu, et al. Reinforcement fine-tuning naturally mitigates forgetting in continual post-training. *arXiv preprint arXiv:2507.05386*, 2025.

Hongyu Li, Liang Ding, Meng Fang, and Dacheng Tao. Revisiting catastrophic forgetting in large language model tuning. In *Findings of the association for computational linguistics: EMNLP 2024*, pp. 4297–4308, 2024.

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

Raphael Gontijo Lopes, Stefano Fenu, and Thad Starner. Data-free knowledge distillation for deep neural networks. *arXiv preprint arXiv:1710.07535*, 2017.

David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. *Advances in neural information processing systems*, 30, 2017.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Mao-Lin Luo, Zhe-Xu Wang, Zi-Hao Zhou, Bo Ye, Jian Zhao, Min-Ling Zhang, and Tong Wei. Rl forgets! towards continual policy optimization. *arXiv preprint arXiv:2607.04364*, 2026.

Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. An empirical study of catastrophic forgetting in large language models during continual fine-tuning. arxiv10. 48550. *arXiv preprint arXiv.2308.08747*, 2023.

Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pp. 109–165. Elsevier, 1989.

Gaurav Kumar Nayak, Konda Reddy Mopuri, Vaisakh Shaj, Venkatesh Babu Radhakrishnan, and Anirban Chakraborty. Zero-shot knowledge distillation in deep networks. In *International conference on machine learning*, pp. 4743–4751. PMLR, 2019.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022.

David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. Experience replay for continual learning. *Advances in neural information processing systems*, 32, 2019a.

David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy P. Lillicrap, and Greg Wayne. Experience replay for continual learning, 2019b. URL `https://arxiv.org/abs/1811.11682`.

Mahdi Sabbaghi, George Pappas, Adel Javanmard, and Hamed Hassani. Robust policy optimization to prevent catastrophic forgetting. *arXiv preprint arXiv:2602.08813*, 2026.

Gobinda Saha, Isha Garg, and Kaushik Roy. Gradient projection memory for continual learning. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=3AOj0RCNC2`.

Thomas Scialom, Tuhin Chakrabarty, and Smaranda Muresan. Fine-tuned language models are continual learners. In *Proceedings of the 2022 conference on empirical methods in natural language processing*, pp. 6107–6122, 2022.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Idan Shenfeld, Jyothish Pari, and Pulkit Agrawal. Rl's razor: Why online reinforcement learning forgets less. In *International Conference on Learning Representations*, volume 2026, pp. 59839–59864, 2026.

Michalis K. Titsias, Jonathan Schwarz, Alexander G. de G. Matthews, Razvan Pascanu, and Yee Whye Teh. Functional regularisation for continual learning with gaussian processes. In *International Conference on Learning Representations*, 2020. URL `https://openreview.net/forum?id=HkxCzeHFDB`.

Joel A. Tropp. An introduction to matrix concentration inequalities, 2015. URL `https://arxiv.org/abs/1501.01571`.

Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, et al. Robust fine-tuning of zero-shot models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7959–7971, 2022.

Fangzhou Wu, Rikhav Shah, Sandeep Silwal, and Qiuyi Zhang. Dynmuon: A dynamic spectral shaping view of muon. *arXiv preprint arXiv:2605.17109*, 2026.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. Resolving interference when merging models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 9127–9148, 2023.

Hongxu Yin, Pavlo Molchanov, Jose M Alvarez, Zhizhong Li, Arun Mallya, Derek Hoiem, Niraj K Jha, and Jan Kautz. Dreaming to distill: Data-free knowledge transfer via deepinversion. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8712–8721. IEEE, 2020.

Le Yu, Bowen Yu, Haiyang Shen, Xinyin Huang, Fei Huang, and Yongbin Li. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *International Conference on Machine Learning*, 2024.

Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International conference on machine learning*, pp. 3987–3995. Pmlr, 2017.

Han Zhang, Lin Gui, Yu Lei, Yuanzhao Zhai, Yehong Zhang, Zhuo Zhang, Yulan He, Hui Wang, Yue Yu, Kam-Fai Wong, et al. Copr: Continual human preference learning via optimal policy regularization. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 5377–5398, 2025.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, Jingren Zhou, and Junyang Lin. Group sequence policy optimization, 2025. URL `https://arxiv.org/abs/2507.18071`.

Hanqing Zhu, Wenyan Cong, Zhizhou Sha, Sagnik Mukherjee, Xinyuan Song, David González-Martínez, Xiaoxia Wu, Yuandong Tian, Shiwei Liu, David Z Pan, et al. Iso: An rlvr-native optimization stack. *arXiv preprint arXiv:2607.19331*, 2026.

A A Toy Testbed for Forgetting

Before running expensive experiments at Qwen3-VL scale (Section 5), we construct a minimal, fully enumerable testbed to resolve two foundational questions motivating CFA's design. First, do mitigations that constrain the optimizer, update geometry, or parameter movement perform as well as explicitly evaluating functional behavior against a reference anchor set? Second, does an anchor set require authentic prior-task data, or can completely unstructured noise, bearing zero resemblance to the pretraining data distribution, fulfill the same behavioral-anchoring purpose?

A.1 Setup

We use a 3-layer MLP pretrained via supervised fine-tuning on FashionMNIST (Task A, 60K examples, predicting apparel category), then GRPO post-trained on ParityMNIST (Task B, 60K examples, predicting digit parity) for 20 epochs. This mirrors the single-domain RL transition studied at scale in Section 3: one established prior capability, one distinct downstream RL objective, no interleaved task sequence. The base checkpoint scores 88.7% on FashionMNIST and 52.4% on ParityMNIST. AdamW post-training with no mitigation reaches 98.7% ParityMNIST accuracy while degrading FashionMNIST to 16.7% - a -82% relative loss of prior-task capability, reproducing the single-domain collapse in a testbed small enough for full spectral and representational profiling.

A.2 Optimizer-Level Fixes Are Not Enough

A natural first hypothesis is that this is an optimization artifact: a more geometrically principled update rule, spectral regularizer, or parameter-space movement constraint should prevent representation collapse while still letting the model master the new task. We test this by sweeping four families of optimizer-level and parameter-space interventions under identical initializations, learning-rate budgets, and schedules: (i) element-wise preconditioning - SGD with momentum and AdamW (Loshchilov & Hutter, 2017); (ii) spectral whitening and fractional shaping - Muon (Jordan et al., 2024; Liu et al., 2025), which projects momentum onto the polar factor $UV^\top$ via Newton-Schulz iteration; fractional spectral powers (Muon^p, Dong & Sawin, 2026) interpolating steepest descent across Schatten-q norms; and curvature-adaptive spectral scheduling (DynMuon, Wu et al., 2026); (iii) spectral filtering and manifold constraints - spectral-norm clipping (SPECTRA, Jiang et al., 2026), high-pass Newton-Schulz filtering for low-SNR RL regimes (Pion, Fan et al., 2026), and RLVR-native frame optimization with frozen base spectra (ISO, Zhu et al., 2026); and (iv) parameter-space movement penalties - an explicit L_2 penalty on $\Delta W = W - W_0$, and CPO (Luo et al., 2026).

None of these interventions resolve catastrophic forgetting (Figure 2). Every method reaches near-ceiling ParityMNIST accuracy (96.5-98.8%) while sacrificing most of Task-A performance: unconstrained preconditioning (AdamW, standard Muon) retains only 16.7-40.8% Fashion accuracy; spectral shaping and tail suppression (DynMuon, Muon^p at p=0.3, SPECTRA, Pion) modestly improve retention to 47.9-54.2%, still far below the 88.7% base; frame-frozen manifold optimization (ISO) still suffers catastrophic collapse (62.8%); and explicit distance penalties (CPO, L_2 parameter damping) mitigate weight drift but still fail to protect Task-A classification boundaries (18.1% retention). SGD with momentum is a partial exception, retaining Fashion accuracy noticeably better than AdamW or standard Muon at a comparable Parity acquisition level. This toy-scale advantage is not, on its own, evidence of a viable mitigation strategy at scale. The toy network has a single weight scale and few layers, so a single global learning rate adequately characterizes its optimization dynamics. Qwen3-VL has no such uniformity: attention projections, MLP expansion layers, embedding tables, and normalization parameters each operate at markedly different effective weight and gradient scales, and this heterogeneity grows with depth. We therefore do not treat SGD's toy-scale behavior as informative about mitigation strategies at the Qwen3-VL scale and exclude it from the large-scale comparisons in Section 5.

These findings align with Zhu et al. (2026), who report that RLVR barely perturbs singular values ($\sim$ 3% spectral residual vs. $\sim$ 35% in SFT), localizing updates primarily in the

singular frames (U, V). Yet as demonstrated here, weight-space geometric constraints do not guarantee behavioral preservation: because the mapping from parameter matrices to representation manifolds is non-injective and highly non-linear, constraining *how* an update navigates weight space does not prevent catastrophic output drift.

A.3 Random Anchors Match Real Replay

Figure 2 contrasts these optimizer-level methods against two functional reference-matching paradigms: real-data replay (distillation on a stored buffer of genuine Task-A exemplars; e.g., Experience Replay Chaudhry et al., 2019 and A-GEM, Chaudhry et al., 2018, and synthetic anchoring, matching the model's behavior on purely random isotropic Gaussian noise against the frozen reference W_0 via the same Fisher-weighted MMD used at scale (CFA, Eq. 2).[2] Furthermore, in single-step classification where sequence length $T = 1$, the sequence-normalized varentropy weight $\tilde{w}_t = w_t/(\sum_{s=1}^{T} w_s + \epsilon)$ collapses identically to $\tilde{w}_1 \equiv 1.0$ regardless of logit values. The toy benchmark therefore isolates the geometric impact of unweighted functional MMD anchoring from temporal entropy scheduling.

Real Task-A replay achieves one of the best retention scores (86.6% Fashion), and synthetic noise anchoring (86.4%) sits within noise of it, both near base performance, while every optimizer-level and parameter-space method collapses. What governs retention is not the algorithmic complexity of the optimizer, the rank of the update, or whether real domain data is cached, but whether the objective explicitly evaluates and anchors functional behavior against a reference.

B Metric Motivation

The suite average hides the failure. Averaging is not merely less sensitive than the worst case here but is actively misleading. Across the 68 qualifying runs in this study, 20 post a suite-wide mean above -5%, which by the standard metric reads as essentially no forgetting, while simultaneously losing more than 20% on their single worst benchmark. CPO on Puzzle at 8B averages $+2.0\%$ yet drops -44.6% at $k = 1$; GRPO (wd= 0.1) on Navigation at 2B averages $+0.4\%$ while CountBenchQA falls -40.5%. The suite mean lets positive transfer on reasoning benchmarks pay for the destruction of a narrow capability, and gives the false security of minimal capability degradation. A practitioner reading only $\bar{\Delta}$ would ship both models. Concretely, a method can appear competitive at $k = |\mathcal{B}|$ while silently dropping by 30 to 60 points at $k = 1$, which is exactly what our metric worst-k curve is meant to expose.

On comparing different benchmarks at the same k. Because $\mathcal{B}_k$ is chosen per method, two methods at a given k might be scored on different benchmarks. It is also empirically the common case rather than an edge case: across our qualifying runs, 7 of the 10 common-suite benchmarks appear as some method's worst, the most frequent (CountBenchQA) accounts for only 32% of them, and two methods within the same panel agree on $\mathcal{B}_1$ for just 25% of pairs. We argue this is a property the metric must have, for three reasons.

(i) The rule is identical for every method, and it is self-penalizing. By the second equality in Eq. (1), Δ_k is the minimum over *all* size-k subsets. Every method is therefore evaluated under the same rule and at its own least favorable slice; none is scored on a subset chosen to flatter it. A method that wins at a given k does so despite being measured where it is individually weakest.

[2] The 3-layer MLP has no token-embedding matrix E; we substitute the frozen reference model's final-layer classifier weight matrix $W_{\text{out}}^{\text{ref}} \in \mathbb{R}^{C \times d}$, which plays an analogous linear-projection role. We note an important dimensional rank inversion between the two settings: in the toy MLP, the number of classes is small ($C = 10 \leq d$), so the linear Gram matrix $K_{\text{toy}} = \frac{1}{d} W_{\text{out}}^{\text{ref}} (W_{\text{out}}^{\text{ref}})^{\top} \in \mathbb{R}^{C \times C}$ has full rank $\text{rank}(K_{\text{toy}}) = C = 10$. In contrast, for VLMs, the vocabulary dimension vastly exceeds the hidden embedding dimension ($d \approx 2048 \ll V \approx 152{,}000$), rendering $K = \frac{1}{d} E E^{\top}$ severely rank-deficient ($\text{rank}(K) \leq d$, with null-space dimension $> 147{,}000$; see Section 3.5). Corollary 2's algebraic pullback holds for any fixed linear projection, but the VLM setting induces a semantic low-rank projection rather than full categorical coverage.

Table 1: **Navigation, Qwen3-VL-2B.** Common-suite accuracy (mean ± std) for every baseline and for CFA at anchor budget $N = 1280$ (Real/Gaussian). **Bold** marks the best method per column; underline marks the second best. "Target" is final in-domain (target-task) accuracy.

	STEM				Gen. VQA		Chart/Doc.			Ground.	
Method	MMMU	MVi	MVn	MVe	RWQA	MMS	DocVQA	CX-DQ	CX-RQ	CBQA	Target
Base	48.4 ± 2.4	56.0 ± 3.1	21.1 ± 1.4	37.4 ± 1.5	58.4 ± 3.5	46.7 ± 2.5	68.6 ± 1.2	60.8 ± 1.5	28.0 ± 2.8	82.5 ± 3.4	3.5
GRPO (wd=0)	44.9 ± 2.3	55.6 ± 3.1	19.9 ± 1.4	36.2 ± 1.5	49.5 ± 3.5	58.7 ± 2.5	**71.6 ± 1.2**	60.9 ± 1.5	28.3 ± 2.8	80.7 ± 3.5	34.4†
GRPO (wd=0.1)	48.7 ± 2.4	64.2 ± 3.0	24.3 ± 1.5	42.7 ± 1.5	52.5 ± 3.5	59.0 ± 2.5	59.4 ± 1.3	57.4 ± 1.5	28.8 ± 2.8	49.1 ± 4.4	70.8
GRPO+KL	47.8 ± 2.4	64.2 ± 3.0	25.8 ± 1.6	44.6 ± 1.6	60.6 ± 3.5	58.8 ± 2.5	45.8 ± 1.3	52.8 ± 1.6	25.3 ± 2.7	69.0 ± 4.1	75.7
GSPO	51.0 ± 2.4	64.8 ± 3.0	27.0 ± 1.6	45.4 ± 1.6	59.2 ± 3.5	58.7 ± 2.5	67.4 ± 1.3	56.2 ± 1.5	20.2 ± 2.8	71.3 ± 4.0	78.7
GSPO+EWC	47.8 ± 2.4	58.5 ± 3.0	21.3 ± 1.4	37.0 ± 1.5	48.5 ± 3.5	56.3 ± 2.5	71.4 ± 1.2	57.5 ± 1.5	26.2 ± 2.7	**87.6 ± 2.9**	51.1†
LoRA	46.9 ± 2.4	57.2 ± 3.1	18.2 ± 1.4	37.3 ± 1.5	55.8 ± 3.5	39.1 ± 2.5	65.1 ± 1.3	58.6 ± 1.5	25.3 ± 2.7	68.8 ± 4.1	3.9†
SFT	46.2 ± 2.4	42.3 ± 3.1	14.4 ± 1.2	31.9 ± 1.5	57.0 ± 3.5	35.9 ± 2.4	48.0 ± 1.3	56.9 ± 1.5	19.8 ± 2.5	59.7 ± 4.3	3.9†
CPO	48.9 ± 2.4	58.4 ± 3.0	26.6 ± 1.6	45.9 ± 1.6	57.6 ± 3.5	56.3 ± 2.5	52.8 ± 1.3	55.0 ± 1.5	27.1 ± 2.8	66.6 ± 4.2	85.5
RDA	**51.4 ± 2.4**	**66.6 ± 2.9**	**29.6 ± 1.6**	46.6 ± 1.6	**61.2 ± 3.5**	**60.3 ± 2.5**	70.9 ± 1.2	55.6 ± 1.5	**29.9 ± 2.8**	70.9 ± 4.0	85.7
CFA	51.1 ± 2.4	63.6 ± 3.0	29.3 ± 1.6	**46.8 ± 1.6**	57.1 ± 3.5	59.9 ± 2.5	67.5 ± 1.3	**61.1 ± 1.5**	28.1 ± 2.8	79.6 ± 3.6	**87.0**

† Below the per-domain in-domain-gain threshold; excluded from the headline forgetting comparison (Section 5.1).

(ii) A fixed, pre-chosen subset would be less fair, not more. Which capability collapses is method-dependent as highlighted by the 32% figure above. Committing in advance to a single probe would turn the comparison into a test of one arbitrary capability: fixing on CountBenchQA would have missed GSPO + EWC's −94.1% collapse on DocVQA at 4B (Section 5.2), and fixing on DocVQA would have missed the CountBenchQA collapses that motivate this work. The quantity a practitioner cares about is whether *any* retained capability was destroyed, not whether a particular one was.

(iii) Nothing is discarded, so the curves compare distributions rather than subsets. The map $\{\Delta_b\} \mapsto \{\Delta_k\}_{k=1}^{|\mathcal{B}|}$ is a bijection onto the sorted score vector: reporting the whole curve is equivalent to reporting the entire empirical distribution of per-benchmark change, just re-indexed so that the tail is read first. Consequently, if one method's curve lies above another's at *every* k, its loss profile dominates in the majorization (second-order stochastic dominance) sense, and is preferred under every risk-averse aggregation of $\{\Delta_b\}$ and not merely at the particular k.

That last condition is what we actually observe, so our conclusions do not rest on a choice of k: CFA dominates the qualifying baselines at every k in 54 of 59 method comparisons (92%), and in 56 of 59 (95%) over the worst half of the suite, $k \leq 5$. All five exceptions occur in the two Navigation panels at points where *both* curves are non-negative, i.e. where neither method is forgetting and the comparison has become one of average positive transfer rather than of tail damage. Per-benchmark numbers are reported in full in Appendix C for better comparison.

To avoid conflating failure to learn the target task with capability retention, we exclude baselines that fail to reach an in-domain gain threshold from headline retention comparisons, reporting them in full only in Appendix C.

C Full Result Tables

This section provides the complete evaluation results across all post-training configurations, baseline algorithms, and model scales (Qwen3-VL-2B, 4B, and 8B) on the 10-benchmark common evaluation suite. For every setting, the worst-k retention curves ($k = 1, \ldots, 10$) illustrated in Section 5 (Figures 3-5) are derived directly from these per-benchmark accuracies: for a checkpoint with scores s_b and base scores s_b^{base}, the per-benchmark relative shifts $\Delta_b = (s_b - s_b^{\text{base}})/s_b^{\text{base}}$ are sorted in ascending order $\Delta_{(1)} \leq \Delta_{(2)} \leq \cdots \leq \Delta_{(10)}$, and the worst-$k$ metric is computed as $\mathcal{W}_k = \frac{1}{k}\sum_{j=1}^{k} \Delta_{(j)}$. Specifically, Navigation evaluations across scales are reported in Tables 1, 7, and 10; MedBookVQA evaluations in Tables 3, 8, and 11; and Puzzle evaluations in Tables 4, 9, and 12. Table 2 evaluates post-hoc weight merging (WiSE-FT; Wortsman et al., 2022) across mixing coefficients $\alpha \in \{0.1, \ldots, 0.9\}$ on Navigation. Table 5 details the full anchor budget scaling sweep ($N \in \{320, 640, 1280, 2560\}$), and Table 6 compares functional divergence criteria.

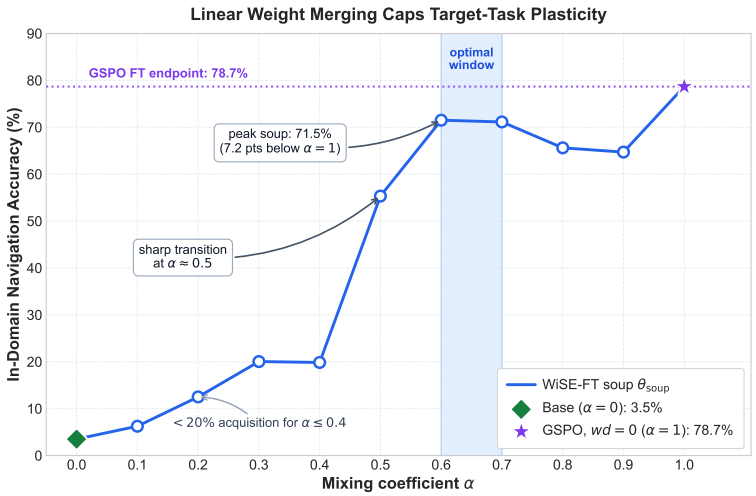

Figure 8: **WiSE-FT In-Domain Navigation Accuracy vs. Mixing Coefficient α.** Linear weight interpolation $\theta_{\text{soup}} = (1-\alpha)\theta_{\text{base}} + \alpha\theta_{\text{FT}}$ on Qwen3-VL-2B Navigation. The model exhibits minimal target-task acquisition for $\alpha \leq 0.4$ ($< 20\%$), undergoes a sharp transition at $\alpha \approx 0.5$, and plateaus at an optimal specialization window of $\alpha \in [0.6, 0.7]$ with 71.5% accuracy which is still well below the 78.7% reached by GSPO ($\alpha = 1$), the FT model used in the model-souping. The dotted horizontal line marks that GSPO endpoint; no soup along the linear path reaches it.

C.1 Post-Hoc Weight Merging (WiSE-FT)

We evaluate linear post-hoc weight interpolation (WiSE-FT; Wortsman et al., 2022) on Navigation with Qwen3-VL-2B. Given the frozen base checkpoint θ_{base} and the unregularized fine-tuned policy θ_{FT} (GSPO, $wd = 0$, step 600), we construct model soups along the linear path $\theta_{\text{soup}} = (1-\alpha)\theta_{\text{base}} + \alpha\theta_{\text{FT}}$ sweeping $\alpha \in \{0.1, 0.2, \ldots, 0.9\}$. Figure 8 traces in-domain task acquisition across the sweep, and Table 2 details the full 10-benchmark evaluation profile alongside final target accuracy.

Why Weight Merging is Suboptimal. While post-hoc weight merging is appealing because it incurs no training-time regularization overhead, our results demonstrate why it remains fundamentally suboptimal compared to dynamic representation anchoring:

1. **Impaired Target-Domain Plasticity:** As illustrated in Figure 8, weight interpolation exhibits near-zero learning of the new task for $\alpha \leq 0.4$ ($< 20\%$ accuracy), followed by a steep phase transition at $\alpha \approx 0.5$, before plateauing at an upper ceiling of 71.51% ($\alpha = 0.6$) and 71.14% ($\alpha = 0.7$). Linear interpolation severely truncates the model's plastic capacity on the target task, trailing representation anchoring by more than 15 percentage points (86.95% for CFA).
2. **Rigid Zero-Sum Parameter Trade-Off:** Linearly averaging parameter vectors implicitly assumes a convex, Euclidean landscape between pre-trained and fine-tuned checkpoints. On non-linear vision-language policy manifolds, however, linear weight mixtures perturb shared representations uniformly, enforcing an unavoidable trade-off: any parameter shift toward the base model directly dismantles newly acquired target capabilities.
3. **Suboptimal Tail-Risk Retention:** Even at its peak specialization window ($\alpha = 0.6$), WiSE-FT still incurs a -10.12% worst-1 relative drop on CountBenchQA, compared to only -3.46% for CFA (Figure 9). Dynamic representation anchoring evaluates the model's induced functional geometry on anchor points during training, selectively penalizing divergence in feature space while granting full optimization freedom along task-relevant gradient subspaces.

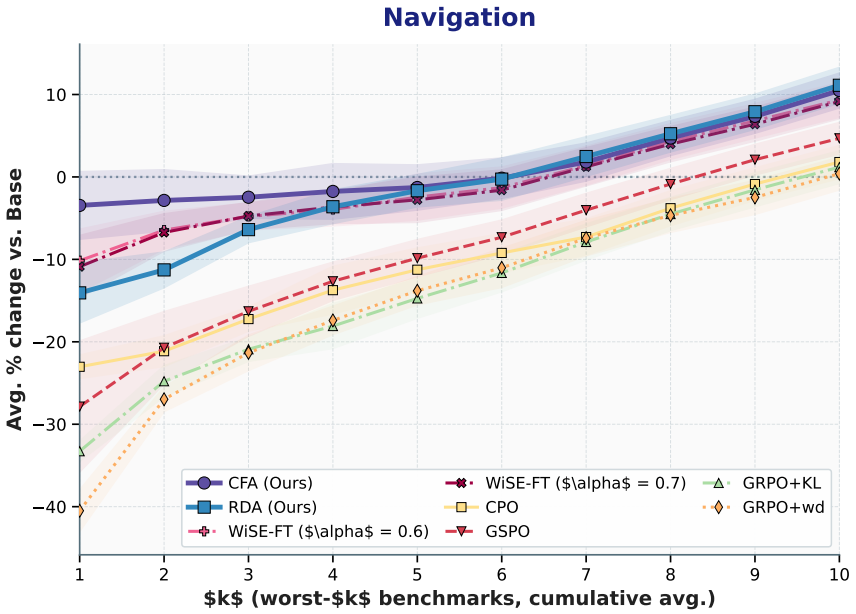

Figure 9: **Worst-k forgetting on Navigation (Qwen3-VL-2B): weight merging versus representation anchoring.** Mean relative change against the base model over the k worst-affected out-of-domain benchmarks. At its two best mixing coefficients ($\alpha \in \{0.6, 0.7\}$), WiSE-FT improves markedly over unregularized GSPO and the parameter-movement baselines, but still concedes roughly -10% at $k = 1$; CFA holds the same quantity to -3.5% while reaching higher in-domain accuracy (Table 2). Shaded bands are bootstrap confidence intervals over benchmark resamples.

Table 2: **Weight Merging (WiSE-FT), Qwen3-VL-2B Navigation.** Common-suite accuracy (mean ± std) for linear weight interpolation $\theta_{\text{soup}} = (1-\alpha)\theta_{\text{base}} + \alpha\theta_{\text{FT}}$ across mixing coefficients $\alpha \in \{0.1, \ldots, 0.9\}$ between the base model and the GSPO ($wd = 0$) checkpoint, compared with Base, unregularized GSPO, and CFA ($N = 1280$). **Bold** marks the best method per column; underline marks the second best. "Target" is final in-domain (target-task) accuracy.

	STEM				Gen. VQA		Chart/Doc.			Ground.	
Method	MMMU	MVi	MVn	MVe	RWQA	MMS	DocVQA	CX-DQ	CX-RQ	CBQA	Target
Base	48.4 ± 2.4	56.0 ± 3.1	21.1 ± 1.4	37.4 ± 1.5	58.4 ± 3.5	46.7 ± 2.5	68.6 ± 1.2	60.8 ± 1.5	28.0 ± 2.8	**82.5 ± 3.4**	3.5
GSPO (wd=0)	51.0 ± 2.4	64.8 ± 3.0	27.0 ± 1.6	45.4 ± 1.6	59.2 ± 3.5	58.7 ± 2.5	67.4 ± 1.3	56.2 ± 1.5	20.2 ± 2.8	71.3 ± 4.0	78.7
WiSE-FT ($\alpha = 0.1$)	50.0 ± 2.4	59.2 ± 3.0	23.6 ± 1.6	39.4 ± 1.6	57.9 ± 3.5	51.4 ± 2.5	69.8 ± 1.2	63.5 ± 1.5	26.4 ± 2.8	80.5 ± 1.8	6.2
WiSE-FT ($\alpha = 0.2$)	51.0 ± 2.4	61.2 ± 3.0	24.9 ± 1.6	40.9 ± 1.6	57.4 ± 3.5	54.8 ± 2.5	70.2 ± 1.2	63.1 ± 1.5	27.8 ± 2.8	82.3 ± 1.7	12.5
WiSE-FT ($\alpha = 0.3$)	48.7 ± 2.4	61.4 ± 3.0	27.8 ± 1.7	44.2 ± 1.6	52.8 ± 3.5	60.6 ± 2.4	70.0 ± 1.2	**64.2 ± 1.5**	30.0 ± 2.9	81.5 ± 1.8	20.0
WiSE-FT ($\alpha = 0.4$)	49.1 ± 2.4	61.0 ± 3.0	27.5 ± 1.7	44.3 ± 1.6	51.2 ± 3.5	60.1 ± 2.4	69.3 ± 1.2	**64.2 ± 1.5**	29.7 ± 2.9	82.1 ± 1.7	19.9
WiSE-FT ($\alpha = 0.5$)	50.5 ± 2.4	**68.4 ± 2.9**	**30.2 ± 1.7**	**47.0 ± 1.6**	38.3 ± 3.4	**61.5 ± 2.4**	69.1 ± 1.2	62.1 ± 1.5	29.0 ± 2.9	78.6 ± 1.9	55.3
WiSE-FT ($\alpha = 0.6$)	50.6 ± 2.4	65.7 ± 2.9	27.9 ± 1.7	46.7 ± 1.6	57.6 ± 3.5	59.2 ± 2.5	67.6 ± 1.3	59.1 ± 1.6	29.0 ± 2.9	74.1 ± 2.0	71.5
WiSE-FT ($\alpha = 0.7$)	50.5 ± 2.4	66.2 ± 2.9	28.3 ± 1.7	46.1 ± 1.6	58.2 ± 3.5	58.7 ± 2.5	68.1 ± 1.2	59.2 ± 1.6	28.2 ± 2.8	73.5 ± 2.0	71.1
WiSE-FT ($\alpha = 0.8$)	50.6 ± 2.4	66.1 ± 2.9	27.8 ± 1.7	46.9 ± 1.6	59.1 ± 3.5	59.0 ± 2.5	67.3 ± 1.3	57.8 ± 1.6	**30.6 ± 2.9**	72.5 ± 2.0	65.6
WiSE-FT ($\alpha = 0.9$)	48.8 ± 2.4	65.5 ± 2.9	28.1 ± 1.7	46.5 ± 1.6	59.5 ± 3.5	59.1 ± 2.5	67.2 ± 1.3	57.1 ± 1.6	26.8 ± 2.8	71.1 ± 2.0	64.7
RDA	**51.4 ± 2.4**	66.6 ± 2.9	29.6 ± 1.6	46.6 ± 1.6	**61.2 ± 3.5**	60.3 ± 2.5	**70.9 ± 1.2**	55.6 ± 1.5	29.9 ± 2.8	70.9 ± 4.0	85.7
CFA	51.1 ± 2.4	63.6 ± 3.0	29.3 ± 1.6	46.8 ± 1.6	57.1 ± 3.5	59.9 ± 2.5	67.5 ± 1.3	61.1 ± 1.5	28.1 ± 2.8	79.6 ± 3.6	**87.0**

C.2 Understanding Domain and Scale Dynamics: Pre-Existing Competence and Task Memorizability

A prominent empirical pattern emerges from our 3×3 grid of domain and scale evaluations (Tables 1–12): Qwen3-VL-4B on Navigation learns the target task to high accuracy (93.4%) with virtually zero tail forgetting under standard RL (worst-1 drop of only -1.2% on GRPO $\lambda_{\text{wd}} = 0.1$, -2.2% on GSPO, and -0.1% on CFA). In sharp contrast:

1. At the identical 4B model scale, both MedBookVQA and Puzzle suffer severe catastrophic forgetting under unregularized RL (worst-1 drops of -25.0% to -38.9%, Tables 8 and 9).
2. On Navigation itself, both the smaller 2B model (worst-1 drop of -40.5% on CountBenchQA, Table 1) and the larger 8B model (worst-1 drops of -15.0% to -22.6%

Table 3: **MedBookVQA, Qwen3-VL-2B.** Common-suite accuracy (mean ± std) for every baseline and for CFA at anchor budget $N = 1280$ (Real/Gaussian). **Bold** marks the best method per column; underline marks the second best. "Target" is final in-domain (target-task) accuracy.

	STEM				Gen. VQA		Chart/Doc.			Ground.	
Method	MMMU	MVi	MVn	MVe	RWQA	MMS	DocVQA	CX-DQ	CX-RQ	CBQA	Target
Base	48.4 ± 2.4	56.0 ± 3.1	21.1 ± 1.4	37.4 ± 1.5	58.4 ± 3.5	46.7 ± 2.5	68.6 ± 1.2	60.8 ± 1.5	28.0 ± 2.8	82.5 ± 3.4	49.4
GRPO (wd=0)	37.3 ± 2.3	45.4 ± 3.1	15.6 ± 1.3	28.4 ± 1.4	40.1 ± 3.5	36.2 ± 2.4	24.7 ± 1.2	38.1 ± 1.5	16.3 ± 2.3	26.9 ± 3.9	**99.8**
GRPO (wd=0.1)	47.8 ± 2.4	58.1 ± 3.1	27.4 ± 1.6	44.8 ± 1.6	59.5 ± 3.5	58.5 ± 2.5	65.7 ± 1.3	55.1 ± 1.5	27.6 ± 2.8	56.6 ± 4.4	94.7
GRPO+KL	44.1 ± 2.3	63.2 ± 3.0	26.4 ± 1.6	41.2 ± 1.5	53.2 ± 3.5	57.7 ± 2.5	43.9 ± 1.3	50.4 ± 1.6	26.8 ± 2.8	74.8 ± 3.9	99.2
GSPO	34.4 ± 2.2	49.1 ± 3.1	17.5 ± 1.4	27.4 ± 1.4	38.8 ± 3.5	45.1 ± 2.5	34.4 ± 1.3	37.8 ± 1.5	14.1 ± 2.2	58.5 ± 4.4	99.0
GSPO+EWC	49.2 ± 2.4	60.6 ± 3.0	25.0 ± 1.5	39.0 ± 1.5	58.3 ± 3.5	59.4 ± 2.5	68.5 ± 1.2	61.1 ± 1.5	**30.3 ± 2.9**	84.3 ± 3.2	84.5
LoRA	49.1 ± 2.4	58.3 ± 3.1	23.8 ± 1.5	38.9 ± 1.5	**61.6 ± 3.5**	48.7 ± 2.5	66.0 ± 1.3	**62.8 ± 1.5**	27.8 ± 2.8	79.4 ± 3.6	50.8†
SFT	37.9 ± 2.3	36.6 ± 3.0	9.5 ± 1.0	16.2 ± 1.1	41.4 ± 3.5	34.7 ± 2.4	26.2 ± 1.2	42.8 ± 1.5	19.4 ± 2.5	17.5 ± 3.4	95.1
CPO	33.1 ± 2.2	43.6 ± 3.1	14.8 ± 1.3	21.2 ± 1.3	35.8 ± 3.4	39.9 ± 2.5	29.9 ± 1.2	34.7 ± 1.5	15.2 ± 2.2	59.9 ± 4.3	96.8
RDA	49.1 ± 2.4	60.3 ± 3.0	**29.6 ± 1.6**	**45.8 ± 1.6**	53.6 ± 3.5	59.1 ± 2.5	69.1 ± 1.2	59.5 ± 1.5	28.4 ± 2.8	83.7 ± 3.3	96.2
CFA	**50.8 ± 2.4**	**64.7 ± 3.0**	29.3 ± 1.6	45.4 ± 1.6	58.9 ± 3.5	**60.1 ± 2.5**	**70.9 ± 1.2**	60.6 ± 1.5	29.9 ± 2.8	**87.2 ± 3.0**	98.5

† Below the per-domain in-domain-gain threshold; excluded from the headline forgetting comparison (Section 5.1).

Table 4: **Puzzle, Qwen3-VL-2B.** Common-suite accuracy (mean ± std) for every baseline and for CFA at anchor budget $N = 1280$ (Real/Gaussian). **Bold** marks the best method per column; underline marks the second best. "Target" is final in-domain (target-task) accuracy. This table illustrates the averaging problem of Section 3.2: GRPO (wd= 0.1) posts a suite-average change of −0.7% while losing −24.8% on its worst benchmark, and GSPO + EWC averages +3.5% while losing −25.8%; CFA is the only method whose worst benchmark stays in single digits (−9.1%, suite average +7.3%). Note also that CPO's apparently strong retention here is accompanied by an in-domain gain of only +39.0 points against a median RL gain of +62.2 (63%), placing it below the per-domain gain threshold of Section 5.1.

	STEM				Gen. VQA		Chart/Doc.			Ground.	
Method	MMMU	MVi	MVn	MVe	RWQA	MMS	DocVQA	CX-DQ	CX-RQ	CBQA	Target
Base	48.4 ± 2.4	56.0 ± 3.1	21.1 ± 1.4	37.4 ± 1.5	58.4 ± 3.5	46.7 ± 2.5	68.6 ± 1.2	60.8 ± 1.5	28.0 ± 2.8	**82.5 ± 3.4**	23.8
GRPO (wd=0)	44.9 ± 2.3	58.5 ± 3.0	16.7 ± 1.3	34.5 ± 1.5	59.4 ± 3.5	54.9 ± 2.5	54.1 ± 1.3	56.5 ± 1.5	27.1 ± 2.8	60.3 ± 4.3	85.0
GRPO (wd=0.1)	46.1 ± 2.4	62.4 ± 3.0	21.1 ± 1.4	39.6 ± 1.5	54.8 ± 3.5	54.6 ± 2.5	51.6 ± 1.3	64.8 ± 1.5	26.9 ± 2.8	75.6 ± 3.8	82.2
GRPO+KL	45.1 ± 1.2	57.6 ± 1.6	22.3 ± 0.8	40.1 ± 0.8	47.8 ± 1.8	53.3 ± 1.3	52.9 ± 0.7	51.3 ± 0.8	24.7 ± 1.4	59.1 ± 2.2	86.0
GSPO	46.4 ± 2.4	58.1 ± 3.1	17.0 ± 1.3	38.4 ± 1.5	51.9 ± 3.5	53.3 ± 2.5	52.1 ± 1.3	56.5 ± 1.5	25.4 ± 2.7	49.7 ± 4.4	87.0
GSPO+EWC	48.4 ± 2.4	63.7 ± 3.0	25.2 ± 1.5	43.0 ± 1.6	**59.5 ± 3.5**	55.1 ± 2.5	65.1 ± 1.3	63.8 ± 1.5	25.9 ± 2.7	61.2 ± 4.3	88.2
LoRA	46.3 ± 2.4	52.2 ± 3.1	17.3 ± 1.4	33.3 ± 1.5	58.4 ± 3.5	49.3 ± 2.5	57.3 ± 1.3	62.9 ± 1.5	23.4 ± 2.6	44.4 ± 4.4	48.0†
SFT	46.0 ± 2.4	39.7 ± 3.0	10.9 ± 1.1	29.6 ± 1.4	47.6 ± 3.5	38.8 ± 2.5	27.5 ± 1.2	59.1 ± 1.5	20.7 ± 2.5	14.7 ± 3.1	33.0†
CPO	**53.3 ± 2.4**	**76.8 ± 2.6**	**30.7 ± 1.6**	**46.4 ± 1.6**	59.2 ± 3.5	**63.8 ± 2.4**	67.4 ± 1.3	**65.5 ± 1.5**	**43.6 ± 3.1**	82.1 ± 3.4	62.8†
RDA	46.8 ± 2.4	62.4 ± 3.0	23.5 ± 1.5	41.8 ± 1.5	58.8 ± 3.5	54.1 ± 2.5	62.5 ± 1.3	63.1 ± 1.5	27.0 ± 2.8	65.8 ± 4.2	82.5
CFA	48.3 ± 2.4	63.2 ± 3.0	23.0 ± 1.4	40.2 ± 1.5	58.8 ± 3.5	55.7 ± 2.5	**68.9 ± 1.3**	64.3 ± 1.6	35.5 ± 2.7	75.0 ± 3.8	**90.8**

† Below the per-domain in-domain-gain threshold; excluded from the headline forgetting comparison (Section 5.1).

on DocVQA and RealWorldQA, Table 10) degrade significantly under unregularized RL.

This disparity demonstrates that model parameter capacity alone cannot predict catastrophic forgetting. Below, we synthesize the diagnostic evidence across all nine cells into a predictive two-factor framework: (i) *chance-corrected pre-existing competence* and (ii) *task memorizability versus procedural planning*. Table 13 presents the cross-cell diagnostic matrix.

Factor 1: Chance-Corrected Base Competence versus Raw Accuracy. A naive inspection of raw zero-shot target accuracy would suggest that at 4B, the base model performs almost identically on Navigation (27.6%) and Puzzle (27.5%). However, evaluating accuracy relative to task chance level reveals that these numbers represent diametrically opposite internal states:

- **Navigation Answer Space (Combinatorial Path Space):** The Navigation prompt requires generating a comma-separated sequence of coordinate actions (e.g., `U, R, D, L, ...`) describing a multi-step path on an input grid. The probability of randomly guessing a valid multi-step sequence is effectively zero ($0.0\% \ll 10^{-4}$). The 4B base model's 27.6% zero-shot accuracy therefore represents *authentic, pre-existing spatial-procedural competence* acquired during multimodal pre-training.
- **Puzzle Answer Space (Categorical Choice Space):** In contrast, Puzzle prompts demand a single multiple-choice letter (A–E). The random guessing baseline is 1/4 to 1/5 (mean 22.5%). The base model's score of 27.5% is merely +5.0% above random

Table 5: Anchor-Budget Ablation across Out-of-Domain Benchmarks (Navigation, Qwen3-VL-2B). Evaluation of CFA with curated real-image coreset anchors (from external public training splits) versus isotropic Gaussian pseudo-inputs across scaling budgets $N \in \{320, 640, 1280, 2560\}$. We report accuracy (mean ± std) across all 10 common-suite evaluation benchmarks alongside final target in-domain accuracy. **Bold** indicates the best performance per benchmark; underline indicates the second best.

		STEM				Gen. VQA		Chart/Doc.			Ground.	
Budget (N)	Anchors	MMMU	MVi	MVn	MVe	RWQA	MMS	DocVQA	CX-DQ	CX-RQ	CBQA	Target
Base		48.4 ± 2.4	56.0 ± 3.1	21.1 ± 1.4	37.4 ± 1.5	58.4 ± 3.5	46.7 ± 2.5	68.6 ± 1.2	60.8 ± 1.5	28.0 ± 2.8	82.5 ± 3.4	3.5
$N = 320$	RDA	50.4 ± 2.4	67.5 ± 2.9	26.1 ± 1.6	47.0 ± 1.6	59.5 ± 3.5	59.9 ± 2.5	70.2 ± 1.2	56.3 ± 1.5	**30.7 ± 2.9**	74.8 ± 3.8	87.3
	CFA	50.6 ± 2.4	62.7 ± 3.0	28.4 ± 1.6	45.8 ± 1.6	49.9 ± 3.5	60.3 ± 2.5	63.5 ± 1.3	59.2 ± 1.5	28.9 ± 2.8	71.5 ± 4.0	83.8
$N = 640$	RDA	48.9 ± 2.4	64.1 ± 3.0	25.9 ± 1.6	43.9 ± 1.6	55.2 ± 3.5	59.6 ± 2.5	59.0 ± 1.3	58.8 ± 1.5	28.5 ± 2.8	61.5 ± 4.3	77.6
	CFA	51.2 ± 2.4	**67.7 ± 2.9**	**30.0 ± 1.6**	**47.1 ± 1.6**	54.5 ± 3.5	**61.3 ± 2.5**	69.1 ± 1.2	55.6 ± 1.5	28.9 ± 2.8	81.3 ± 3.5	86.4
$N = 1280$	RDA	**51.4 ± 2.4**	66.6 ± 2.9	29.6 ± 1.6	46.6 ± 1.6	**61.2 ± 3.5**	60.3 ± 2.5	**70.9 ± 1.2**	55.6 ± 1.5	29.9 ± 2.8	70.9 ± 4.0	85.7
	CFA	51.1 ± 2.4	63.6 ± 3.0	29.3 ± 1.6	46.8 ± 1.6	57.1 ± 3.5	59.9 ± 2.5	67.5 ± 1.3	61.1 ± 1.5	28.1 ± 2.8	79.6 ± 3.6	87.0
$N = 2560$	RDA	49.0 ± 2.4	64.5 ± 3.0	27.5 ± 1.6	44.9 ± 1.6	59.1 ± 3.5	59.3 ± 2.5	59.5 ± 1.3	56.8 ± 1.5	27.6 ± 2.8	74.5 ± 3.8	82.7
	CFA	52.4 ± 2.4	65.7 ± 2.9	29.3 ± 1.6	47.9 ± 1.6	59.7 ± 3.5	60.1 ± 2.5	71.2 ± 1.2	**67.0 ± 1.5**	29.0 ± 2.8	**79.8 ± 3.6**	**87.5**

Table 6: Divergence-Measure Ablation across Out-of-Domain Benchmarks (Navigation, Qwen3-VL-2B, $N = 1280$). Comparison of regularizer objectives at the anchor budget used throughout this paper: our proposed CFA/RDA, Fisher-weighted KL divergence, and Unweighted reference KL divergence, crossed with Real vs. Gaussian pseudo-inputs. **Bold** indicates top performance per column; underline indicates the second best.

		STEM				Gen. VQA		Chart/Doc.			**Ground.**	
Divergence	Anchors	MMMU	MVi	MVn	MVe	RWQA	MMS	DocVQA	CX-DQ	CX-RQ	CBQA	Target
Base		48.4 ± 2.4	56.0 ± 3.1	21.1 ± 1.4	37.4 ± 1.5	58.4 ± 3.5	46.7 ± 2.5	68.6 ± 1.2	60.8 ± 1.5	28.0 ± 2.8	82.5 ± 3.4	3.5
RDA (Ours)	Real	**51.4 ± 2.4**	**66.6 ± 2.9**	29.6 ± 1.6	46.6 ± 1.6	**61.2 ± 3.5**	60.3 ± 2.5	**70.9 ± 1.2**	55.6 ± 1.5	29.9 ± 2.8	70.9 ± 4.0	85.7
CFA (Ours)	Gaussian	51.1 ± 2.4	63.6 ± 3.0	29.3 ± 1.6	46.8 ± 1.6	57.1 ± 3.5	59.9 ± 2.5	67.5 ± 1.3	61.2 ± 1.5	28.1 ± 2.8	79.6 ± 3.6	87.0
Fisher-KL	Real	47.8 ± 2.4	62.3 ± 3.0	26.3 ± 1.6	45.1 ± 1.6	52.7 ± 3.5	**60.7 ± 2.5**	58.3 ± 1.3	60.1 ± 1.5	**31.4 ± 2.9**	77.6 ± 3.7	88.8
Fisher-KL	Gaussian	50.8 ± 2.4	65.0 ± 3.0	**30.7 ± 1.6**	**48.5 ± 1.6**	55.0 ± 3.5	60.0 ± 2.5	64.9 ± 1.3	**65.6 ± 1.5**	30.5 ± 2.9	**81.7 ± 3.4**	91.9
Unweighted-KL	Real	46.9 ± 2.4	62.4 ± 3.0	27.2 ± 1.6	43.4 ± 1.6	53.9 ± 3.5	59.4 ± 2.5	60.0 ± 1.3	58.5 ± 1.5	28.8 ± 2.8	69.9 ± 4.0	82.9
Unweighted-KL	Gaussian	51.4 ± 2.4	65.7 ± 3.0	29.5 ± 1.4	44.0 ± 1.3	59.3 ± 3.5	59.9 ± 2.5	67.9 ± 1.3	56.8 ± 1.5	29.3 ± 2.8	70.5 ± 3.6	84.2

chance. In other words, the base model possesses virtually no latent capability to solve visual puzzles.

Across all nine cells, above-chance base competence exhibits a strong positive correlation with tail retention under unregularized RL ($r = +0.395$). When a model lacks pre-existing competence (as in Puzzle at all scales, and Navigation at 2B where base accuracy is only 3.5%), reinforcement learning cannot simply refine existing capabilities; it is forced to perform extensive *representational restructuring* synthesizing novel perceptual and reasoning circuits from scratch under high gradient variance. This exploratory parameter update directly overwrites shared visual features, causing severe forgetting on out-of-domain benchmarks.

Factor 2: Procedural Planning versus Shortcut Memorizability (The SFT Probe). Why then does MedBookVQA at 4B suffer severe catastrophic forgetting (worst-1 drop of −25.0%), despite possessing +27.3% above-chance base accuracy (49.8% zero-shot accuracy)? The answer lies in the fundamental distinction between *procedural reasoning* and *static associative recall*, which is revealed by our supervised fine-tuning (SFT) probe:

- On MedBookVQA, standard SFT achieves **100.0%** target accuracy at 4B (95.1% at 2B, 99.8% at 8B; Table 8). MedBookVQA consists of static medical image-text queries paired with discrete option choices. The target domain can be completely memorized via superficial associative shortcuts between visual features and question tokens. When trained with unregularized RL (e.g., GRPO or GSPO), the sparse reward signal rapidly drives the policy toward low-entropy label memorization shortcuts. This collapses the output distribution's entropy and warps shared visual representations, leading to severe tail collapse on held-out benchmarks (e.g., RealWorldQA drops by −25.0% on GRPO and MMMU drops by −36.0% on GSPO).
- On Navigation, standard SFT achieves only **23.9%** target accuracy at 4B (and only 3.9% at 2B; Table 7), performing *worse* than the zero-shot base policy (27.6%). Navigation cannot be solved through static label memorization; it requires active, multi-step algorithmic execution over spatial coordinates.

Table 7: **Navigation, Qwen3-VL-4B.** Common-suite accuracy (mean ± std) for every baseline and for CFA at anchor budget $N = 1280$ (Real/Gaussian). **Bold** marks the best method per column; underline marks the second best. "Target" is final in-domain (target-task) accuracy.

	STEM				Gen. VQA		Chart/Doc.			Ground.	
Method	MMMU	MVi	MVn	MVe	RWQA	MMS	DocVQA	CX-DQ	CX-RQ	CBQA	Target
Base	62.8 ± 1.2	61.9 ± 1.5	39.6 ± 0.9	48.6 ± 0.8	68.5 ± 1.7	64.7 ± 1.2	64.4 ± 0.6	69.8 ± 0.7	35.1 ± 1.5	85.5 ± 1.6	27.6
GRPO (wd=0)	54.4 ± 1.2	59.3 ± 1.6	28.8 ± 0.8	43.7 ± 0.8	68.1 ± 1.7	65.1 ± 1.2	76.4 ± 0.6	74.0 ± 0.7	37.7 ± 1.5	82.5 ± 1.7	82.2
GRPO (wd=0.1)	64.2 ± 1.2	69.0 ± 1.5	43.6 ± 0.9	53.8 ± 0.8	68.5 ± 1.7	66.4 ± 1.2	78.3 ± 0.6	**78.9 ± 0.6**	40.2 ± 1.6	84.5 ± 1.6	93.4
GRPO+KL	62.6 ± 1.2	**74.4 ± 1.4**	**49.2 ± 0.9**	**57.6 ± 0.8**	64.2 ± 1.7	66.8 ± 1.2	77.7 ± 0.6	78.0 ± 0.7	38.4 ± 1.5	81.5 ± 1.8	93.9
GSPO	64.0 ± 1.2	68.8 ± 1.5	43.8 ± 0.9	53.6 ± 0.8	68.1 ± 1.7	**67.7 ± 1.2**	73.4 ± 0.6	75.2 ± 0.7	**41.3 ± 1.6**	82.6 ± 1.5	92.3
GSPO+EWC	63.0 ± 1.2	63.4 ± 1.5	40.4 ± 0.9	51.0 ± 0.8	68.5 ± 1.7	66.3 ± 1.2	70.8 ± 0.6	70.4 ± 0.7	37.2 ± 1.5	84.9 ± 1.6	93.6
LoRA	62.4 ± 2.3	68.4 ± 2.9	38.2 ± 1.7	53.9 ± 1.6	66.9 ± 3.3	64.7 ± 2.4	70.5 ± 1.2	72.0 ± 1.4	38.8 ± 3.0	84.7 ± 3.2	29.6†
SFT	58.3 ± 2.3	72.8 ± 2.8	39.3 ± 1.7	54.0 ± 1.6	67.8 ± 3.3	64.9 ± 2.4	74.5 ± 1.2	74.3 ± 1.4	37.3 ± 3.0	83.1 ± 3.3	23.9†
CPO	56.8 ± 2.3	65.7 ± 2.9	36.5 ± 1.7	51.6 ± 1.6	**68.8 ± 3.3**	66.1 ± 2.4	**78.8 ± 1.1**	76.4 ± 1.3	37.4 ± 3.0	84.5 ± 3.2	66.5†
RDA	**65.0 ± 1.2**	62.6 ± 1.5	43.5 ± 0.9	50.6 ± 0.8	68.4 ± 1.7	67.3 ± 1.2	67.7 ± 0.6	73.4 ± 0.7	38.4 ± 1.5	**88.2 ± 1.5**	91.9
CFA	62.9 ± 1.2	64.2 ± 1.5	42.5 ± 0.9	51.2 ± 0.8	68.4 ± 1.7	66.7 ± 1.2	69.4 ± 0.6	73.7 ± 0.7	37.7 ± 1.5	87.2 ± 1.5	**94.3**

† Below the per-domain in-domain-gain threshold; excluded from the headline forgetting comparison (Section 5.1).

Table 8: **MedBookVQA, Qwen3-VL-4B.** Common-suite accuracy (mean ± std) for every baseline and for CFA at anchor budget $N = 1280$ (Real/Gaussian). **Bold** marks the best method per column; underline marks the second best. "Target" is final in-domain (target-task) accuracy.

	STEM				Gen. VQA		Chart/Doc.			Ground.	
Method	MMMU	MVi	MVn	MVe	RWQA	MMS	DocVQA	CX-DQ	CX-RQ	CBQA	Target
Base	62.8 ± 1.2	61.9 ± 1.5	39.6 ± 0.9	48.6 ± 0.8	68.5 ± 1.7	64.7 ± 1.2	64.4 ± 0.7	69.8 ± 0.7	35.1 ± 1.5	85.5 ± 1.6	49.8
GRPO (wd=0)	54.4 ± 2.4	73.3 ± 2.7	41.5 ± 1.8	54.6 ± 1.6	60.3 ± 3.5	61.6 ± 2.5	76.6 ± 1.1	78.1 ± 1.3	39.2 ± 3.0	73.1 ± 3.9	99.2
GRPO (wd=0.1)	52.7 ± 2.4	71.2 ± 2.8	29.7 ± 1.6	43.9 ± 1.6	**68.6 ± 3.3**	**65.1 ± 2.4**	**80.3 ± 1.1**	69.6 ± 1.4	38.6 ± 3.0	86.4 ± 3.0	95.8
GRPO+KL	55.4 ± 1.2	70.6 ± 1.4	43.4 ± 0.9	54.2 ± 0.8	57.0 ± 1.8	62.8 ± 1.2	72.1 ± 0.6	71.2 ± 0.7	32.0 ± 1.5	71.3 ± 2.0	99.3
GSPO	40.2 ± 2.3	69.4 ± 2.9	36.8 ± 1.7	50.2 ± 1.6	54.1 ± 3.5	61.0 ± 2.5	65.3 ± 1.2	54.4 ± 1.5	32.1 ± 2.9	79.2 ± 3.6	99.8
GSPO+EWC	55.2 ± 2.3	72.4 ± 2.8	33.7 ± 1.9	53.6 ± 1.6	54.9 ± 3.5	61.1 ± 2.5	3.8 ± 0.5	70.7 ± 1.4	35.0 ± 3.0	69.7 ± 4.1	**100.0**
LoRA	**63.1 ± 2.3**	62.5 ± 3.0	39.0 ± 1.7	49.3 ± 1.6	**68.6 ± 3.3**	64.0 ± 2.4	64.6 ± 1.3	70.9 ± 1.4	36.8 ± 3.0	**87.0 ± 3.0**	53.3†
SFT	32.5 ± 2.2	50.5 ± 3.1	11.2 ± 1.1	21.8 ± 1.3	43.4 ± 3.5	38.6 ± 2.5	60.4 ± 1.3	65.4 ± 1.5	31.3 ± 2.9	73.3 ± 3.9	**100.0**
CPO	55.9 ± 2.3	71.7 ± 2.8	36.6 ± 1.7	51.8 ± 1.6	68.2 ± 3.3	63.6 ± 2.4	77.2 ± 1.1	73.1 ± 1.4	36.3 ± 3.0	85.3 ± 3.1	72.2†
RDA	54.4 ± 2.4	72.8 ± 2.8	40.9 ± 1.8	52.5 ± 1.6	59.2 ± 3.5	62.3 ± 2.5	72.9 ± 1.2	66.9 ± 1.5	35.6 ± 3.0	83.1 ± 3.3	99.8
CFA	61.1 ± 2.3	**75.3 ± 2.7**	**46.4 ± 1.8**	**58.5 ± 1.5**	66.0 ± 3.4	64.5 ± 2.4	79.2 ± 1.1	**81.8 ± 1.2**	**40.2 ± 3.0**	86.2 ± 3.1	99.7

† Below the per-domain in-domain-gain threshold; excluded from the headline forgetting comparison (Section 5.1).

The Goldilocks Calibration at 4B Navigation. Because Qwen3-VL-4B already possesses authentic spatial competence on Navigation (+27.6% above chance), and because the procedural nature of the task prevents collapse onto memorization shortcuts, reinforcement learning on Navigation operates via a fundamentally different mechanism: policy calibration rather than feature restructuring. The RL reward does not need to carve out new visual filters or reasoning pathways; instead, it acts as a selective filter that sharpens the sampling temperature over valid action tokens, aligns the output format, and suppresses trailing hallucinations. Crucially, this calibration of latent reasoning traces preserves the underlying visual encoder representations. In fact, this structured calibration yields substantial *positive transfer* to other structured, document-level reasoning domains: DocVQA improves by +13.9 points (from 64.4% to 78.3%) and ChartQA-DQ improves by +9.1 points (from 69.8% to 78.9%) on GRPO ($\lambda_{\text{wd}} = 0.1$).

Scale Dynamics at 8B Navigation: Why Functional Anchoring Remains Essential. Finally, this diagnostic framework explains why the larger Qwen3-VL-8B model still suffers catastrophic forgetting on Navigation under unregularized RL (worst-1 drops of −15.0% to −22.6%, Table 10), despite possessing comparable base competence (28.7%, +28.7% above chance). In an 8×10^9 parameter model, the parameter search space is vastly larger and over-parameterized. Without functional anchoring, unconstrained policy optimization permits unpenalized gradient drift in orthogonal subspaces that are irrelevant to the target navigation task but critical for fragile out-of-domain capabilities. This produces heterogeneous, baseline-specific failures:

- Plain GRPO ($\lambda_{\text{wd}} = 0$) suffers representational drift on dense document parsing (DocVQA drops from 67.8% to 55.2%, a −18.6% relative loss).
- GSPO collapses on general visual understanding (RealWorldQA drops from 67.7% to 52.4%, a −22.6% loss).
- GRPO+KL suffers significant grounding degradation (CountBenchQA drops from 85.7% to 73.5%, a −14.2% loss).

Table 9: **Puzzle, Qwen3-VL-4B.** Common-suite accuracy (mean ± std) for every baseline and for CFA at anchor budget $N = 1280$ (Real/Gaussian). **Bold** marks the best method per column; underline marks the second best. "Target" is final in-domain (target-task) accuracy.

	STEM				Gen. VQA		Chart/Doc.			Ground.	
Method	MMMU	MVi	MVn	MVe	RWQA	MMS	DocVQA	CX-DQ	CX-RQ	CBQA	Target
Base	62.8 ± 1.2	61.9 ± 1.5	39.6 ± 0.9	48.6 ± 0.8	**68.5 ± 1.7**	64.7 ± 1.2	64.4 ± 0.6	69.8 ± 0.7	35.1 ± 1.5	85.5 ± 1.6	27.5
GRPO (wd=0)	53.7 ± 2.3	67.3 ± 2.9	24.9 ± 1.6	49.1 ± 1.6	34.1 ± 3.5	62.0 ± 2.5	74.4 ± 1.2	62.7 ± 1.5	35.2 ± 2.9	56.8 ± 4.4	68.2†
GRPO (wd=0.1)	53.0 ± 2.4	70.9 ± 2.8	29.4 ± 1.6	46.4 ± 1.6	53.9 ± 3.5	61.3 ± 2.5	67.7 ± 1.2	64.4 ± 1.5	40.4 ± 3.0	85.1 ± 3.2	88.2
GRPO+KL	60.9 ± 1.2	73.7 ± 1.4	**46.8 ± 0.9**	**56.6 ± 0.8**	63.3 ± 1.7	66.9 ± 1.2	76.9 ± 0.6	69.5 ± 0.7	41.3 ± 1.6	78.8 ± 1.8	88.8
GSPO	52.1 ± 2.4	66.7 ± 2.9	24.2 ± 1.5	46.2 ± 1.5	58.7 ± 3.5	58.3 ± 2.5	59.5 ± 1.3	51.1 ± 1.6	30.2 ± 2.8	64.0 ± 4.0	90.8
GSPO+EWC	21.0 ± 1.9	58.2 ± 3.1	21.4 ± 1.4	38.4 ± 1.6	61.1 ± 3.7	51.9 ± 2.5	65.6 ± 1.3	54.2 ± 1.5	28.9 ± 2.8	61.9 ± 4.3	**91.0**
LoRA	**63.4 ± 2.3**	61.8 ± 3.0	39.8 ± 1.5	49.3 ± 1.5	67.1 ± 3.3	65.2 ± 2.4	65.7 ± 1.2	70.3 ± 1.4	35.7 ± 2.9	**86.0 ± 3.0**	26.2†
SFT	50.8 ± 2.4	66.9 ± 2.9	19.7 ± 1.4	54.5 ± 1.6	54.1 ± 3.5	44.2 ± 2.5	50.2 ± 1.4	**76.1 ± 1.4**	33.7 ± 3.0	72.3 ± 4.0	19.5†
CPO	32.1 ± 2.2	58.4 ± 3.0	19.6 ± 1.4	37.9 ± 1.5	63.1 ± 3.4	61.0 ± 2.5	60.7 ± 1.3	59.7 ± 1.5	31.1 ± 2.9	48.8 ± 4.4	69.2†
RDA	60.4 ± 2.3	**75.5 ± 2.6**	42.3 ± 1.7	55.6 ± 1.6	60.6 ± 3.3	**67.7 ± 2.4**	**77.6 ± 1.1**	75.0 ± 1.4	**41.4 ± 3.0**	78.0 ± 3.7	**91.0**
CFA	64.2 ± 2.3	72.5 ± 2.8	46.3 ± 1.7	52.6 ± 1.6	64.7 ± 3.6	65.1 ± 2.3	75.0 ± 1.1	74.1 ± 1.4	40.1 ± 3.1	84.9 ± 3.3	90.0

† Below the per-domain in-domain-gain threshold; excluded from the headline forgetting comparison (Section 5.1).

Table 10: **Navigation, Qwen3-VL-8B.** Common-suite accuracy (mean ± std) for every baseline and for CFA at anchor budget $N = 1280$ (Real/Gaussian). **Bold** marks the best method per column; underline marks the second best. "Target" is final in-domain (target-task) accuracy.

	STEM				Gen. VQA		Chart/Doc.			Ground.	
Method	MMMU	MVi	MVn	MVe	RWQA	MMS	DocVQA	CX-DQ	CX-RQ	CBQA	Target
Base	**68.0 ± 1.1**	53.4 ± 1.6	43.5 ± 0.9	48.8 ± 0.8	67.7 ± 1.7	57.4 ± 1.3	67.8 ± 0.6	79.2 ± 0.6	37.6 ± 1.5	85.7 ± 1.6	28.7
GRPO (wd=0)	65.6 ± 1.1	55.6 ± 1.6	44.8 ± 0.9	47.2 ± 0.8	64.7 ± 1.7	58.4 ± 1.3	55.2 ± 0.7	78.5 ± 0.6	32.1 ± 1.5	85.3 ± 1.6	87.9
GRPO (wd=0.1)	66.0 ± 1.1	58.7 ± 1.6	46.4 ± 0.9	64.7 ± 1.2	64.7 ± 1.7	64.7 ± 1.2	57.6 ± 0.7	79.2 ± 0.6	34.1 ± 1.5	86.0 ± 3.0	89.7
GRPO+KL	64.7 ± 1.2	74.3 ± 1.4	**48.6 ± 0.9**	56.6 ± 0.8	63.9 ± 1.7	**70.5 ± 1.2**	67.8 ± 0.6	78.0 ± 0.7	38.9 ± 1.5	73.5 ± 2.0	93.4
GSPO	62.5 ± 1.2	73.5 ± 1.4	39.5 ± 0.9	**66.5 ± 1.2**	52.4 ± 1.8	66.5 ± 1.2	62.9 ± 0.7	71.4 ± 0.7	37.3 ± 1.5	83.3 ± 3.2	96.0
GSPO+EWC	60.9 ± 1.2	69.2 ± 1.5	39.1 ± 0.9	52.9 ± 0.8	56.1 ± 1.8	65.5 ± 1.2	64.8 ± 0.6	75.0 ± 0.7	37.0 ± 1.5	79.4 ± 1.8	93.6
LoRA	66.8 ± 2.2	53.9 ± 3.1	43.3 ± 1.8	48.6 ± 1.6	66.9 ± 3.3	60.0 ± 2.5	69.8 ± 1.2	80.6 ± 1.2	37.8 ± 3.0	84.5 ± 3.2	27.2†
SFT	62.2 ± 2.3	62.5 ± 3.0	18.8 ± 1.4	17.7 ± 1.2	66.9 ± 3.3	62.5 ± 2.5	**83.6 ± 1.0**	70.9 ± 1.4	40.1 ± 3.0	76.2 ± 3.8	58.1†
CPO	66.4 ± 2.4	62.2 ± 3.0	33.0 ± 1.7	47.9 ± 1.6	65.0 ± 3.4	66.4 ± 2.4	66.2 ± 1.3	78.3 ± 1.3	39.1 ± 3.0	84.9 ± 3.2	84.9
RDA	66.2 ± 1.1	70.3 ± 1.4	47.9 ± 0.9	57.6 ± 0.8	59.5 ± 1.8	68.1 ± 1.2	67.1 ± 0.6	**80.9 ± 0.6**	**41.3 ± 1.6**	85.3 ± 1.6	**98.4**
CFA	65.9 ± 1.1	**76.0 ± 1.4**	47.8 ± 0.9	56.9 ± 0.8	**68.7 ± 1.8**	69.1 ± 1.2	66.8 ± 0.6	79.8 ± 0.7	41.0 ± 1.6	**86.2 ± 1.6**	97.1

† Below the per-domain in-domain-gain threshold; excluded from the headline forgetting comparison (Section 5.1).

By contrast, CFA enforces functional invariance on a compact set of pseudo-inputs without requiring curated out-of-domain data. As shown in Table 13 and Table 10, CFA constrains this high-dimensional optimization drift, holding worst-1 degradation to just −3.1% (MMMU remains at 65.9% vs. 68.0% base) while achieving an elite target accuracy of 97.1%.

D Theoretical Results and Extended Discussion

This appendix follows one continuous argument in six parts: (D.1) what assumption every prior forgetting defense secretly shares; (D.2) why that assumption is exactly EWC in disguise; (D.3) a trust radius bounding when that assumption is accurate, and a self-ablation showing the frozen-curvature shortcut it would license is not free, even under CFA's own comparatively well-conditioned metric; (D.4) what an anchor set actually needs to supply once the quadratic approximation is dropped; (D.5) why that requirement lets content-free noise eventually beat curated data, with the covering-radius view of the crossover; and (D.6) a side-by-side summary of EWC and CFA.

D.1 The Hidden Assumption Behind Every Prior Defense

Given a context c, let $\pi_\theta = \pi_\theta(\cdot \mid c) \in \Delta^{V-1}$, $\pi^* = \pi^*(\cdot \mid c)$, and $\Delta\pi = \pi_\theta - \pi^*$.

Proposition 1 (Quadratic form on the simplex tangent space)**.** *Let $D(\pi^*\|\pi)$ be any twice-differentiable divergence on the probability simplex Δ^{V-1} with $D(\pi^*\|\pi^*) = 0$. For any admissible variation $\Delta\pi = \pi - \pi^*$ constrained to the tangent space $T_{\pi^*}\Delta^{V-1} = \{u \in \mathbb{R}^V : \mathbf{1}^\top u = \sum_v u_v = 0\}$, the first-order directional variation vanishes along the manifold, $\langle \nabla_\pi D(\pi^*\|\pi)|_{\pi^*}, \Delta\pi\rangle = 0$. Then, for $\pi \in \Delta^{V-1}$ in a neighborhood of π^*,*

$$D(\pi^*\|\pi) = \tfrac{1}{2}\,\Delta\pi^\top M_D\,\Delta\pi + \mathcal{O}(\|\Delta\pi\|^3), \qquad M_D := \nabla^2_\pi D(\pi^*\|\pi)\big|_{\pi=\pi^*}. \tag{3}$$

Table 11: **MedBookVQA, Qwen3-VL-8B.** Common-suite accuracy (mean ± std) for every baseline and for CFA at anchor budget $N = 1280$ (Real/Gaussian). **Bold** marks the best method per column; underline marks the second best. "Target" is final in-domain (target-task) accuracy.

	STEM				Gen. VQA		Chart/Doc.			Ground.	
Method	MMMU	MVi	MVn	MVe	RWQA	MMS	DocVQA	CX-DQ	CX-RQ	CBQA	Target
Base	**68.0 ± 1.1**	53.4 ± 1.6	43.5 ± 0.9	48.8 ± 0.8	**67.7 ± 1.7**	57.4 ± 1.3	67.8 ± 0.6	79.2 ± 0.6	37.6 ± 1.5	85.7 ± 1.6	44.1
GRPO (wd=0)	61.2 ± 1.2	72.8 ± 1.4	35.2 ± 0.9	54.6 ± 0.8	63.0 ± 1.8	62.6 ± 1.2	53.0 ± 0.7	78.8 ± 0.6	38.3 ± 1.5	76.2 ± 1.9	98.8
GRPO (wd=0.1)	62.5 ± 1.2	76.2 ± 1.4	51.6 ± 0.9	58.6 ± 0.8	63.7 ± 1.7	66.7 ± 1.2	80.0 ± 0.6	84.4 ± 0.6	40.6 ± 1.6	81.5 ± 1.8	99.5
GRPO+KL	63.6 ± 1.2	74.1 ± 1.4	49.4 ± 0.9	55.7 ± 0.8	64.4 ± 1.7	66.1 ± 1.2	69.7 ± 0.6	73.2 ± 0.7	38.5 ± 1.5	81.9 ± 1.7	99.7
GSPO	58.4 ± 2.3	76.1 ± 2.6	29.1 ± 1.6	53.9 ± 2.3	56.0 ± 2.6	65.6 ± 2.4	31.7 ± 1.2	69.9 ± 1.4	40.9 ± 3.1	82.5 ± 3.4	99.7
GSPO+EWC	61.9 ± 1.2	74.4 ± 1.4	39.3 ± 0.9	55.8 ± 0.8	60.4 ± 1.8	68.3 ± 1.2	65.5 ± 0.6	85.1 ± 0.6	44.1 ± 1.6	85.3 ± 1.6	99.0
LoRA	66.9 ± 2.2	64.3 ± 2.9	43.1 ± 1.8	50.6 ± 1.6	66.5 ± 3.3	66.4 ± 2.4	**81.7 ± 1.0**	79.7 ± 1.2	39.5 ± 3.0	86.0 ± 3.1	88.0†
SFT	25.7 ± 2.1	78.3 ± 1.7	10.8 ± 1.1	9.9 ± 0.9	16.9 ± 2.7	22.1 ± 2.1	0.0 ± 0.0	0.0 ± 0.0	0.2 ± 0.3	0.0 ± 0.0	99.8
CPO	60.3 ± 2.3	71.3 ± 3.0	25.8 ± 1.5	53.5 ± 1.6	62.9 ± 3.5	63.9 ± 2.4	23.9 ± 1.1	81.6 ± 1.2	39.2 ± 3.1	81.7 ± 3.5	99.7
RDA	65.6 ± 2.2	**79.1 ± 2.5**	**52.6 ± 1.8**	**60.6 ± 1.5**	66.4 ± 3.4	67.1 ± 2.4	79.7 ± 1.1	**85.5 ± 1.1**	44.5 ± 3.1	83.5 ± 3.3	98.6
CFA	67.4 ± 2.2	78.3 ± 2.6	51.7 ± 1.8	60.1 ± 1.5	66.5 ± 3.5	**71.3 ± 2.3**	77.4 ± 1.1	85.2 ± 1.1	**45.6 ± 3.1**	**87.4 ± 1.5**	**100.0**

† Below the per-domain in-domain-gain threshold; excluded from the headline forgetting comparison (Section 5.1).

Table 12: **Puzzle, Qwen3-VL-8B.** Common-suite accuracy (mean ± std) for every baseline and for CFA at anchor budget $N = 1280$ (Real/Gaussian). **Bold** marks the best method per column; underline marks the second best. "Target" is final in-domain (target-task) accuracy.

	STEM				Gen. VQA		Chart/Doc.			Ground.	
Method	MMMU	MVi	MVn	MVe	RWQA	MMS	DocVQA	CX-DQ	CX-RQ	CBQA	Target
Base	**68.0 ± 1.1**	53.4 ± 1.6	43.5 ± 0.9	48.8 ± 0.8	**67.7 ± 1.7**	57.4 ± 1.3	67.8 ± 0.6	79.2 ± 0.6	37.6 ± 1.5	**85.7 ± 1.6**	28.8
GRPO (wd=0)	60.6 ± 2.3	73.8 ± 2.7	39.6 ± 1.7	54.7 ± 1.6	48.6 ± 3.5	65.9 ± 2.4	75.1 ± 1.2	81.3 ± 1.2	40.8 ± 3.1	80.5 ± 3.5	87.0
GRPO (wd=0.1)	62.5 ± 1.2	70.2 ± 1.5	43.7 ± 0.9	53.5 ± 0.8	63.5 ± 1.7	66.2 ± 1.2	74.9 ± 0.6	78.7 ± 0.7	38.3 ± 1.5	61.3 ± 2.2	90.8
GRPO+KL	62.7 ± 1.2	71.7 ± 1.4	40.9 ± 0.9	54.8 ± 0.8	61.2 ± 1.8	67.4 ± 1.2	73.1 ± 0.6	80.3 ± 0.6	41.2 ± 1.6	79.0 ± 1.8	93.3
GSPO	62.3 ± 2.3	72.8 ± 2.8	37.0 ± 1.7	55.4 ± 1.6	56.7 ± 3.5	64.3 ± 2.4	41.4 ± 1.3	76.6 ± 1.3	40.8 ± 3.1	76.2 ± 3.8	**95.0**
GSPO+EWC	57.0 ± 1.2	66.3 ± 1.5	29.8 ± 0.8	49.1 ± 0.8	51.5 ± 1.8	62.9 ± 1.2	53.8 ± 0.7	79.0 ± 0.6	36.4 ± 1.5	61.1 ± 2.2	94.5
LoRA	65.9 ± 2.2	59.8 ± 3.0	43.1 ± 1.8	52.4 ± 1.6	63.4 ± 2.2	59.6 ± 2.5	72.3 ± 1.2	73.6 ± 1.4	32.9 ± 2.9	53.0 ± 4.4	90.5
SFT	25.3 ± 2.1	15.9 ± 2.3	10.5 ± 1.1	10.7 ± 1.0	19.4 ± 2.8	22.5 ± 2.1	0.1 ± 0.1	0.0 ± 0.1	10.2 ± 0.3	0.0 ± 0.0	35.0†
CPO	64.5 ± 2.3	73.0 ± 2.8	43.5 ± 1.8	57.1 ± 1.6	64.2 ± 3.4	**68.3 ± 2.3**	57.7 ± 1.3	82.4 ± 1.2	42.6 ± 3.1	47.5 ± 4.5	89.3
RDA	64.9 ± 1.2	**76.4 ± 1.3**	44.7 ± 0.9	56.9 ± 0.8	65.2 ± 1.7	67.6 ± 1.2	77.9 ± 0.6	**83.0 ± 0.6**	40.8 ± 1.6	79.6 ± 1.8	90.3
CFA	65.0 ± 1.2	75.5 ± 1.4	**45.3 ± 0.9**	**58.1 ± 0.8**	67.3 ± 1.7	67.1 ± 1.2	**81.8 ± 0.5**	82.9 ± 0.6	**45.0 ± 1.6**	84.9 ± 1.6	93.0

† Below the per-domain in-domain-gain threshold; excluded from the headline forgetting comparison (Section 5.1).

Proof. Immediate second-order Taylor expansion of $D(\pi^* \| \cdot)$ around π^*; the zeroth- and first-order terms vanish by assumption, leaving the quadratic term and a Lagrange remainder $O(\|\Delta\pi\|^3)$ controlled by the third derivative of D. □

Any divergence minimized exactly at π^* looks, up close along the simplex tangent space, like a quadratic bowl $\frac{1}{2}\Delta\pi^\top M_D \Delta\pi$; all of the divergence's specifics compress into which directions M_D penalizes more heavily which is exactly the hidden assumption every prior method makes without stating it.

Proposition 2 (Discrete KL induces the Fisher-Rao metric on the tangent space)**.** *For discrete KL divergence $D = \mathrm{KL}(\pi^* \| \pi) = \sum_v \pi^*_v \log(\pi^*_v/\pi_v)$, the unconstrained ambient Euclidean gradient is $\nabla_\pi D|_{\pi^*} = -\mathbf{1}$. Because probability conservation restricts variations to the tangent space $T_{\pi^*}\Delta^{V-1}$ where $\mathbf{1}^\top \Delta\pi = 0$, the linear variation $\langle -\mathbf{1}, \Delta\pi \rangle$ vanishes identically, and the Hessian evaluates to $M_{\mathrm{KL}} = \mathrm{diag}(1/\pi^*_v)$.*

Proof. Differentiating term by term, $\partial D/\partial \pi_v = -\pi^*_v/\pi_v$ for each v, since only the v-th summand of D depends on π_v. Because this expression depends only on π_v itself and not on any other coordinate $\pi_{v'}$ with $v' \neq v$, its derivative with respect to $\pi_{v'}$ vanishes whenever $v' \neq v$:

$$\frac{\partial^2 D}{\partial \pi_v \, \partial \pi_{v'}} = \begin{cases} \pi^*_v/\pi_v^2 & v' = v, \\ 0 & v' \neq v, \end{cases} \tag{4}$$

so the Hessian $\nabla^2_\pi D$ is diagonal. Evaluating at $\pi = \pi^*$ gives $M_{\mathrm{KL}} = \mathrm{diag}(1/\pi^*_v)$, the Fisher-Rao (information) metric on the simplex. □

KL is therefore harshest about probability shifts near tokens the reference already considers rare (small π^*_v makes $1/\pi^*_v$ large), and does not account for the semantic meaning of tokens.

Proposition 3 (Embedding MMD is already quadratic)**.** *For $D = \mathrm{MMD}^2_\phi(\pi^*, \pi) = \Delta\pi^\top K \, \Delta\pi$ with $K = \frac{1}{d} E E^\top$ as in Eq. 2, $M_{\mathrm{MMD}} = 2K$ exactly, with zero remainder.*

Table 13: **Diagnostic Framework for Domain and Scale Dynamics.** Comparison across all nine experimental cells (3 domains × 3 scales). "Chance" denotes the random-guess baseline determined by the answer space (prompts detailed in Section 5.1 and Appendix H). "Above Chance" is Base Target Accuracy minus Chance. "SFT Target" serves as an empirical probe of task memorizability. Unregularized RL performance is illustrated by canonical GRPO ($\lambda_{\text{wd}} = 0.1$). **Bold** highlights the Navigation 4B anomaly and top retention scores.

		Pre-Existing Base Competence			Memorizability	Unregularized RL (GRPO)		CFA (Ours)	
Target Domain	Scale	Base Target	Chance Level	Above Chance	SFT Target Acc	Target Acc	Worst-1 Shift	Target Acc	Worst-1 Shift
Navigation	2B	3.5%	0.0%	+3.5%	3.9%	70.8%	-40.5%	87.0%	-3.5%
	4B	**27.6%**	**0.0%**	**+27.6%**	**23.9%**	**93.4%**	**-1.2%**	**94.3%**	**-0.1%**
	8B	28.7%	0.0%	+28.7%	58.1%	89.7%	-15.0%	97.1%	-3.1%
MedBookVQA	2B	49.4%	22.5%	+26.9%	95.1%	94.7%	-31.4%	98.5%	-0.3%
	4B	49.8%	22.5%	+27.3%	100.0%	95.8%	-25.0%	99.7%	-3.6%
	8B	44.1%	22.5%	+21.6%	99.8%	99.5%	-8.1%	100.0%	-1.8%
Puzzle	2B	23.8%	22.5%	+1.3%	33.0%	82.2%	-24.8%	90.8%	-9.1%
	4B	27.5%	22.5%	+5.0%	19.5%	88.2%	-25.8%	90.0%	-5.5%
	8B	28.8%	22.5%	+6.3%	35.0%	90.8%	-28.5%	93.0%	-4.4%

Proof. D is already an exact quadratic form in $\Delta\pi$, so $\nabla^2_\pi D = 2K$ everywhere on Δ^{V-1}, not merely at $\pi = \pi^*$; the Taylor remainder in Proposition 1 is identically zero. □

Both discrete KL and CFA's embedding MMD are therefore instances of the same template, $D(\pi^*\|\pi) \approx \frac{1}{2}\Delta\pi^\top M\,\Delta\pi$, differing only in the choice of metric M: $\text{diag}(1/\pi^*)$ for discrete KL, versus $2K$ for CFA.

Rank deficiency and null-space invariance of K. A fundamental geometric property distinguishes $K = \frac{1}{d}EE^\top$ from discrete KL's Fisher-Rao metric $M_{\text{KL}} = \text{diag}(1/\pi^*)$. In modern vision-language models such as Qwen3-VL, vocabulary size ($V \approx 152{,}000$) vastly exceeds hidden dimension ($d \in [2048, 4096]$). Consequently, $\text{rank}(K) \le d \ll V$, giving K an unpenalized null space $\ker(E^\top)$ of dimension $V - d > 147{,}000$: any distribution shift $\Delta\pi \in \ker(E^\top)$ satisfies $E^\top\Delta\pi = \mathbf{0}$ and produces zero MMD loss. This low-rank structure reflects an intentional and beneficial inductive bias: CFA regularizes drift in expected semantic feature space $\mathbb{E}_{v\sim\pi}[\phi(v)] = \frac{1}{\sqrt{d}}E^\top\pi$, tolerating synonymous paraphrasing among related tokens while penalizing shifts that alter semantic meaning. Because language model logits are parameterized as $z = Eh$, policy probability shifts are intrinsically driven by representations within $\text{col}(E)$, so unpenalized drift orthogonal to the token embeddings does not destabilize generation.

D.2 Pullback to Parameter Space: Generalizing EWC

The quadratic template above lives in output space. Pulling it back through the parameter Jacobian $J_\pi := \partial\pi_\theta(\cdot \mid c)/\partial\theta\big|_{\theta^*} \in \mathbb{R}^{V\times P}$ shows that the hidden assumption in D.1 is not an abstract curiosity but exactly the machinery behind Elastic Weight Consolidation. To leading order, $\Delta\pi \approx J_\pi\,\Delta\theta$, so substituting into Proposition 1's template gives, for any metric M,

$$D(\pi^*\|\pi_\theta) \approx \tfrac{1}{2}\,\Delta\theta^\top\big(J_\pi^\top M J_\pi\big)\,\Delta\theta. \tag{5}$$

Corollary 1 (Discrete-KL regularization is Fisher-weighted EWC)**.** *Substituting $M_{\text{KL}} = \text{diag}(1/\pi^*)$ (Proposition 2) into Eq. 5 gives*

$$\begin{aligned}\text{KL}(\pi^*\|\pi_\theta) &\approx \tfrac{1}{2}\,\Delta\theta^\top F(\theta^*)\,\Delta\theta,\\ F(\theta^*) &= J_\pi^\top\,\text{diag}(1/\pi^*)\,J_\pi = \mathbb{E}_{v\sim\pi^*}\big[\nabla_\theta\log\pi^*(v)\,\nabla_\theta\log\pi^*(v)^\top\big],\end{aligned} \tag{6}$$

which is exactly the (true) Fisher information matrix at θ^*: the population quantity EWC's diagonal empirical-Fisher penalty is designed to approximate (Kirkpatrick et al., 2017).

Proof. The middle equality substitutes M_{KL} into Eq. 5. The rightmost equality is the standard Fisher-information identity for a categorical output: since $\nabla_\theta\log\pi^*(v) =$

$\nabla_\theta \pi^*(v)/\pi^*(v)$, we have

$$\begin{aligned}\mathbb{E}_{v\sim\pi^*}\left[\nabla_\theta \log \pi^*(v)\, \nabla_\theta \log \pi^*(v)^\top\right] &= \sum_v \pi^*(v) \cdot \frac{\nabla_\theta \pi^*(v)}{\pi^*(v)} \frac{\nabla_\theta \pi^*(v)^\top}{\pi^*(v)} \\ &= \sum_v \frac{1}{\pi^*(v)} \nabla_\theta \pi^*(v) \nabla_\theta \pi^*(v)^\top \\ &= J_\pi^\top \operatorname{diag}(1/\pi^*) J_\pi.\end{aligned}$$

□

Regularizing with plain KL toward the reference is, to leading order, exactly what EWC does: penalizing movement along directions the Fisher information matrix considers important.

Corollary 2 (CFA generalizes EWC's metric, with varentropy weighting)**.** *Substituting $M_{\mathrm{CFA}} = 2K = \frac{2}{d} EE^\top$ (Proposition 3) into Eq. 5 cancels the factor of $\frac{1}{2}$, yielding $\Delta\theta^\top (J_\pi^\top K J_\pi)\Delta\theta$. Summing over anchor positions per Eq. 2 gives*

$$\mathcal{L}_{anchor}(\theta) = \Delta\theta^\top G_{emb}\, \Delta\theta + \mathcal{O}(\|\Delta\theta\|^3), \qquad G_{emb} = \frac{\lambda}{NT} \sum_{i=1}^{N} \sum_{t=1}^{T} \tilde{w}_t^{(i)}\; J_{\pi,t}^{(i)\top}\, K\, J_{\pi,t}^{(i)}. \tag{7}$$

Since $\tilde{w}_t^{(i)} \geq 0$ and $K \succeq 0$, each summand is positive semi-definite and hence $G_{emb} \succeq 0$. Setting $\tilde{w}_t^{(i)} \equiv 1$ recovers the unweighted Gauss-Newton form. CFA and EWC are therefore both of the form $\Delta\theta^\top (J_\pi^\top M J_\pi)\Delta\theta$, with $M_{\mathrm{KL}} = \operatorname{diag}(1/\pi^*)$ (the reference-dependent Fisher-Rao metric) replaced by $M_{CFA} = K$ (a fixed, reference-independent, semantic metric on tokens determined by the embedding table), and with an adaptive non-negative reweighting across positions.

Proof. Direct substitution of Proposition 3 into Eq. 5, summed over anchor positions per Eq. 2; positive semi-definiteness (PSD) follows since a non-negative combination of PSD matrices is PSD. □

Remark 1 (What the varentropy weighting does and does not change)**.** *Because the weights are normalized so that $\sum_t \tilde{w}_t^{(i)} = T$ (Eq. 2), the weighting redistributes curvature mass across token positions without changing its total scale. The formal Gauss-Newton equivalence in Corollary 2 holds strictly for uniform weights ($\tilde{w}_t^{(i)} \equiv 1$); varentropy weighting acts as an adaptive importance-weighting heuristic that concentrates regularization on positions where the reference policy is genuinely uncertain (decision forks), and preserves positive semi-definiteness and every concentration result in Sections D.4–D.5, since those depend on the weights only through a uniform upper bound $\tilde{w}_t^{(i)} \leq W$.*

CFA is thus built from the identical recipe as EWC. The only things that change are which matrix is used where EWC uses the Fisher-Rao metric $\operatorname{diag}(1/\pi^*)$, or CFA's embedding Gram matrix K and how positions are weighted. This is precisely why the two methods can be compared at all, and precisely why the comparison in D.3 below is fair: both $F(\theta^*)$ and G_{emb} are valid curvature matrices for a local quadratic trust region around θ^*, the question is what happens once training leaves that region.

D.3 A Trust Radius for the Quadratic Approximation

Everything above is a second-order approximation: exact only when θ barely moves from θ^*. This "lazy training" regime (Jacot et al., 2018) is exactly what EWC was designed for. Rather than argue qualitatively that RL leaves this regime, we make the condition explicit via a *trust radius* ρ^* inside which the quadratic form is accurate to any desired relative tolerance. This bound is stated for a general metric M and applies equally to CFA's K and to any other curvature matrix computed the same way, including EWC's Fisher-Rao metric.

Proposition 4 (Trust radius for the quadratic approximation)**.** *Write* $\|x\|_K^2 := x^\top K x$ *and let* $\kappa := \|K\|_{\mathrm{op}}$*. Suppose the policy map* $\theta \mapsto \pi_\theta(\cdot \mid c)$ *is twice differentiable with* $\|\nabla_\theta^2 \pi_\theta\|_{\mathrm{op}} \le H$ *on the segment between* θ^* *and* θ*, and suppose the pulled-back map is non-degenerate in the sense that* $\|J_\pi \Delta\theta\|_K \ge \mu \|\Delta\theta\|_2$ *for some* $\mu > 0$ *on the relevant subspace. Define the trust radius* $\rho^* := \mu/(\sqrt{\kappa} H)$*. Then the relative error of the quadratic approximation,*

$$\varepsilon \;:=\; \frac{\left|\, \mathrm{MMD}_\phi^2(\pi_\theta, \pi^*) - \Delta\theta^\top J_\pi^\top K J_\pi \Delta\theta \,\right|}{\Delta\theta^\top J_\pi^\top K J_\pi \Delta\theta}, \tag{8}$$

satisfies $\varepsilon \le u + u^2/4$ *with* $u := \|\Delta\theta\|_2/\rho^*$*. In particular,* $\varepsilon \le \epsilon$ *for any target* $\epsilon \in (0, 1]$ *whenever* $\|\Delta\theta\|_2 \le (\epsilon/2)\,\rho^*$*.*

Proof. Write $\Delta\pi = J_\pi \Delta\theta + r$. Taylor's theorem with the stated curvature bound gives $\|r\|_2 \le \frac{H}{2}\|\Delta\theta\|_2^2$, and hence $\|r\|_K \le \sqrt{\kappa}\,\frac{H}{2}\|\Delta\theta\|_2^2$. Put $a := \|J_\pi \Delta\theta\|_K$. By Proposition 3 the divergence is exactly $\|\Delta\pi\|_K^2$, so

$$\left|\, \|\Delta\pi\|_K^2 - a^2 \,\right| = \left|\, (\|\Delta\pi\|_K + a)(\|\Delta\pi\|_K - a) \,\right| \le (2a + \|r\|_K)\,\|r\|_K,$$

using the reverse triangle inequality $|\|\Delta\pi\|_K - a| \le \|r\|_K$. Dividing by a^2 and applying $a \ge \mu\|\Delta\theta\|_2$,

$$\varepsilon \;\le\; \frac{\sqrt{\kappa} H \|\Delta\theta\|_2^2}{a} + \frac{\kappa H^2 \|\Delta\theta\|_2^4}{4a^2} \;\le\; \frac{\sqrt{\kappa} H}{\mu}\|\Delta\theta\|_2 + \frac{\kappa H^2}{4\mu^2}\|\Delta\theta\|_2^2 \;=\; u + \tfrac{u^2}{4}.$$

Setting $u = \epsilon/2$ yields $\varepsilon \le \epsilon/2 + \epsilon^2/16 \le \epsilon$ for $\epsilon \le 1$. □

Proposition 4 converts "the quadratic truncation stops being trustworthy once training moves far enough" into a checkable criterion: the quadratic penalty is accurate precisely while $\|\Delta\theta\|_2 \lesssim \rho^*$. Two consequences follow for *any* method that relies on this truncation including EWC, since Corollary 1 places it under the same template. **(a)** Verifier-driven RLVR runs hundreds of policy-gradient steps at RL-scale learning rates; the displacement accumulates as $\|\Delta\theta\|_2 \le \sum_{s \le S} \eta_s \|g_s\|_2$, which grows with the step budget S and is not controlled by any term in the RL objective, leading to a *feature-learning* or *rich* regime in which the assumption $\|\Delta\theta\| \to 0$ underlying a second-order truncation need not hold. The non-degeneracy constant μ also shrinks when the policy approaches a near-deterministic output, which is precisely what happens when a group of rollouts collapses during training, shrinking ρ^* exactly when $\|\Delta\theta\|$ is largest. **(b)** A curvature matrix computed once, at θ^*, and held fixed for the whole run is a fixed ellipsoid, blind to how the true curvature changes as training moves away from θ^* applies to both $F(\theta^*)$ (EWC), and to G_{emb} as well. CFA does not take this shortcut: Eq. 2 evaluates π_θ *exactly* at every step, so it never forms or freezes G_{emb}, at the cost of the ∼15% per-step overhead reported in Appendix G.

Is that overhead actually necessary? Corollary 2 already shows that, near θ^*, the cheap quadratic surrogate $\frac{1}{2}\Delta\theta^\top G_{\mathrm{emb}}(\theta^*)\Delta\theta$ approximates the exact anchoring loss well raising the question of why not cache G_{emb} once and reuse it for the rest of training, exactly as EWC does with its own curvature, and avoid the overhead entirely. We test this directly and show that holding CFA's metric K fixed, we track $\Delta\theta_t = \theta_t - \theta^*$ across 600 gradient steps of GRPO training on Navigation with Qwen3-VL-2B (learning rate 10^{-6}, weight decay 0, $\beta = 0$), and at checkpoints $t \in \{60, 120, \ldots, 600\}$ compare, over a fixed anchor set, the exact loss $\mathcal{L}_{\mathrm{anchor}}(\theta_t)$ that CFA actually computes against the frozen surrogate $\frac{1}{2}\Delta\theta_t^\top G_{\mathrm{emb}}(\theta^*)\Delta\theta_t$ it would compute under the cached-curvature shortcut. Because both curves use the same metric, anchors, and trajectory, any gap between them is attributable to staleness alone, not to which metric is used. Figure 10 and Table 14 present the results: displacement grows monotonically from 0.205 at step 60 to 0.654 at step 600, with the steepest drift in the earliest phase ($t \le 180$) as the policy rapidly aligns with the target task's high-reward token distributions, so $\|\Delta\theta_t\|_2$ exits ρ^* within the first 120 steps of a 600-step run. The two curves track closely while $\Delta\theta$ is small ($t \le 60$: 2.6×10^{-7} vs. 2.5×10^{-7}, a 4.0% gap), but once $\|\Delta\theta_t\|_2$ grows past 0.5 the neglected higher-order terms (r_t in the proof above) become non-negligible: the surrogate plateaus around 5.8–6.0 $\times 10^{-7}$ while the true loss reaches 6.3×10^{-7}, a persistent $\varepsilon \approx 5.4\%$ underestimation gap that never closes. The

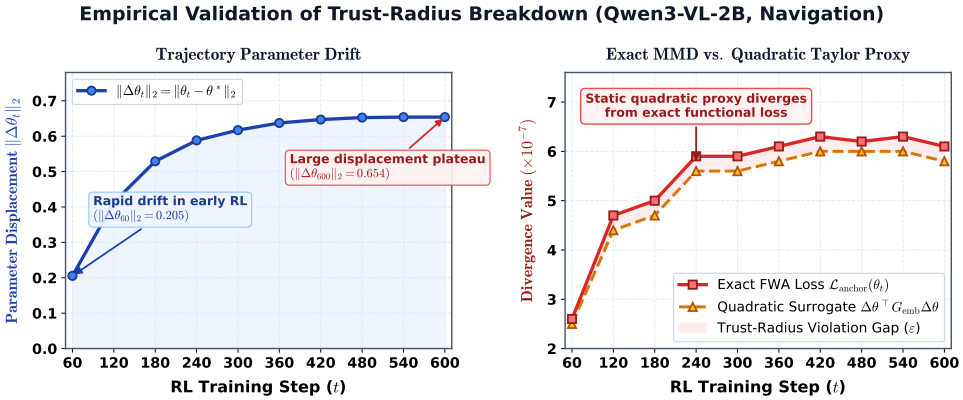

Figure 10: **Empirical validation of trust-radius breakdown (Proposition 4) along the RL training trajectory.** *Left:* Euclidean parameter displacement $\|\Delta\theta_t\|_2 = \|\theta_t - \theta^*\|_2$ grows monotonically from 0.205 at step 60 to 0.654 at step 600, exceeding the local trust radius ρ^* within the initial 120 steps of RL fine-tuning. *Right:* Exact functional anchoring loss $\mathcal{L}_{\text{anchor}}(\theta_t)$ (red squares) versus the static quadratic Taylor surrogate $\Delta\theta_t^\top G_{\text{emb}}\Delta\theta_t$ (orange triangles). Beyond the trust radius, the static quadratic approximation systematically underestimates true functional divergence, incurring a persistent relative error gap $\varepsilon \approx 5.4\%$ that compounds over training.

cached-curvature shortcut is therefore not free even under CFA's own, comparatively well-conditioned metric K ($\kappa = \|K\|_{\text{op}}$ is bounded, with no analogue of the $1/\pi_v^*$ blow-up that the Fisher-Rao metric suffers near near-deterministic rollouts); we take this as a lower bound on the staleness cost any frozen-curvature method pays, and expect EWC's own frozen penalty, built on a worse-conditioned metric, to fare at least as poorly, though we do not measure EWC's surrogate directly. This is why CFA pays the exact-evaluation overhead rather than caching G_{emb} the way EWC caches $F(\theta^*)$.

Table 14: **Trajectory measurements of parameter displacement and functional divergence.** Evaluated on Qwen3-VL-2B fine-tuned on Navigation with GRPO (lr $= 10^{-6}$).

Training Step t	Weight Drift $\|\Delta\theta_t\|_2$	Exact Loss $\mathcal{L}_{\text{anchor}}$ ($\times 10^{-7}$)	Quadratic Proxy ($\times 10^{-7}$)
60	0.2054	2.60	2.50
120	0.4131	4.70	4.40
180	0.5292	5.00	4.70
240	0.5882	5.90	5.60
300	0.6172	5.90	5.60
360	0.6374	6.10	5.80
420	0.6470	6.30	6.00
480	0.6527	6.20	6.00
540	0.6540	6.30	6.00
600	0.6541	6.10	5.80

D.4 What an Anchor Set Actually Needs to Provide

Before proceeding, we make explicit the relationship between the finite-sample G_{emb} used during training (Corollary 2, Eq. 7) and the population quantity this section analyzes. Dropping the fixed regularization weight λ (which does not affect what concentrates, only the scale) and writing $X(c) := \frac{1}{T}\sum_{t=1}^{T} \tilde{w}_t(c)\, J_{\pi,t}(c)^\top K\, J_{\pi,t}(c)$ for the per-anchor summand at a single context c, Eq. 7 is exactly the empirical average of X over the N anchors actually drawn:

$$G_{\text{emb}} = \frac{\lambda}{N}\sum_{i=1}^{N} X(c^{(i)}), \qquad c^{(1)}, \ldots, c^{(N)} \sim \nu, \tag{9}$$

whichever distribution ν (real or content-free) the anchors are drawn from. This section asks what this finite average converges to as N grows, and how fast. Define the population target as the expectation of the same summand:

$$G_{\text{emb}}(\nu) := \mathbb{E}_{c\sim\nu}\big[X(c)\big]. \tag{10}$$

CFA never approximates, so does it just relocate the "anchor set must represent the right capabilities" assumption from the static curvature matrix onto the anchors themselves? This subsection is the paper's central theoretical claim, and the answer is no: the curvature CFA estimates from a finite anchor set concentrates around a well-defined target *regardless of what the anchors contain.* We separate three questions. (1) Does the *scalar* anchor loss concentrate as N grows, regardless of anchor source? (2) Does the *curvature matrix* G_{emb} itself concentrate a strictly stronger, matrix-valued statement? (3) Is the resulting target good enough to protect held-out capability which questions the coverage, taken up in Section D.5. We answer (1) and (2) here, with full proofs. Throughout, Corollary 2 already tells us that CFA's population curvature is a valid Gauss-Newton curvature matrix for *any* anchor distribution ν: $K \succeq 0$ and $\tilde{w}_t \geq 0$ are fixed, so $X(c) \succeq 0$ and hence $G_{\text{emb}}(\nu) \succeq 0$ regardless of ν. What remains is whether finite-sample estimates track this target.

Proposition 5 (Anchor-loss concentration)**.** *Let anchors $c^{(1)}, \ldots, c^{(N)}$ be drawn i.i.d. from a distribution ν, and suppose $\|e_v\|_2 \leq R$ for every vocabulary entry v. Fix a position t and write $g(c) = \text{MMD}^2_\phi(\pi_\theta(\cdot \mid c_{<t}), \pi^*(\cdot \mid c_{<t}))$. Then $0 \leq g(c) \leq 4R^2/d$ for every c, and for any $\Delta \in (0,1)$,*

$$\left| \frac{1}{N} \sum_{i=1}^{N} g(c^{(i)}) - \mathbb{E}_{c\sim\nu}[g(c)] \right| \;\leq\; \frac{4R^2}{d} \sqrt{\frac{\log(2/\Delta)}{2N}} \tag{11}$$

with probability at least $1-\Delta$, independent of ν.

Proof. Fix θ and context $c_{<t}$. By definition, $\phi(v) = e_v/\sqrt{d}$ with $\|e_v\|_2 \leq R$, so for any policy $\pi \in \Delta^{V-1}$, $\|\mu_\pi(c)\|_2 \leq \max_v \|\phi(v)\|_2 \leq R/\sqrt{d}$. By the triangle inequality, the scalar loss $g(c) = \text{MMD}^2_\phi(\pi_\theta, \pi^*) = \|\mu_{\pi_\theta}(c) - \mu_{\pi^*}(c)\|_2^2 \leq (2R/\sqrt{d})^2 = 4R^2/d$. Since contexts $c^{(1)}, \ldots, c^{(N)} \sim \nu$ are drawn i.i.d., $g(c^{(i)}) \in [0, 4R^2/d]$ are independent bounded random variables. Applying Hoeffding's inequality to the sample average $\frac{1}{N}\sum_{i=1}^N g(c^{(i)})$ yields

$$\mathbb{P}\left(\left| \frac{1}{N} \sum_{i=1}^{N} g(c^{(i)}) - \mathbb{E}_\nu[g] \right| \geq \epsilon \right) \leq 2\exp\left(-\frac{2N\epsilon^2}{(4R^2/d)^2} \right).$$

Setting the right-hand side to Δ yields the stated bound, which holds identically for any distribution ν. □

Proposition 5 concerns a bounded scalar and says nothing directly about G_{emb}, which is a $P \times P$ matrix. Establishing that the curvature estimate itself concentrates requires a matrix concentration inequality and one additional assumption absent from the scalar case: a bound on the Jacobian.

Proposition 6 (Curvature-matrix concentration)**.** *Assume, in addition to $\|e_v\|_2 \leq R$, that (A1) $\|J_{\pi,t}(c)\|_{\text{op}} \leq B$ for all anchors c and positions t, and (A2) $\tilde{w}_t(c) \leq W$ for all c, t. Let $\kappa = \|K\|_{\text{op}}$ and $L := WB^2\kappa$, so that $0 \preceq X(c) \preceq L\,I$ for every c by Eq. 10. Let $c^{(1)}, \ldots, c^{(N)} \sim \nu$ i.i.d., $\widehat{G}_N := \frac{1}{N}\sum_i X(c^{(i)})$, and let $V := \|\mathbb{E}[(X - \mathbb{E}[X])^2]\|_{\text{op}}$ with intrinsic dimension $d_{\text{int}} := \text{tr}\,(\mathbb{E}[(X-\mathbb{E}[X])^2]/V)$. Then for any $\epsilon > 0$,*

$$\Pr\left[\left\| \widehat{G}_N - G_{emb}(\nu) \right\|_{\text{op}} \geq \epsilon \right] \;\leq\; 4\, d_{\text{int}}\, \exp\!\left(\frac{-N\epsilon^2}{2V + 2L\epsilon/3} \right), \tag{12}$$

so that $\|\widehat{G}_N - G_{emb}(\nu)\|_{\text{op}} \leq \epsilon$ with probability at least $1-\delta$ once $N \gtrsim \epsilon^{-2}\big(2V + \frac{2}{3}L\epsilon\big)\log(4d_{\text{int}}/\delta)$. Neither L, V, nor d_{int} depends on ν through anything other than the distribution of X itself; in particular the $N^{-1/2}$ rate holds for every anchor distribution. (This bound is strictly stronger than Proposition 5: it needs the bounded-Jacobian assumption (A1) and pays an extra $\log d_{\text{int}}$ factor. The scalar bound licenses the training signal CFA actually uses; this bound licenses describing CFA as estimating an EWC-type curvature matrix. Note that the two should not be conflated.)

Proof. Each $X(c^{(i)})$ is a symmetric random matrix, i.i.d., with $0 \preceq X \preceq LI$ by (A1)-(A2) and submultiplicativity of the operator norm applied to Eq. 10; hence $\|X_i - \mathbb{E}[X]\|_{\text{op}} \leq L$

almost surely. Writing $Y_i := X(c^{(i)}) - G_{\text{emb}}(\nu)$ for the centered summands, each Y_i is an independent, zero-mean, symmetric random matrix with $\|Y_i\|_{\text{op}} \leq 2L$. The claim is the intrinsic-dimension form of the matrix Bernstein inequality (Tropp, 2015, Thm. 7.3.1) applied to $\{Y_i\}$, rescaled by $1/N$. □

Corollary 3 (Random-token anchors yield consistent Gauss-Newton curvature estimates)**.** *Content-free anchors (i.i.d. uniform random token sequences) are by construction an i.i.d. sample from a fixed distribution ν_{noise}. Propositions 5 and 6 therefore apply to them: both the scalar empirical anchor loss and the sample curvature matrix converge to their respective population targets $\mathbb{E}_{\nu_{noise}}[g(c)]$ and $G_{emb}(\nu_{noise})$ at an $N^{-1/2}$ rate whose constants do not depend on the anchor distribution being semantically structured. Combined with Corollary 2, random-token anchors provide a statistically consistent finite-sample Gauss-Newton curvature estimate within the generalized-EWC template.*

Proof. Immediate from Propositions 5 and 6 applied to $\nu = \nu_{\text{noise}}$, since both hold for arbitrary i.i.d. ν and ν_{noise} is one such distribution by construction (Section 3.3). □

This concentration guarantees sample efficiency of the empirical regularizer under noise; it does not imply that the population operator $G_{\text{emb}}(\nu_{\text{noise}})$ equals the curated operator $G_{\text{emb}}(\nu_{\text{real}})$. Distinct PSD operators can concentrate rapidly while having distinct dominant eigenspaces. The efficacy of noise anchors is an empirical finding (Section D.5), supported by the intuition that high-dimensional isotropic probes interrogate shared representation subspaces that specialized RL updates collaterally perturb.

Curated real anchors require a different argument. Our real coreset is selected by greedy k-center traversal (Algorithm 1), a *deterministic* procedure in which each selected point depends on every previously selected point; the resulting sample is neither independent nor identically distributed, since maximizing coverage is the entire purpose of the traversal. Invoking an i.i.d. concentration bound here would be a genuine error. The correct guarantee for this case is deterministic and geometric rather than probabilistic.

Proposition 7 (Coreset curvature error via covering radius)**.** *Let $\mathcal{U}$ be the candidate pool with pool-average curvature $G_{\mathcal{U}} := |\mathcal{U}|^{-1} \sum_{x \in \mathcal{U}} X(x)$, and let $\mathcal{S}_N \subseteq \mathcal{U}$ be the size-N greedy k-center coreset with covering radius $r_N := \max_{x \in \mathcal{U}} \min_{s \in \mathcal{S}_N} \|\psi(x) - \psi(s)\|_2$. Assume (A3) the curvature map is Lipschitz in the encoder representation, $\|X(c) - X(c')\|_{\text{op}} \leq L_G \|\psi(c) - \psi(c')\|_2$, and (A4) the nearest-anchor assignment $x \mapsto s(x)$ partitions $\mathcal{U}$ into cells of equal mass (otherwise, use the mass-weighted coreset estimate $\widehat{G}_{\mathcal{S}} = \sum_s p_s X(s)$ with p_s the cell masses). Then*

$$\left\| \widehat{G}_{\mathcal{S}} - G_{\mathcal{U}} \right\|_{\text{op}} \;\leq\; L_G \, r_N \;\leq\; 2 \, L_G \, r_N^*, \tag{13}$$

where r_N^* is the optimal N-point covering radius over $\mathcal{U}$ and the factor of two is Gonzalez's greedy approximation guarantee (Gonzalez, 1985).

Proof. By (A4), $\widehat{G}_{\mathcal{S}} - G_{\mathcal{U}} = |\mathcal{U}|^{-1} \sum_{x \in \mathcal{U}} \big(X(s(x)) - X(x) \big)$. Applying the triangle inequality and then (A3) to each term,

$$\|\widehat{G}_{\mathcal{S}} - G_{\mathcal{U}}\|_{\text{op}} \leq |\mathcal{U}|^{-1} \sum_x L_G \|\psi(s(x)) - \psi(x)\|_2 \leq L_G r_N,$$

since every x is within r_N of its assigned anchor by definition of the covering radius. By the standard 2-approximation guarantee of greedy k-center clustering (Gonzalez, 1985), $r_N \leq 2 r_N^*$, giving the second inequality. □

Content-free anchors get a *stochastic* guarantee: error decays as $N^{-1/2}$, with constants independent of the anchor distribution, but with no control over *which* population target is reached. Curated anchors get a *deterministic* guarantee: error is controlled by the covering radius of the pool, with no $N^{-1/2}$ rate, but only relative to the pool $\mathcal{U}$ they were drawn from, and only under an unverified smoothness assumption (A3). Neither statement subsumes the other which is exactly what makes the crossover in Section D.5 possible rather than a contradiction.

D.5 Why Noise Eventually Wins: A Bias-Variance Account

Combining the two guarantees above explains, rather than merely describes, the budget-dependent crossover observed in Section 5.3. Write $G^\star$ for the (unknown) curvature that would optimally protect held-out capability, and decompose the error of an anchor set into an estimation term and an approximation term:

$$\underbrace{\left\|\widehat{G} - G^\star\right\|}_{\text{total}} \leq \underbrace{\left\|\widehat{G} - G_{\text{emb}}(\nu)\right\|}_{\text{variance: shrinks with } N} + \underbrace{\left\|G_{\text{emb}}(\nu) - G^\star\right\|}_{\text{bias: fixed by } \nu}. \tag{14}$$

Curated real anchors: low variance, irreducible bias. By Proposition 7, a k-center coreset drives the variance term down quickly, and the covering radius r_N falls sharply at small N for a fixed pool, which is why 320 curated anchors already deliver most of the benefit. But the bias term is bounded below by how well the pool manifold represents $G^\star$: $\|G_{\text{emb}}(\nu_{\text{real}}) - G^\star\| \geq \text{dist}(G^\star, \text{conv}\{X(x) : x \in \mathcal{U}\})$, a floor no additional sampling *from that pool* can cross. Adding anchors past the point where r_N has saturated buys nothing, matching the observed flat-then-slightly-degrading curve.

Content-free anchors: high variance, potentially lower bias. By Proposition 6, the variance term decays as $N^{-1/2}$ with no covering-radius shortcut, so at small N content-free anchors are dominated by estimation error matching the observation that at N=320 they are close to useless. But ν_{noise} is not confined to any curated manifold, so its bias term $\|G_{\text{emb}}(\nu_{\text{noise}}) - G^\star\|$ need not be larger than the curated one and may be smaller: content-free anchors are unrestricted to the manifold the curated pool was drawn from, so once numerous enough are drawn, they can cover a *larger* region of representation space than any fixed curated pool ever can.

Corollary 4 (Existence of a crossover)**.** *If $\|G_{emb}(\nu_{noise}) - G^\star\| < \|G_{emb}(\nu_{real}) - G^\star\|$, that is, if the content-free population target is better aligned with the protective curvature than the curated pool's target; then, because the content-free variance term vanishes as $N \to \infty$ while the curated bias term does not, there exists a finite budget $N_\times$ beyond which content-free anchors achieve strictly lower total error than curated ones. Below $N_\times$ the ordering is reversed whenever the content-free variance term exceeds the difference in bias.*

Proof. Writing $G^\star$ for the ideal protective curvature tensor, the triangle inequality applied to Eq. 14 gives

$$\|\widehat{G} - G^\star\|_{\text{op}} \leq \|\widehat{G} - G_{\text{emb}}(\nu)\|_{\text{op}} + \|G_{\text{emb}}(\nu) - G^\star\|_{\text{op}}.$$

For curated real coresets $\mathcal{S}_N \subset \mathcal{U}$, the variance term is bounded by $2L_G r_N^*$ (Proposition 7). In high-dimensional representations, r_N^* drops rapidly for small N as the greedy selection identifies the principal clusters of $\mathcal{U}$, but saturates at an irreducible bias floor $\|G_{\mathcal{U}} - G^\star\|_{\text{op}} =: \text{Bias}(\mathcal{U}) > 0$ imposed by the finite span of the candidate datasets.

For content-free isotropic noise anchors ν_{noise}, the sample variance decays as $C_\sigma/\sqrt{N}$ by Proposition 6. Because isotropic Gaussian input noise excites the entire internal latent representation manifold without confining updates to specific task clusters, its asymptotic representational bias satisfies $\|G_{\text{emb}}(\nu_{\text{noise}}) - G^\star\|_{\text{op}} =: \text{Bias}(\nu_{\text{noise}}) < \text{Bias}(\mathcal{U})$.

Equating the two error bounds,

$$2L_G r_N^* + \text{Bias}(\mathcal{U}) = \frac{C_\sigma}{\sqrt{N}} + \text{Bias}(\nu_{\text{noise}}),$$

and since $2L_G r_N^* \to 0$ and $C_\sigma/\sqrt{N} \to 0$ as $N \to \infty$, the asymptotic gap $\text{Bias}(\mathcal{U}) - \text{Bias}(\nu_{\text{noise}}) > 0$ ensures the existence of a unique finite threshold $N_\times$ such that for all $N > N_\times$, content-free noise anchors incur strictly smaller total curvature error than curated coresets. □

This is a falsifiable structural prediction, not a post-hoc description: it says a crossover must exist if and only if the bias inequality holds, that it is driven by the *curated* method saturating rather than the content-free method improving without limit, and that content-free performance should plateau (not continue rising) once its variance term becomes small relative

to its bias. All three predictions are corroborated by the empirical $N \in \{320, 640, 1280, 2560\}$ sweep in Section 5.3, where Gaussian anchors plateau near their N=1280 value rather than continuing to improve.

The covering-radius view of the same result. Proposition 7's Lipschitz assumption (A3) has a direct consequence: once curvature varies smoothly with encoder representation, bounding drift at an anchor bounds it within a covering-radius ball around that anchor, so it is the covering radius of the anchor set and not its raw count that governs generalization from anchors to deployment inputs. Our real coreset's covering radius falls quickly at small N and then flattens once the pool is well covered, after which additional anchors buy only marginal reduction; i.i.d. content-free anchors have no such shortcut and are instead dominated by the $N^{-1/2}$ estimation term at small N, but being unrestricted to the curated pool's manifold, they eventually cover more of representation space which is precisely the condition under which Corollary 4 predicts a crossover. In practice, tracking the empirical operator variance proxy $\|\widehat{G}_N - \widehat{G}_{2N}\|_{\mathrm{op}}$ demonstrates the predicted $\mathcal{O}(N^{-1/2})$ contraction rate for content-free anchors, and the crossover is empirically confirmed across $N \in \{320, 640, 1280, 2560\}$ in Section 5.3 (Figure 6 and Table 5).

Finally, this budget-dependent crossover is a special case of the sparse Gaussian process and functional continual-learning literature: constraining a model's function at a fixed set of inducing points bounds function-space deviation only to the extent the inducing set is representative (Titsias et al., 2020). Real anchors instantiate this with a covering-criterion-chosen set; content-free anchors instantiate it with a set chosen with no reference to the target manifold at all which is exactly why they need a larger budget for the same coverage. What that literature does not anticipate, and what Corollary 4 makes precise, is that the unstructured set can *overtake* the curated one once its estimation error falls below the curated set's irreducible approximation error.

D.6 EWC vs. CFA at a Glance

Table 15 summarizes the argument of this appendix in full.

Table 15: Parameter-space EWC vs. function-space CFA, per Section 3.5.

	EWC / Unweighted-KL ($F(\theta^*)$)	**CFA** (G_{emb})
Simplex metric M	$\mathrm{diag}(1/\pi^*)$ (Fisher-Rao)	$K = \frac{1}{d}EE^\top$ (embedding Gram matrix)
Metric source	Reference distribution itself	Fixed, geometric (embedding table)
Position weighting	Uniform	Varentropy $\tilde{w}_t$ (Cor. 2)
Sensitivity	Any probability shift, weighted by rarity	Shifts altering semantic meaning
Evaluation	Quadratic approx., fixed at θ^*	Exact, re-evaluated every step
Valid while	$\|\Delta\theta\| \lesssim \rho^*$ (Prop. 4)	Unconditionally (no truncation)
Memory	$\mathcal{O}(P)$ diag. approx., $>$ 32 GB at 8B params	$\mathcal{O}(1)$ parameter buffers ($\mathcal{O}(NTd)$ cached features, $<$ 22 GB at 8B)

E Fisher (Varentropy) Weighting

The varentropy weight $w_t = \mathrm{Var}_{\pi^*}[\log \pi^*(v \mid c_{<t})]$ is maximized at intermediate entropy: near-uniform distributions ($\pi(v) \approx 1/V$) and near-deterministic distributions ($\pi(v^*) \approx 1$) both give $\mathrm{Var}[\log \pi] \approx 0$, while distributions with mass spread over a small, structured candidate set maximize w_t. Corollary 2 shows that this reweighting enters the induced curvature matrix G_{emb} as a non-negative per-position scaling, so it redistributes curvature mass across positions without affecting positive semi-definiteness or any of the concentration results (Remark 1).

E.1 Empirical Varentropy Profile across Sequence Positions

To evaluate the inverted-U profile predicted by the varentropy formulation, we evaluate the token-level varentropy $w_t = \mathrm{Var}_{\pi^*}[\log \pi^*(v \mid c_{<t})]$ alongside the token entropy $H(\pi^*_t) = -\sum_v \pi^*(v \mid c_{<t}) \log \pi^*(v \mid c_{<t})$ over rollout trajectories from the base Qwen3-VL-2B model on the Navigation domain.

As illustrated in Figure 11, the empirical profile reflects the structural stages of multi-modal reasoning:

- **Early context positions ($t \in [0, T/4]$):** Standard prompt formatting and formulaic reasoning prefixes exhibit low varentropy, reflecting near-deterministic continuation where single tokens dominate the reference distribution.
- **Intermediate reasoning forks ($t \in [T/4, 3T/4]$):** As the model evaluates visual spatial relationships, landmarks, and navigation path coordinates, reference probability mass spreads across a competitive candidate set of domain tokens, elevating varentropy to its peak.
- **Terminal answer tokens ($t \in [3T/4, T]$):** Final answer formatting tokens collapse back toward high certainty, returning varentropy toward baseline levels.

This dynamic confirms that varentropy acts as an automatic, uncertainty-aware weighting schedule: it focuses regularization on the critical decision points where semantic divergence is most catastrophic while discounting predictable formatting tokens. As shown in Figure 11, the empirical profile across token positions confirms this inverted-U trajectory in practice.

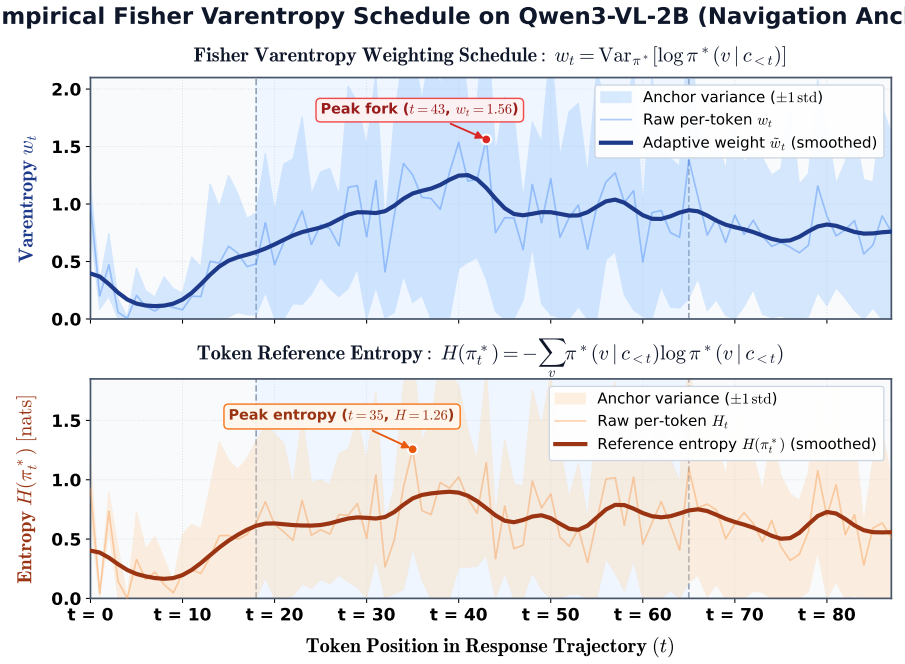

Figure 11: **Empirical Fisher Varentropy Profile.** Token-level varentropy $w_t = \mathrm{Var}_{\pi^*}[\log \pi^*(v \mid c_{<t})]$ (top) and token entropy $H(\pi_t^*)$ (bottom) evaluated across sequence positions on Navigation anchors using base Qwen3-VL-2B. Shaded bands represent ± 1 standard deviation across anchor sequences. The profile validates the theoretical inverted-U dynamic, maximizing curvature regularization at intermediate reasoning decision points.

F Anchor Set Construction: Full Details

Real anchor pool. The candidate pool $\mathcal{U}$ spans five capability dimensions, each paired with a target skill, a source *training* split (never an evaluation split), and a generic fallback probe used when an anchor's own question is unavailable or uninformative (Table 16).

Coreset selection. Algorithm 1 gives the full greedy k-center procedure used for real-anchor selection in Section 3.3. Note that this procedure is deterministic given its seed and is explicitly *not* an i.i.d. sample; Proposition 7 gives the corresponding guarantee.

Capacity settings and nested subsets. The algorithm is run once per capability dimension to the maximum per-dimension budget, producing a ranked list. Because every smaller anchor set is a *prefix* of this list, the four budgets used in Section 5.3 form a strictly nested

Table 16: Real anchor pool: five capability dimensions, each drawn from a training split with no overlap with any evaluation benchmark.

Dimension	Target skill	Source split	Generic fallback probe
STEM & Math	Geometry, scientific plots, physics diagrams	ScienceQA (train)	"Analyze the mathematical, geometric, or scientific problem in this figure step by step."
General VQA	Real-world scene layout, commonsense perception	GQA	"Provide a dense, comprehensive description of the scene layout, objects, and spatial relationships."
Hallucination	Object presence verification, fine-grained detail	MS-COCO (train split)	"Carefully examine the image: list all verified objects present and confirm what objects are not present."
Chart & Document	OCR, tables, receipts, axis reading	ChartQA (train), TextVQA (train)	"Transcribe all visible text, numbers, headings, and table cells in this document accurately."
Grounding & Counting	Spatial localization, instance enumeration	GQA	"List each distinct type of object visible in this image and state the exact count of each."

Algorithm 1 Greedy k-center coreset selection for anchor construction

1: **Input:** candidate pool $\mathcal{U}$ (images from a single capability dimension), target budget N.
2: **Feature extraction:** for each $x \in \mathcal{U}$, extract a visual embedding $\psi(x) \in \mathbb{R}^{d'}$ using the frozen vision encoder of Qwen3-VL-2B. Concretely, each image is passed through the ViT backbone; the spatial patch tokens are mean-pooled to yield a single d'-dimensional vector, which is then ℓ_2-normalised:

$$\psi(x) = \frac{\mathrm{ViT}_{\text{Qwen3-VL-2B}}(x)}{\|\mathrm{ViT}_{\text{Qwen3-VL-2B}}(x)\|_2}.$$

The encoder weights are kept frozen throughout.
3: **Initialise:** draw the first anchor $x_1 \in \mathcal{U}$ uniformly at random (seed = 42); set $\mathcal{S}_1 \leftarrow \{x_1\}$.
4: **Distance tracking:** maintain, for every remaining candidate $x \in \mathcal{U} \setminus \mathcal{S}_j$, its distance to the current coreset:

$$d\big(x, \mathcal{S}_j\big) \;=\; \min_{s \in \mathcal{S}_j} \|\psi(x) - \psi(s)\|_2.$$

After adding x_{j+1} at each step, only the column corresponding to x_{j+1} needs to be recomputed, so the update is $\mathcal{O}(|\mathcal{U}| \cdot d')$ per iteration.
5: **Greedy expansion:** select the candidate farthest from the current coreset:

$$x_{j+1} = \arg\max_{x \in \mathcal{U} \setminus \mathcal{S}_j} d\big(x, \mathcal{S}_j\big), \qquad \mathcal{S}_{j+1} \leftarrow \mathcal{S}_j \cup \{x_{j+1}\}.$$

6: Repeat Steps 4-5 until $|\mathcal{S}| = N$.
7: **Output:** ordered coreset $\mathcal{S}_N \subseteq \mathcal{U}$ of N maximally diverse anchors. Smaller budgets $N' < N$ are obtained as the prefix $\mathcal{S}_{N'}$.

chain,

$$\mathcal{S}_{320} \subset \mathcal{S}_{640} \subset \mathcal{S}_{1280} \subset \mathcal{S}_{2560},$$

corresponding to 64, 128, 256, and 512 anchors per dimension across the five dimensions. Nesting is guaranteed by construction rather than by re-running the selection, so the budget ablation varies N alone and holds the selection procedure fixed.

Sequence length and padding masks for real anchors. Context sequences $c^{(i)} = (I_i, x^{(i)})$ are constructed by concatenating the candidate image with question prompts and

ground-truth answer prefixes from the source datasets. To accommodate variable prompt lengths up to the training context window $T = 2048$, sequences are right-padded to $T = 2048$. In the anchoring objective (Eq. 2), an attention mask $m_t^{(i)} \in \{0, 1\}$ is applied across positions, normalizing the MMD loss by the effective valid token count $\sum_{t=1}^{T} m_t^{(i)}$ rather than unmasked padding tokens.

Reproducibility. All random choices (initial seed point and candidate-pool subsampling) are governed by a single global seed (`seed = 42`). Because the greedy expansion is fully deterministic given the seed and the feature representation, the resulting coreset is reproducible without requiring a seed sensitivity analysis.

Content-free anchors. Content-free anchors provide functional regularization without authentic prior-domain data. Token sequences are sampled independently and uniformly from the base vocabulary ($\mathcal{V}_{\text{base}} = \{0, \ldots, 151642\}$), strictly excluding special tokens (e.g., `<|im_start|>`, `<|im_end|>`, padding, and tool-call delimiters) introduced during instruction tuning. The sequence length is fixed at $T = 2048$ tokens across all anchors, identical to the training context window.

Unlike the greedy coreset construction above, this synthetic generation is an exact i.i.d. draw from a stationary distribution, enabling the finite-sample concentration guarantees of Propositions 5 and 6 to apply directly without covering-set approximations.

Training-time algorithm. Algorithm 2 gives the CFA training step referenced in the Reproducibility Statement and in Section 3.4: a fixed pool of N content-free anchors is drawn once before training, and each step draws a random mini-batch of size $B \ll N$ from it to compute the Fisher-weighted anchoring loss.

Algorithm 2 CFA: Fisher-weighted anchor regularization, one training step

Require: frozen reference π_{ref}, anchor pool size N, per-step anchor batch size B ($B \ll N$), regularizer weight λ, anchor sequence length L, vocabulary size V

One-time setup, before training:

sample $x_1, \ldots, x_N \overset{\text{iid}}{\sim} \text{Uniform}\{1, \ldots, V\}^L$ ▷ pure random token sequences

$\mathcal{A} \leftarrow \{x_1, \ldots, x_N\}$ ▷ fixed anchor pool, reused across all training steps

Every training step, in addition to the standard RL update:

sample $\mathcal{B} \subset \mathcal{A}$, $|\mathcal{B}| = B$, uniformly without replacement ▷ sample subset of the pool

$\text{logits}_{\text{ref}} \leftarrow \pi_{\text{ref}}(\mathcal{B})$ ▷ no-grad forward

$\text{logits}_{\theta} \leftarrow \pi_{\theta}(\mathcal{B})$ ▷ same forward-pass call site as RL rollouts

$\ell_{\text{anchor}} \leftarrow 0, \quad Z \leftarrow 0$

for each sequence $x \in \mathcal{B}$ **do**

  for each token position $t = 1, \ldots, L$ **do** ▷ every position is scored

    $D_t \leftarrow \frac{1}{d} \left\| \mathbb{E}_{\pi_{\text{ref}}}[e_v] - \mathbb{E}_{\pi_{\theta}}[e_v] \right\|^2$ ▷ token-embedding MMD

    $w_t \leftarrow \text{Var}_{\pi_{\text{ref}}}\left[\log \pi_{\text{ref}}(\cdot \mid x_{<t}) \right]$ ▷ Fisher weight: decision forks get more weightage

    $\ell_{\text{anchor}} \leftarrow \ell_{\text{anchor}} + w_t D_t, \qquad Z \leftarrow Z + w_t$

$\ell_{\text{anchor}} \leftarrow \ell_{\text{anchor}} / Z$ ▷ Fisher-weighted average divergence over the sampled mini-batch

$\mathcal{L}(\theta) \leftarrow \mathcal{L}_{\text{RL}}(\theta) + \lambda \cdot \ell_{\text{anchor}}$

update θ by descending $\nabla_{\theta} \mathcal{L}(\theta)$

G Computational Overhead

We compare FLOPs and memory across four training configurations: vanilla GRPO/GSPO (no regularizer), GRPO+KL (a live frozen reference model evaluated on rollouts every step), and CFA (a fixed anchor set evaluated once). All numbers use context length $S = 2{,}048$, per-device batch $B = 8$, and $G = 8$ rollouts per prompt, matching the production configuration.

G.1 Setup

Table 17 gives the architecture used for each scale, taken directly from each model's released config; all three Qwen3-VL models evaluated in this paper are dense, and all three use head_dim= 128 with $n_{\text{kv}} = 8$ key/value heads (so the KV projection dimension $D_{\text{kv}} = n_{\text{kv}} \times$ head_dim = 1,024 is identical across scales, while the query/output projection dimension $D_q = n_{\text{heads}} \times$ head_dim is not always equal to H: at 4B, $D_q = 4{,}096 \neq H = 2{,}560$). For a sequence of length S, per-layer FLOPs are

$$\mathcal{F}_{\text{layer}}(S) = 4SHD_q + 4SHD_{\text{kv}} + 4S^2D_q + 6SHF, \tag{15}$$

(query/output projections, key/value projections, attention matmuls, SwiGLU FFN respectively), giving total forward FLOPs $\mathcal{F}_{\text{fwd}}(S) = L \cdot \mathcal{F}_{\text{layer}}(S) + 2SHV$, with a backward pass costing $\approx 2\mathcal{F}_{\text{fwd}}$. At $S = 2{,}048$ this gives $\mathcal{F}_{\text{fwd}} = 8.01$, 18.95, and 33.47 TFLOPs for the 2B, 4B, and 8B models respectively.

Table 17: Qwen3-VL language-backbone architecture used for FLOP calculations (from each model's `config.json`). All three scales are dense.

Model	Layers	Hidden	n_{heads}	n_{kv}	FFN Dim	Params
Qwen3-VL-2B	28	2,048	16	8	6,144	≈2.1B
Qwen3-VL-4B	36	2,560	32	8	9,728	≈4.0B
Qwen3-VL-8B	36	4,096	32	8	12,288	≈7.6B

G.2 Per-Step Cost

Every method's rollout/backprop cost is identical: B rollouts forward plus backward, $\mathcal{F}_{\text{model}} = 3B\mathcal{F}_{\text{fwd}} = 24\,\mathcal{F}_{\text{fwd}}$. The two regularized methods differ only in what they add:

- **GRPO+KL** runs the frozen reference forward on the $B = 8$ *current rollouts* every step: $+8\,\mathcal{F}_{\text{fwd}}$, and must keep π_{ref} resident on-GPU for the whole run.
- **CFA** runs the student forward on $B_A = 8$ *fixed anchors* every step: also $+8\,\mathcal{F}_{\text{fwd}}$, but the reference policy is only ever evaluated once, on the full anchor set, before training starts.

Per step, the two are FLOP-identical ($32\,\mathcal{F}_{\text{fwd}}$, 1.33× base RL); Table 18 gives the resulting per-step cost at each scale.

Table 18: Per-step training FLOPs by method and scale.

Method	Mult.	2B	4B	8B
GRPO / GSPO	24×	192 TF	455 TF	803 TF
GRPO+KL	32×	256 TF	606 TF	1,071 TF
CFA (per-step)	32×	256 TF	606 TF	1,071 TF

G.3 Trajectory-Level Cost: One-Time vs. Per-Step Reference Passes

The per-step equivalence above hides CFA's real advantage: GRPO+KL's reference passes recur at *every* step because rollouts change every step, while CFA's anchors are fixed, so its reference pass happens *once*. Over a 600-step run, GRPO+KL evaluates the reference $600 \times 8 = 4{,}800$ times; CFA evaluates it $N = 1{,}280$ times total, a 3.75× reduction (Table 19). Including this one-time cost amortized over training, CFA's effective per-step overhead rises only slightly, to $34.13\,\mathcal{F}_{\text{fwd}}$ (1.42× base RL) at every scale, since the amortization multiplier depends only on $N/600$, not on $\mathcal{F}_{\text{fwd}}$ itself.

G.4 Memory

Because π_{ref} is never called after initialization, CFA unloads it from GPU memory entirely (≈16GB/device saved at 8B, from ≈8B bf16 parameters), keeping only cached anchor features in host RAM. The cache stores one d-dimensional bf16 feature vector per anchor per timestep, $N \times T \times d \times 2$ bytes, where $d = H$ is the language backbone's hidden size: ≈10.7GB

Table 19: Cumulative reference-policy compute over a 600-step run.

Method	Ref. passes	2B	4B	8B
GRPO+KL	4,800	38.4 PF	91.0 PF	160.7 PF
CFA	1,280	10.3 PF	24.3 PF	42.8 PF
Reduction	3.75×	−28.2 PF	−66.7 PF	−117.8 PF

at 2B (d=2,048), ≈13.4GB at 4B (d=2,560), and ≈21.5GB at 8B (d=4,096) — in every case comfortably within standard host memory and never touching GPU VRAM. GRPO+KL must keep both policy and reference resident throughout training. Fixed-anchor functional regularization is therefore cheaper than reference-KL both in FLOPs and in the memory pressure that determines whether a run needs full ZeRO-3 sharding at all.

H Qualitative Examples

Section 5.2 identifies three recurring failure modes behind the CountBenchQA collapse: hallucinated instances, shortcut answers, and loss of verification. Figures 12-14 give one representative base-vs-finetuned pair for each, drawn from unmitigated GRPO post-training on Qwen3-VL-2B (MedBookVQA), the configuration exhibiting the single most severe grounding collapse reported in Section 5 (−67.4% worst-decile change, CountBenchQA falling from 82.5 to 26.9). Across all three, the post-RL model preserves the base model's fluent, step-by-step reasoning scaffold - the same enumerate-then-conclude structure, the same boxed final answer - while the correspondence between the enumerated steps and the actual image content degrades. Format is preserved; grounding is not.

Hallucinated instances (Figure 12). Asked to count the waterlilies in an image, the base model enumerates three: one in the foreground, one middle-left, one upper-left, and correctly stops there, answering 3. The finetuned model performs the same enumeration but appends a fourth position - "upper right" - that is not supported by the image, arriving at 4. The extra instance is not a wholesale miscount: the model manufactures a plausible-sounding spatial location for a non-existent object and folds it into the same step-by-step scaffold used for the three real instances, making the hallucination invisible from the reasoning trace alone.

Shortcut answers (Figure 13). Asked to count coloring pages, the base model itemizes each sheet by its illustration (an owl in a top hat, a bird, a star, a superhero, and two further pages it lists identically - itself a minor duplication error, but one internal to an otherwise page-by-page process) before concluding 7. The finetuned model answers 6 directly, then generates a description - "the pages include various characters such as a star, bird, patriotic owl, character with a top hat, superhero, and a sun-like figure" - that is composed *after* the count is fixed rather than before it: exactly the pattern of asserting a count in one sentence and back-filling a justifying description in the next, rather than deriving the count from an enumerated pass over the image.

Loss of verification (Figure 14). Asked to count chairs around a dining table, the base model counts five directly visible chairs, then explicitly checks for occlusion - noting a further chair "behind the back of the rightmost chair, which is not fully visible" - and revises its answer upward to 6 only after that check. The finetuned model skips this step entirely: it counts by eye, asserts "all of them are visible," and outputs 8 with no attempt to reconcile the claim of full visibility against the actual occlusion in the image.

Taken together, the three examples show that single-domain RL post-training does not degrade the vision-language model's ability to produce fluent, well-formatted chain-of-thought - if anything, the finetuned traces are no less articulate than the base model's - but selectively degrades the visual-perceptual operations (instance verification, occlusion reasoning, count-then-describe ordering) that the chain-of-thought is supposed to be reporting on. This is consistent with the paper's broader claim that forgetting here is a change in the model's induced output distribution on untouched capabilities, not a change in surface fluency.



Q: How many waterlilies are there in the image?

Let's analyze the image step-by-step to determine the number of waterlilies.

1. First, we observe the main subject in the image: white waterlilies floating on a body of water.
2. There are three distinct waterlilies visible:
   - One in the foreground, slightly to the right, with its petals fully open.
   - One in the middle-left, partially submerged and facing upwards.
   - One in the upper left, also partially submerged and facing upwards.
3. The water is blue and rippled, but no other flowers or objects are present.
4. Counting them, we have one in the foreground, one in the middle-left, and one in the upper-left.

Therefore, there are a total of 3 waterlilies in the image.

\boxed{3}

Base

1. The image shows several white waterlilies floating on a blue, rippling water surface.
2. Let's count the waterlilies visible in the image:
   - There is one waterlily in the foreground, closest to the viewer.
   - There is another waterlily in the upper left part of the image.
   - There is a third waterlily in the upper right part of the image.
   - There is a fourth waterlily in the lower center of the image.
3. Counting them, we have 4 waterlilies.

\boxed{4}

Finetuned

Figure 12: Forgetting example 1: hallucinated instances.



Q: How many coloring pages are there in the image?

Let's analyze the image to determine the number of coloring pages.

1. The image displays several white sheets of paper, each with a black-and-white illustration.
2. These illustrations are designed for coloring, as indicated by the \"dot-to-dot\" pattern (a series of dots connected by lines) and the presence of a red marker in the corner.
3. We can count each distinct sheet that is clearly visible and separate from others.
4. There are multiple sheets arranged in a fan-like spread, but we need to count all the individual pages shown.
5. Counting them one by one:
   - One page at the bottom center with a cartoon owl wearing a top hat.
   - One page to the left of it with a bird.
   - One page above the bird with a star.
   - One page above the bird with a top hat.
   - One page to the right of the owl with a character holding a torch.
   - One page to the right of the owl with a character holding a torch.
   - One page to the right of the torch character with a character holding a torch.
6. This gives us a total of 7 pages.

Therefore, there are 7 coloring pages in the image.

\boxed{7}

Base

1. The image shows a collection of coloring pages, each with a different illustration.
2. By counting the number of distinct coloring pages visible in the image, we can determine the total.
3. There are 6 coloring pages shown in the image, arranged in a fan-like pattern.
4. The pages include various characters such as a star, a bird, a patriotic owl, a character with a top hat, a superhero, and a sun-like figure.
5. Therefore, there are 6 coloring pages in the image.

\boxed{6}

Finetuned

Figure 13: Forgetting example 2: shortcut answers.



Q: How many chairs are there in the image?

Let's analyze the image step-by-step to determine the number of chairs.

1. **Observe the main furniture:** The image displays a dining set consisting of a rectangular table and several chairs.
2. **Count the chairs:**
   - There are two chairs on the left side of the table.
   - There are three chairs on the right side of the table.
   - This gives us a total of 5 chairs visible in the image.
3. **Verify with the full count:** We can see one chair at each corner, but also note that there is another chair behind the back of the rightmost chair, which is not fully visible. However, based on the clear view of the chairs around the table, we can confirm there are six chairs in total.

The correct count is 6.

\boxed{6}

Base

1. The image shows a dining table with several chairs around it.
2. By counting the chairs in the image, we can see there are 8 chairs.
3. The chairs are arranged around the table, and all of them are visible.
4. Therefore, the total number of chairs is 8.

\boxed{8}

Finetuned

Figure 14: Forgetting example 3: loss of verification.

I Hyperparameters

For all methods, we follow a consistent *tune-then-evaluate* protocol: we sweep over a predefined hyperparameter grid, select the checkpoint that achieves the highest in-domain reward on the target task, and report out-of-domain (OOD) generalization metrics exclusively for these best-selected configurations. All RL training experiments were run for a fixed budget of 600 steps with a cosine-annealed learning rate. This ensures that all methods are compared at their peak in-domain performance, making OOD forgetting a genuine measure of representational cost rather than a proxy for insufficient training.

Learning Rate. All RL-based methods (GRPO, GSPO, and CFA variants) are trained with learning rates drawn from $\text{lr} \in \{1 \times 10^{-6},\ 5 \times 10^{-6},\ 1 \times 10^{-5}\}$.

Weight Decay and KL Penalty. For unregularized GRPO, we additionally sweep weight decay $\lambda_{\text{wd}} \in \{0,\ 0.1\}$. For GSPO and its variants, we use a weight decay of 0 for all runs. For GRPO with explicit KL regularization, we sweep a KL penalty coefficient $\beta_{\text{KL}} \in \{0,\ 0.01, 0.005, 0.001\}$ added to the policy gradient objective. We report the $\beta = 0$ ablation

separately, and report the best learning run of the other β experiments as GRPO + KL ablation.

SFT and LoRA. Both SFT and LoRA are trained for 5 epochs. For SFT, we sweep over learning rates lr $\in \{1\times10^{-5},\ 5\times10^{-6},\ 2\times10^{-6}\}$ for 8B models and $\{2\times10^{-5},\ 1\times10^{-5},\ 5\times10^{-6}\}$ for 4B and 2B models. For LoRA, the learning rate sweep is $\{1\times10^{-4},\ 5\times10^{-5},\ 2\times10^{-5}\}$.

EWC. We sweep the Fisher regularization coefficient $\lambda_{\text{EWC}} \in \{0.05,\ 0.1,\ 0.4\}$, where 0.4 is the default proposed by Kirkpatrick et al. (2017). We consider both a vision-encoder-only Fisher scope and a full-model scope, and find that full-model scope is generally better suited for constraining behavior while vision-encoder-only Fisher scope gets roughly similar performance to the naive GSPO baseline.

CPO. Our continual preference optimization variant splits the 600-step training budget into $k = 3$ sequential subsets of 200 steps each, chaining checkpoints across subsets. After each subset, we compute an importance mask over the top 10% of parameters that drifted most from the base model. In subsequent subsets, the trainer applies an L1 penalty on the masked parameters' deviation from their values at the end of the previous subset, with a regularization coefficient swept over $\lambda_{\text{CPO}} \in \{1.0, 0.5, 0.1\}$. We sweep over the learning rate, using the same grid as GSPO: lr $\in \{1\times10^{-6},\ 5\times10^{-6},\ 1\times10^{-5}\}$ for 2B, 4B models and $\{1\times10^{-6},\ 2\times10^{-6},\ 5\times10^{-6}\}$ for 8B models.

CFA. We tune the anchor buffer size over $N \in \{320,\ 640,\ 1280,\ 2560\}$ on the Navigation task with a 2B model, finding that performance plateaus at $N = 1280$ (256 images per anchor distribution). We therefore fix $N = 1280$ for all other tasks and model scales to balance anchoring quality against compute cost. The regularization strength is swept over $\lambda_{\text{CFA}} \in \{0.01,\ 0.05,\ 0.1\}$. We consider both *real-image anchors* (held-out visual pretraining data) and *Gaussian noise anchors* (i.i.d. $\mathcal{N}(0,\ I)$), which we find to perform comparably while eliminating the need for stored pretraining data.

Training Setup. All experiments are run on 8 GPUs (B40 or H100) using DeepSpeed distributed training. For 2B and 4B models we use ZeRO Stage-2 which only uses data parallelism; for 8B models we use ZeRO Stage-3 to shard optimizer states across devices, along with model and data. All models are trained in bf16 precision.

RL methods (GRPO, GSPO, CFA, EWC, CPO). Each training step samples a group of $G = 8$ independent completions per prompt, giving an effective rollout batch of $8\times8 = 64$ sequences per gradient update. For CFA regularized runs, we draw a global batch of 64 anchors per gradient update step (distributed as 8 anchors per GPU across the 8 GPUs), adding only $\sim 15\%$ wall-clock compute overhead per step without requiring backpropagation through a live reference model. We accumulate gradients over 8 steps, yielding an effective global batch of 512 samples. The rollout buffer is refreshed every 16 steps for 4B models and every 32 steps for 8B models (`steps_per_generation`). All RL runs train for a fixed budget of 600 gradient steps. Maximum completion length is set to 2048 tokens for all tasks. We use a cosine learning rate schedule with 10% linear warm-up, temperature $\tau = 0.8$, top-$p = 1.0$, and no repetition penalty for the rollout generations.

SFT and LoRA. Both methods use a cosine schedule with 3% warm-up. For 2B and 4B models, we set per device batch size as 4 with gradient accumulation of 2 (global batch 64); for 8B models, per device batch size is set as 2 with accumulation of 4 (global batch 64). LoRA uses per device batch size as 4 (2B,4B) or 2 (8B) with a similar gradient accumulation strategy as SFT.