CARP: CONTINUOUS LATENT ROUTING FOR MIXTURE-OF-LORA UNDER NON-STATIONARY OPPONENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Specialising LoRA adapters to non-stationary opponents is commonly done by *hard-routing* each forward pass through the adapter selected by a predicted cluster label. We show this interface is structurally fragile: the predicted-cluster map is non-Lipschitz, so a small error in the inferred context can flip the cluster and silently swap adapters. We propose CARP, which instead routes through a soft mixture of LoRA prototypes indexed by a continuous latent c, with weights $\mathrm{softmax}(-\|c - \mu_k\|^2/\tau)$ trained by an EMA teacher–student on the realised context. The map is L_w-Lipschitz, so context error β enters the regret as $O(L_w\beta)$ rather than as a 0–1 flip. We prove a two-sided guarantee: a Lipschitz upper bound for CARP and an $\Omega(1)$ per-round lower bound for *any* hard router under non-zero predictor noise. Empirically the separation appears exactly where the theory places it. On a controlled strategic game (30 seeds, 6 baselines) CARP beats hard routing by $+2.6$ LTR out-of-distribution and is 5–9$\times$ less sensitive to predictor noise; a c-conditioned *linear* gate recovers most of the gain, so the interface, not the kernel, carries it. On MovieLens-1M, Last.fm-1K and Amazon-Books the predicted cross-over materialises: CARP wins on drifting or noisy-boundary splits, hard routing wins on cleanly partitioned ones, and all routers tie where the context geometry is near-linear. At LM scale (Qwen2.5 0.5B–3B) hard routing on the inferred context is worse than no routing, and an oracle-context control confirms the mechanism: given the true context the hard router's penalty vanishes. The Lipschitz gain is therefore robustness to context-inference error—which also locates the LM-scale bottleneck in context inference, not routing.

1 INTRODUCTION

Conditional low-rank adaptation (LoRA, Hu et al., 2022) is widely used to specialise a shared backbone under varying task or opponent conditions. In multi-agent RL, recommendation, and strategic LLM agents, a prevalent design pattern is to (a) cluster offline unlabeled trajectories into m discrete *regimes* via k-means or a learned classifier and (b) *hard-route* each forward pass through the single adapter selected by the predicted cluster, in the spirit of type-conditioned expert mixtures (He et al., 2016) and top-k expert routing (Shazeer et al., 2017; Fedus et al., 2022). We show this interface is structurally fragile: the predicted-cluster mapping is non-Lipschitz in the predictor logits, so infinitesimal representation drift can flip the cluster assignment and silently swap adapters, causing high-variance OOD performance and steep degradation under predictor noise.

A simple replacement is *soft mixture-of-LoRA routing* indexed by a continuous Gaussian latent $c \in \mathbb{R}^{d_c}$ over M learned prototypes $\mu_1, \dots, \mu_M$, with weights $w_k(c) = \mathrm{softmax}(-\|c - \mu_k\|^2/\tau)_k$. The mapping is L_w-Lipschitz with $L_w \le 2 \max_k \|\mu_k\|/\tau$, so a small predictor error $\beta_t = \|\hat{c}_t - c_t^\star\|$ contributes $O(L_w\beta_t)$ to the regret rather than a 0–1 cluster flip. We call this routing module CARP (**C**ontinuous **A**nd **R**egime-aware **P**rediction). The rest of the paper shows that CARP: **(i)** admits a two-sided guarantee: a Lipschitz contextual regret upper bound (Thm. 2) that recovers the switching-experts bound at $\tau \to 0$, and a lower bound (Thm. 3) that *any* hard router pays $\Omega(1)$ per-round regret under non-zero predictor noise while a Lipschitz router stays $O(L_w\beta)$; **(ii)** on a controlled strategic game (30 seeds, 6 baselines) beats hard routing and a same-capacity LearnedGate, holds where every baseline collapses on an adversarial split, and is 5–9$\times$ less predictor-noise-sensitive

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.

across two orders of magnitude in τ (§6); **(iii)** owes this gain to the Lipschitz *interface*, not the distance kernel: a c-conditioned linear gate recovers most of it and reproduces CARP's exact win/loss pattern (App. I); **(iv)** transfers to real users: on MovieLens-1M (30 seeds) CARP beats every non-c baseline on all splits and HDR on all but the cleanly partitioned ones, the cross-over Thm. 3 predicts (§6.3); **(v)** has bounded scope, mapped rather than hidden: Last.fm-1K reproduces the cross-over, Amazon-Books-2023 is ITT-negative except on drift-aware splits, and CraigslistBargain's near-linear geometry makes all routers tie (Tab. 3); **(vi)** at LM scale (Qwen2.5 0.5B–3B) shows hard routing on the inferred context is worse than no routing, and an oracle-context control shows the hard penalty vanishes given the true context—so the gain is robustness to context-inference error, and context inference, not routing, is the LM-scale bottleneck (§7).

Three structural pathologies of hard discrete routing. Non-Lipschitz mapping. A small prediction error δ can flip $\hat{z}$ across a k-means boundary, invoking a completely different adapter. Table 2 (5 seeds) shows HDR's long-term return falls at slope -9.68 LTR per unit predictor-noise probability, vs. CARP's -1.11 to -1.81 across five τ values—a 5–9× improvement.

k-means clusters overfit the training population. When the test opponent distribution shifts—routine under self-play—the centroids no longer cover the data: across 30 seeds HDR's OOD return is $+0.20 \pm 1.63$ vs. CARP's $+2.80 \pm 0.60$, a 2.7× seed std driven by coverage gaps (per-seed in App. B). The *hybrid-mix*/*hybrid-switch* splits, which interpolate in-dist and OOD within/across episodes, show the same pattern (HDR -1.19/-0.35 vs. CARP, $p < 10^{-3}$), so the failure is not tied to one OOD geometry.

Cannot track within-episode drift. Real adversaries adapt within a single trajectory. A discrete regime label, snapped to a single z, ignores the realised drift; CARP's per-turn c_t from a teacher–student EMA tracks it.

Method overview and scope. CARP (§4) comprises (a) a continuous Gaussian latent $q_\psi(c \mid h_t) = \mathcal{N}(\mu_\psi, \mathrm{diag}\sigma^2_\psi)$ replacing the categorical predictor; (b) M learned prototypes $\{\mu_k\}$, each with a low-rank delta, mixed by top-K softmax weights; (c) teacher–student Gaussian-NLL c-supervision on the realised $c^\star_t$ (a SimCLR add-on *hurts*, App. E). CARP targets routing geometry only: SFT, reward, GRPO, backbone and LoRA factorisation are held fixed, and LearnedGate/LG-c isolate c and the kernel at identical MoLoRA capacity. CFR solvers (Brown et al., 2019; Moravčík et al., 2017) target zero-sum equilibria, a complementary problem. We make no deployment claim at LM scale; per-seed collapse rates and CIs are reported throughout.

2 Related Work

MoE and LoRA routing. Sparse expert routing (Shazeer et al., 2017; Bengio et al., 2013) and low-rank adapters (Hu et al., 2022; Dai et al., 2024) are the dominant interfaces for conditional specialisation. Most MoE/LoRA routers select experts from a learned linear gate on the hidden state or a categorical task indicator (Switch-T (Fedus et al., 2022), Soft-MoE). CARP couples a MoLoRA mixture to a *learned latent context* c via a Lipschitz softmax-of-distance; the LearnedGate, Switch and Soft-MoE ablations isolate this design choice at identical capacity.

Opponent / context latents in multi-agent RL. DRON (He et al., 2016), policy embeddings (Grover et al., 2018) and LIAM (Papoudakis et al., 2021) learn an opponent latent but feed it as just another input to a single shared policy (DRON-MoE gates a few experts on it, without any continuity guarantee). CARP's update is routed *through* a c-indexed LoRA mixture, and the dependence is reflected in $L_w \sum_t \beta_t$ of Theorem 2; expert mixtures that route by a discrete gate—top-k selection (Shazeer et al., 2017; Fedus et al., 2022) or a predicted opponent type—use the interface we identify as fragile.

Routing fragility under distribution shift. Recent MoE work has documented routing instability under distribution shift (Fedus et al., 2022), motivating load-balancing and noise injection during training. Soft-MoE (Puigcerver et al., 2024) replaces argmax with a continuous softmax-weighted assignment, but conditions on the hidden state rather than on a continuous *learned* latent context. Trained with their own load-balance / soft-assignment tricks, both still lose $+2.9$ to $+6.0$ LTR to CARP at identical MoLoRA capacity (§6.1, App. H).

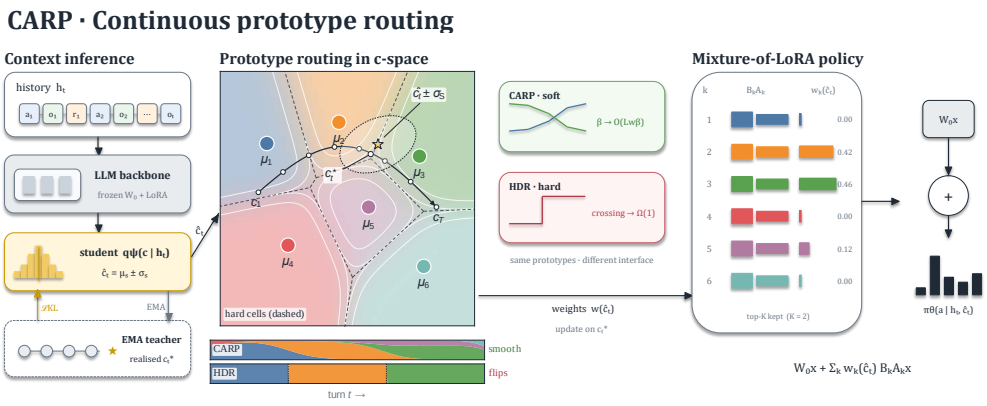


Figure 1: **CARP overview.** A student reads h_t through the backbone and outputs a Gaussian $\hat{c}_t$, supervised by an EMA teacher's realised $c_t^\star$ (§4.3). In c-space, M prototypes μ_k induce the smooth field $w_k(c) = \mathrm{softmax}_k(-\|c - \mu_k\|^2/\tau)$ (colour); a hard router uses the Voronoi cells (dashed) and flips adapter at every crossing (bottom strips)—the $\Omega(1)$ penalty of Thm. 3. $w(\hat{c}_t)$ mixes the top-K LoRA deltas $B_k A_k$ on the frozen W_0 (§4.2).

Non-stationary online learning. Online MD with experts (Hazan & Kale, 2009) and switching-expert analyses (Freund & Schapire, 1997; Blum, 1997; Koolen et al., 2014) charge $O(L_T \log K)$ for regime changes. Path-length bounds (Chiang et al., 2012; Rakhlin & Sridharan, 2013) and the RVU property (Syrgkanis et al., 2015) charge regret to the variation of the loss sequence rather than to the horizon. Our bound combines a per-prototype RVU with a Lipschitz routing penalty; $\tau \to 0$ recovers switching-expert (§5).

Strategic LLM agents and poker AI. PokerGPT (Huang et al., 2024), Cicero (Meta Fundamental AI Research Diplomacy Team (FAIR), 2022), PokéChamp (Karten et al., 2025), strategic-reasoning benchmarks (Duan et al., 2024; Costarelli et al., 2024) and CFR solvers (Brown et al., 2019; Moravčík et al., 2017) target reward functions or equilibria, not the routing interface studied here.

3 Problem Setup: Non-Stationary Latent-Conditioned Routing

Setup. A learner at round t observes history h_t and chooses $a_t \in \mathcal{A}$ from K macro actions; the opponent/environment is summarised by a hidden regime $c_t \in \mathcal{C}$. A shared backbone has M LoRA prototypes $\{\Delta W_k\}$ and a router $w(\cdot) : \mathcal{C} \to \Delta^{M-1}$ that combines them. We assume a context predictor $\hat{c}_t = \hat{c}(h_t)$, a realised supervision signal $c_t^\star$ (available at update time) with round-by-round error $\beta_t = \|\hat{c}_t - c_t^\star\|$, and bounded per-round losses $\|\ell_t\|_\infty \le 1$. The router induces effective adapter $\Delta\theta = \sum_k w_k \Delta W_k$.

Hard discrete routing (HDR). The discrete-regime baseline k-means-clusters offline trajectories into m regimes and routes via $w_k(c) = \mathbb{1}\{k = \arg\min_j \|c - \mu_j\|\}$. This map is non-Lipschitz: an ϵ-perturbation straddling a Voronoi boundary flips the adapter, contributing $O(1)$ regret.

Continuous-latent soft routing (CARP). We replace cluster assignment with a Gaussian latent and a softmax-of-distance router $w_k(c) = \mathrm{softmax}(-\|c - \mu_k\|^2/\tau)_k$, L_w-Lipschitz with $L_w \le 2 \max_k \|\mu_k\|/\tau$, so per-round routing error contributes $O(L_w \beta_t)$ rather than $O(1)$. The hard limit $\tau \to 0$ recovers HDR. $\mathrm{Var}(N_k) = \sum_{t \in N_k, t \ge 2} \|\ell_t - \ell_{t-1}\|_\infty$ is the per-prototype path length in the sense of Chiang et al. (2012); Rakhlin & Sridharan (2013).

4 METHOD: CARP

4.1 CONTINUOUS STRATEGIC LATENT

We posit a continuous latent style coordinate $c \in \mathbb{R}^{d_c}$ that summarises the opponent's behaviour. The predictor outputs a Gaussian

$$q_\psi(c \mid h_t) \;=\; \mathcal{N}\big(\mu_\psi(h_t),\ \mathrm{diag}(\sigma^2_\psi(h_t))\big). \tag{1}$$

The marginal q may be a true Gaussian mixture by tying the predictor head to multiple sub-heads; we use one head plus per-prototype LoRA, which yields an implicit mixture in policy space (§4.2).

4.2 MIXTURE-OF-LORA ROUTING

We replace one layer of the LLM actor (typically the action head) with

$$\textsc{MoLoRA}(x; c) \;=\; W_0 x \;+\; \sum_{k=1}^{M} w_k(c)\, B_k A_k x, \qquad w_k(c) = \mathrm{softmax}\Big(-\tfrac{\|c-\mu_k\|^2}{\tau}\Big)_k, \tag{2}$$

where $A_k \in \mathbb{R}^{r\times d_{\text{in}}}$, $B_k \in \mathbb{R}^{d_{\text{out}}\times r}$ are per-prototype LoRA factors, W_0 is the frozen base, and $\{\mu_k\}_{k=1}^M$ are learned prototypes. We further sparsify via top-K routing ($K = 2$ in our experiments), saving 50–75% of LoRA compute without affecting performance.

Lipschitz property. Because $w_k(c)$ is the softmax of a quadratic, it is $\big(\frac{2\max_k \|\mu_k\|}{\tau}\big)$-Lipschitz in c. The induced policy logits are therefore Lipschitz in c as well — this is the structural property absent from hard discrete routing (HDR), and the formal lever that drives the regret bound (§5).

4.3 AUXILIARY C-SUPERVISION VIA TEACHER–STUDENT NLL

At LM scale, the student has only prefix h_t available; an EMA teacher with full trajectory $h_1, \ldots, h_T$ supplies the realised context $c^\star_t = \mu_{\mathrm{T}}(h_1, \ldots, h_T)$ to supervise the student via Gaussian NLL:

$$\mathcal{L}_{\mathrm{KL}} \;=\; \tfrac{1}{2}\mathbb{E}\big[(\mu_{\mathrm{S}} - c^\star)^\top \Sigma^{-1}(\mu_{\mathrm{S}} - c^\star) + \log\det\Sigma\big]. \tag{3}$$

At synthetic scale where the true c is available, $c^\star$ is the ground-truth opponent style at turn t; the EMA teacher mirrors the same signal at LM scale where ground truth is hidden. We initialise the teacher equal to the student and update via EMA: $\theta_{\mathrm{T}} \leftarrow \alpha\theta_{\mathrm{T}} + (1-\alpha)\theta_{\mathrm{S}}$ with $\alpha = 0.99$.

A 5-seed ablation (Tab. 14, App. E) shows that (a) removing $\mathcal{L}_{\mathrm{KL}}$ entirely costs $+3.64$ LTR on OOD and $+5.93$ on adversarial ($p < 10^{-3}$, $5/5$ paired wins), and (b) adding a SimCLR contrastive term on μ from two prefix views *hurts* OOD by $+1.11$ LTR and adversarial by $+3.54$ ($p < 10^{-3}$); we therefore use $\mathcal{L}_{\mathrm{KL}}$ alone.

Routed update. At *acting* time we use $c_{\mathrm{act}} = \mu_{\mathrm{S}}(h_t)$ (predicted); at *update* time we use the teacher's $c^\star_t = \mu_{\mathrm{T}}$ (realised; at synthetic scale equal to the ground truth). The predicted-for-act, realised-for-update split mirrors the standard counterfactual update in routed contextual bandit / GRPO recipes, here applied in the continuous-latent setting.

The complete training step is summarised in Appendix A.

5 THEORY: CONTINUOUS CONTEXTUAL REGRET BOUND

We give the main theoretical guarantee here; the full proof and assumptions are in Appendix M; the argument applies the RVU property of Syrgkanis et al. (2015) prototype by prototype and adds a Lipschitz routing term.

Position of the bound. Our regret guarantee is, to our knowledge, the first *explicit Lipschitz-routing* bound for mixture-of-LoRA actors with a continuous latent context. It is a *strict generalisation* of two classical endpoints: at $\tau \to 0$ the bound recovers the $O(L_T \log K)$ switching-experts regret of

Freund & Schapire (1997); Blum (1997); Koolen et al. (2014) (where L_T counts adapter swaps); at $\tau \to \infty$ the routing collapses to uniform mixture and the bound reduces to a single-expert OMD bound (Hazan & Kale, 2009). The intermediate regime (finite τ) is the new contribution: the regret degrades *linearly* in the predictor error β_t via L_w rather than via the 0–1 flip suffered by hard discrete routing, and the constant $L_w \leq 4R/\tau$ (Lemma 1) makes the trade-off between routing softness and noise sensitivity explicit. This unification of switching-experts and continuous online MD as two endpoints of a single Lipschitz family is what we formally prove below; the empirical τ-sweep (§6.2) directly confirms the predicted direction.

Lemma 1 (Lipschitz constant of softmax-of-distance routing)**.** *Let $w(c) = \mathrm{softmax}_k(-\|c-\mu_k\|^2/\tau)$ over prototypes $\{\mu_k\}_{k=1}^M \subset \mathbb{R}^{d_c}$ with $\|\mu_k\| \leq R$, and let $c, c' \in \mathbb{R}^{d_c}$ with $\|c\|, \|c'\| \leq R$. Then*

$$\|w(c) - w(c')\|_1 \;\leq\; L_w \, \|c - c'\|_2, \qquad L_w \;\leq\; \frac{2(R + \max_k \|\mu_k\|)}{\tau} \;\leq\; \frac{4R}{\tau}.$$

Moreover, the inequality is tight up to a constant when $\max \|\mu_k\| = R$ and the prototypes are well-separated.

The proof (Appendix M) bounds the softmax Jacobian via $\|\partial_c w\|_{1\to 2} \leq 2\tau^{-1} \sum_k w_k \|c - \mu_k\|$ and applies the data-domain radius. The bound separates two intuitive quantities: the prototype radius R (capacity of c's support) and the routing softness τ (interpolation aggressiveness). **Three consequences for the regret analysis:** (a) L_w is *independent of* M, so larger prototype banks do not inflate the Lipschitz penalty; (b) hard routing corresponds to $\tau \to 0$, $L_w \to \infty$, recovering one-hot regret (any $\beta_t > 0$ can flip the assignment); (c) for finite τ the regret of a small predictor error is at most linear in β_t rather than the worst-case 0–1 flip.

Theorem 2 (CARP contextual regret; informal)**.** *Assume bounded losses $\|\ell_t\|_\infty \leq 1$, Lipschitz routing $\|w(c) - w(c')\|_1 \leq L_w \|c - c'\|_*$ with $L_w \leq 2 \max_k \|\mu_k\|/\tau$, and a per-prototype RVU property (Syrgkanis et al., 2015) for each prototype's active set N_k. Then the cumulative regret of* CARP *satisfies, with the balanced step size,*

$$R_T \;\leq\; O\Big(\sqrt{\big(M \log K + L_w \textstyle\sum_t \beta_t\big)\big(\sum_k \mathrm{Var}(N_k) + 1\big)}\Big),$$

where $\beta_t = \|\hat{c}_t - c_t^\star\|_*$ is the student–teacher alignment error.

Three structural consequences. **(i) Hard-routing limit.** As $\tau \to 0$ and c concentrates on prototypes, $w_k(c)$ collapses to one-hot and β_t becomes an integer mis-cluster count L_T; the bound's Lipschitz term reduces to the classical switching-experts penalty $O(L_T \log K)$ (Freund & Schapire, 1997; Blum, 1997; Koolen et al., 2014) up to constants, so hard discrete routing (HDR) is a strict special case that we recover at $\tau \to 0$. **(ii) Training/deployment coupling.** The $L_w \sum_t \beta_t$ term is *directly* what the Gaussian NLL c-supervision $\mathcal{L}_{\mathrm{KL}}$ (Eq. 3) minimises, tightly coupling training-time predictor accuracy with deployment-time exploitability. The ablation in Table 14 confirms this: removing $\mathcal{L}_{\mathrm{KL}}$ collapses OOD by -3.64 LTR. **(iii) Routing-softness premium.** $L_w \leq 2 \max \|\mu_k\|/\tau$ means larger τ shrinks the penalty for β_t; the trade-off is implicit in $\mathrm{Var}(N_k)$ (softer mixture $\Rightarrow$ larger effective N_k). We empirically confirm this trade-off via the noise-robustness τ-sweep of Table 2.

5.1 A MATCHING LOWER BOUND: HARD ROUTING IS FRAGILE

Theorem 2 *upper-bounds* CARP's regret as a linear function of the predictor error β_t. We now show the *lower bound* that hard discrete routing under any non-zero representation noise is provably $\Omega(1)$ *per round* in the worst case, while a Lipschitz router degrades only as $O(L_w \beta_t)$. This is what makes the comparison of CARP vs. HDR a sharp asymptotic separation, not a constant-factor tuning question.

Theorem 3 (Lower bound: hard routing pays a constant penalty under any non-zero predictor noise)**.** *Fix any $M \geq 2$ prototypes $\{\mu_k\}_{k=1}^M$ with $\min_{j \neq k} \|\mu_j - \mu_k\|_2 \geq 2\delta > 0$ (cluster separation), and let π_k denote the per-prototype policy of any hard router whose output is $\pi(c) = \pi_{\arg\min_k \|c - \mu_k\|}$. There exists a loss family $\{\ell_t\}_t \in [0,1]$ and a predictor-noise distribution $\hat{c}_t = c_t^\star + \eta_t$ with $\mathbb{E}\|\eta_t\|_2 = \beta$ such that for any $\beta \in (0, \delta)$ the expected per-round regret satisfies*

$$\mathbb{E}\left[\langle \pi(\hat{c}_t) - \pi(c_t^\star), \ell_t \rangle\right] \;\geq\; \tfrac{1}{2} \Pr\big[\hat{c}_t \text{ crosses a Voronoi boundary}\big] \;\geq\; \tfrac{1}{2} \tfrac{\beta}{2\delta} \overset{\beta = \Theta(\delta)}{=} \Omega(1).$$

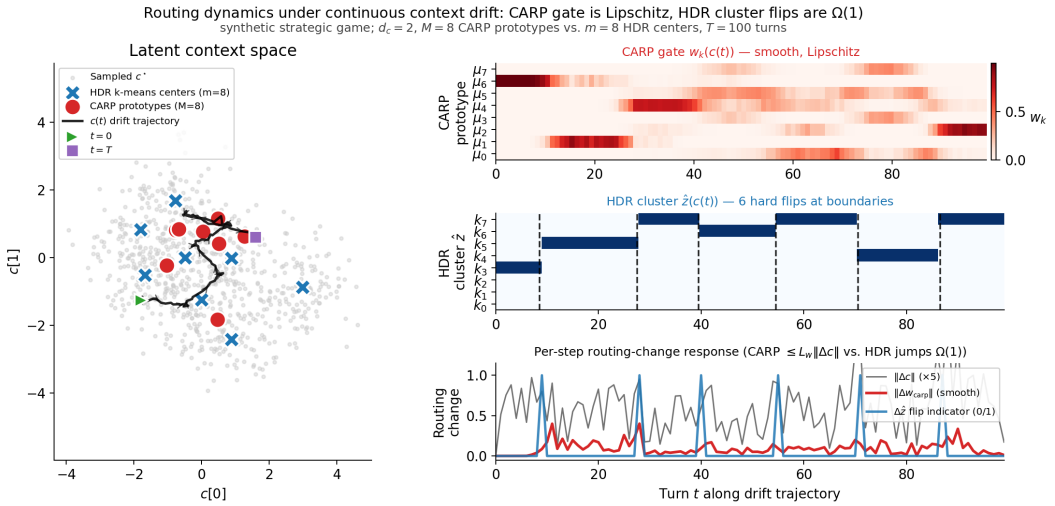


Figure 2: **Visual illustration of the upper / lower bound separation under continuous context drift.** Left: a T=100-turn trajectory through latent context space crosses six Voronoi cell boundaries of the HDR k-means partition. Top right: CARP's gate $w_k(c(t))$ is a smooth function of c, whose change per step is bounded by Lemma 1's Lipschitz constant $L_w \le 4R/\tau$ (mean per-step $\|\Delta w\|_1 = 0.09$). Middle right: HDR's discrete cluster index $\hat{z}(c(t))$ has 6 hard flips at boundary crossings; each flip is the $\Omega(1)$ penalty event of Theorem 3. Bottom right: per-step routing change overlaid with $\|\Delta c\|$ makes the separation visible—CARP tracks c smoothly, while HDR contributes 0 outside flips and 1 at each crossing. This panel directly visualises the asymptotic comparison $O(L_w\beta)$ vs. $\Omega(1)$ that the theorems formalise.

In contrast, any Lipschitz router with constant L_w (Lemma 1) satisfies

$$\mathbb{E}\left[\langle w(\hat{c}_t) - w(c_t^\star), \ell_t\rangle\right] \;\le\; L_w \cdot \beta,$$

which is $O(\beta)$ for fixed τ.

The proof is short (Appendix M.3): the worst-case loss assigns reward 1 to the prototype-$k^\star$ action and reward 0 to all others; any boundary crossing then incurs reward 0 instead of 1. The crossing probability is at least $\beta/(2\delta)$ by a one-dimensional projection of η_t onto the Voronoi normal. Taking $\beta = \Theta(\delta)$ makes the regret $\Omega(1)$ per round, hence $\Omega(T)$ cumulative.

Discussion: this is what *cannot* be patched in HDR by training tricks. The lower bound is information-theoretic about the routing *interface*: no choice of predictor, training loss, or per-prototype policy avoids paying $\Omega(1)$ per boundary crossing once $\beta \ge \delta$. Lipschitz routing trades $\Omega(1)$-per-crossing for L_w-per-unit-β, the right choice whenever $L_w\beta < 1$ (i.e., $\beta < \tau/(2\max\|\mu_k\|)$). The τ-sweep of Table 2 empirically maps this trade-off: CARP's slope $|\mathrm{dLTR}/\mathrm{d}p|$ never crosses HDR's even at the smallest τ we test.

6 Synthetic Strategic Game

Setup. We construct a 2D continuous-style strategic game (details in Appendix A). Each episode samples an opponent style $c \in \mathbb{R}^2$ from one of three regimes: *in-dist* ($c \sim \mathrm{Cat}(4\text{ prototypes}) + \mathcal{N}(0, \sigma)$, training distribution), *OOD* (4 prototypes shifted out by $\times 2.2$ and rotated $\pi/4$), and *drift* ($c_t = c_0 + 0.05\sum_{s\le t}\xi_s$ within the episode). The hero observes the last 4 opponent actions; reward is a quadratic function of the relative orientation of hero embedding and opponent style; the optimal hero plays $\mathrm{softmax}(\langle e_a, c_t\rangle)$. We compare four methods at identical backbone, batch size, optimiser, and 1000 training steps: **HDR** (hard-discrete router, m=5 k-means regimes with routed GRPO); **CARP** (M=8 prototypes, d_c=2, τ=0.5, top-K=2, EMA α=0.99); **NoRouter** (single LoRA, no predictor); and **LearnedGate** (same MoLoRA as CARP, weights from a learned linear gate

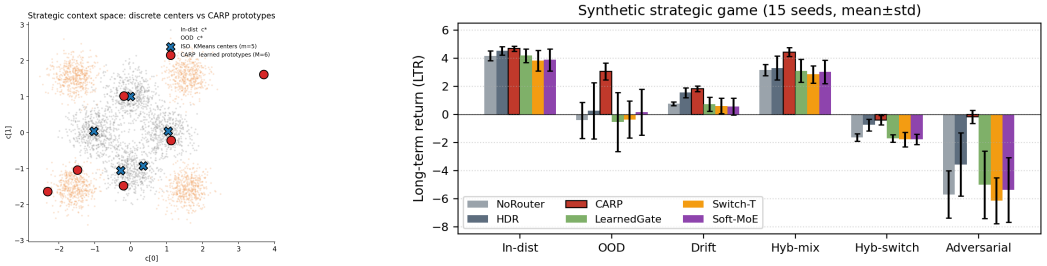


Figure 3: (a) Learned CARP prototypes (red ×) overlaid on the 4D PCA-projected hidden representation. (b) Mean±std LTR per split across 6 methods (15 seeds); CARP (red, outlined) is the only method that holds non-negative LTR on the worst-case *adversarial* split.

$\mathrm{softmax}(g(h_t))$ instead of $\mathrm{softmax}(-\|c-\mu_k\|^2/\tau)$—isolates the contribution of c from MoLoRA capacity alone).

6.1 Main results

Table 1: Long-term return (LTR, higher is better) on the synthetic game at 1000 steps, 1024 eval episodes per cell, mean±std over **30 seeds**. CARP is best on every split; the largest gaps are OOD and the worst-case *adversarial* split (opponent styles minimising a uniform policy's return; App. A), where every baseline collapses to -4.3 to -6.0 while CARP holds at -0.50. p: two-sided paired bootstrap (20,000 resamples); Wilcoxon confirms every $p<.01$ (App. E).

	In-dist	OOD	Drift	Hybrid-mix	Hybrid-switch	Adversarial
NoRouter	$+3.50_{\pm 2.13}$	$-0.79_{\pm 2.16}$	$+0.29_{\pm 1.56}$	$+2.59_{\pm 2.03}$	$-1.94_{\pm 1.16}$	$-5.83_{\pm 2.67}$
LearnedGate (no c)	$+4.17_{\pm 0.48}$	$-0.62_{\pm 1.76}$	$+0.68_{\pm 0.47}$	$+3.08_{\pm 0.71}$	$-1.72_{\pm 0.25}$	$-5.63_{\pm 1.99}$
HDR (m=5)	$+4.44_{\pm 0.33}$	$+0.20_{\pm 1.63}$	$+1.44_{\pm 0.32}$	$+3.15_{\pm 0.76}$	$-0.81_{\pm 0.37}$	$-4.32_{\pm 1.80}$
Switch-T (top-1+LB)	$+3.93_{\pm 0.60}$	$-0.55_{\pm 1.29}$	$+0.62_{\pm 0.44}$	$+2.92_{\pm 0.52}$	$-1.76_{\pm 0.47}$	$-5.96_{\pm 1.48}$
Soft-MoE	$+3.90_{\pm 0.72}$	$-0.34_{\pm 1.68}$	$+0.52_{\pm 0.58}$	$+2.96_{\pm 0.76}$	$-1.82_{\pm 0.28}$	$-5.75_{\pm 2.27}$
CARP	$\mathbf{+4.68_{\pm 0.20}}$	$\mathbf{+2.80_{\pm 0.60}}$	$\mathbf{+1.76_{\pm 0.17}}$	$\mathbf{+4.34_{\pm 0.29}}$	$\mathbf{-0.46_{\pm 0.29}}$	$\mathbf{-0.50_{\pm 1.21}}$
$\Delta_{\text{CARP}-\text{LG}}$ (p)	$+0.51$ ($<10^{-3}$)	$\mathbf{+3.42}$ ($\mathbf{<10^{-3}}$)	$\mathbf{+1.08}$ ($\mathbf{<10^{-3}}$)	$\mathbf{+1.26}$ ($\mathbf{<10^{-3}}$)	$\mathbf{+1.26}$ ($\mathbf{<10^{-3}}$)	$\mathbf{+5.13}$ ($\mathbf{<10^{-3}}$)
$\Delta_{\text{CARP}-\text{HDR}}$ (p)	$+0.24$ ($<10^{-3}$)	$\mathbf{+2.60}$ ($\mathbf{<10^{-3}}$)	$+0.31$ ($<10^{-3}$)	$\mathbf{+1.19}$ ($\mathbf{<10^{-3}}$)	$+0.35$ ($<10^{-3}$)	$\mathbf{+3.82}$ ($\mathbf{<10^{-3}}$)
$\Delta_{\text{CARP}-\text{NR}}$ (p)	$+1.17$ ($<10^{-3}$)	$\mathbf{+3.59}$ ($\mathbf{<10^{-3}}$)	$\mathbf{+1.47}$ ($\mathbf{<10^{-3}}$)	$\mathbf{+1.75}$ ($\mathbf{<10^{-3}}$)	$\mathbf{+1.48}$ ($\mathbf{<10^{-3}}$)	$\mathbf{+5.33}$ ($\mathbf{<10^{-3}}$)

The principal headline number is the LearnedGate ablation: at identical MoLoRA architecture but with c-conditioned gating replaced by a learned linear gate on h_t, OOD LTR drops from $+2.80$ to -0.62 ($\mathbf{+3.42}$ LTR gap, $p<10^{-3}$, 95% CI $[+2.84, +4.01]$), drift drops by $+1.08$, both hybrid splits by $\sim+1.26$ each, and adversarial by $\mathbf{+5.13}$ LTR; this isolates the *continuous latent c supervised by the EMA teacher* from MoLoRA capacity. Against HDR at matched informational capacity, CARP retains a $\mathbf{+2.60}$ OOD (95% CI $[+1.99, +3.23]$) and $\mathbf{+3.82}$ adversarial advantage (95% CI $[+3.11, +4.48]$), is best on every split, with $3-8\times$ smaller seed variance than HDR on OOD, hybrid, and adversarial (per-seed numbers in App. B). No method is trained on the adversarial split; the continuous-latent gate generalises to opponents that regime-indexed baselines have no expert for.

Drift-aware training. A 30% drift mixture in training nearly doubles both methods' drift LTR; CARP's edge over HDR narrows from $+0.27$ to $+0.13$ (App. G).

6.2 Predictor-noise robustness and the τ sweep

We inject noise into the predictor output: with probability p, HDR's $\hat{z}$ is replaced by a uniformly random regime label; CARP's $\hat{c}$ is corrupted by $\mathcal{N}(0, p\sigma_c)$ where σ_c is the in-distribution standard deviation of c. We fit LTR as a linear function of p and report the slope (LTR per unit p) per method.

CARP's smallest slope (1.11 at τ=2) is $8.7\times$ smaller than HDR's 9.68; the largest (1.81 at τ=0.25) still $5.3\times$ smaller (Fig. 4, App. B). The empirical fit of $|\text{slope}|$ vs $1/\tau$ is sublinear (R^2=0.38), suggesting most robustness comes from neighbour-prototype fill-in rather than the Lipschitz bound alone. Capacity sweep $M \in \{4, 8, 16\}$ and standard MoE community baselines (Switch-T (Fedus

Table 2: Noise-robustness slope per routing softness τ (600 train steps, **5 seeds**, eval over 1024 episodes per noise level $p \in \{0, 0.1, 0.3, 0.5, 0.7\}$). All CARP settings are 5–9× less noise-sensitive than HDR, with the slope $|\mathrm{dLTR}/\mathrm{d}p|$ approximately monotonic in τ over two orders of magnitude.

Method	τ	LTR at p=0	|slope|
HDR	–	$+4.46 \pm 0.26$	$\mathbf{9.68 \pm 0.54}$
CARP	0.10	$+4.33 \pm 0.21$	1.68 ± 0.68
CARP	0.25	$+4.45 \pm 0.13$	1.81 ± 0.44
CARP	0.50	$+4.41 \pm 0.27$	1.51 ± 0.34
CARP	1.00	$+4.45 \pm 0.15$	1.49 ± 0.55
CARP	2.00	$+4.37 \pm 0.16$	$\mathbf{1.11 \pm 0.30}$

et al., 2022), Soft-MoE) in Appendices B, H: CARP beats Switch by $+0.85$ (in-dist) to $\mathbf{+5.98}$ (adversarial) and Soft-MoE by $+0.79$ to $\mathbf{+5.19}$ (adversarial) LTR (paired bootstrap $p < 10^{-3}$ on every cell), establishing that the gain is the c-conditioning of the gate, not the load-balance trick or soft-mixing per se.

Scaling to 4 simultaneous opponents. At d_c=8 (4 concatenated opponents, 10 seeds, 1500 steps; App. F), CARP beats LearnedGate by $+0.86$ to $+0.93$ LTR on every split ($p < 10^{-3}$) and HDR on in-dist, OOD and hybrid-mix ($p \leq .017$); on drift and hybrid-switch the two tie, since averaging over 4 opponents smooths the k-means coverage gaps that dominated at d_c=2.

6.3 Second domain: contextual recommendation on MovieLens-1M

To stress-test CARP against a *different, non-strategic* domain with *real user data*, we build a contextual-recommendation benchmark from MovieLens-1M (Harper & Konstan, 2015) that shares only the API with the synthetic game (continuous latent context, in-dist / OOD / drift / hybrid splits). Each episode is one of 6,040 users; the hero recommends one of K=18 movie genres per turn and gets the user's centered mean rating of that genre as reward (squashed by $\tanh$). The user's hidden style $c \in \mathbb{R}^4$ is the top-4 PCA of their genre-preference vector. The dataset is split by k-means (k=8) on c: in-dist = top-3 clusters (3,240 users); OOD = bottom-3 (1,426); drift = early-vs-late half-ratings; hybrid-mix and hybrid-switch as in the synthetic env. Training: 800 PG steps, batch 128, 30 seeds. Same router hyperparameters as Table 1.

Result. On real users, CARP wins all 5 splits against every non-c baseline (LearnedGate, Switch, Soft-MoE, NoRouter) and 4 of 5 against HDR: on OOD $\Delta_{\text{CARP-HDR}}$=$+0.67$ (95% CI $[+0.30, +1.05]$) and $\Delta_{\text{CARP-NONC}} \in [+1.71, +2.08]$ (all $p < 10^{-3}$); on the two hybrid splits $+0.28$/$+0.44$ over HDR and $\geq +1.02$ over every non-c router. HDR retakes only the intra-cluster drift split ($+3.53$ vs $+3.34$), which crosses no cluster boundary and so never penalises a hard index. Gaps are smaller than on synthetic ($+2.60$/$+3.42$) but significant (Table 24, App. J).

Stronger piecewise-in-c drift. The original drift split is locally smooth in c. To probe Thm. 3 directly we add three piecewise-in-c splits (Table 24, App. J): TEMPORAL_OOD (top-quartile drift users on late-half c), DRIFT_CROSS (in-dist→OOD cluster jump at t=$T/3$) and QUARTERLY_SHIFT (four cleanly partitioned segments). Across 30 seeds CARP beats HDR by $\mathbf{+0.45}$ and $\mathbf{+0.43}$ on the first two, while HDR *wins* $\mathbf{+0.25}$ on QUARTERLY_SHIFT (all $p < 10^{-3}$, App. J)—as Thm. 3 predicts: clean segments are zero-noise, so the hard router pays nothing and CARP's interpolation pays a small bias. Against every non-c baseline CARP wins all three splits at $p < .005$ (Fig. 6).

7 LM-scale Experiments

Setup. CARP replaces the action head of Qwen2.5 (0.5B/1.5B/3B) with MoLoRA (M=8, K=16) atop LoRA (r=8 on q/v; ∼0.07% trainable, bf16) vs. HDR (m=5 k-means heads on $\hat{c}$), LearnedGate (LG) and NoRouter (NR) on (a) **Synthetic-LM** (prompted §6 game) and (b) **Mini-NLHE** (App. A); one 500-step recipe, 8 paired seeds per condition and scale (App. C).

Hard routing on $\hat{c}$ is the failure mode at LM scale too. HDR is below the plain head on $13/15$ split×scale cells (all 5 significantly at the 3B final checkpoint): a predicted-cluster→head switch

Table 3: **When does Lipschitz routing help?** Every result, placed on Thm. 3's two axes: does the c geometry have boundaries a hard router can cross, and does the routed context carry error β there. Δ = CARP−HDR (LTR) unless stated; negative = hard wins. Details: §6, §6.3, App. K, L, N, C.

Domain	Split(s)	c geometry	β at boundary	Predicted	Observed Δ
Synthetic	OOD, adversarial	piecewise	large (off-support)	CARP$\gg$HDR	$+2.60$; $+3.8$–5.5 vs. all
Synthetic	τ-sweep (injected)	piecewise	controlled	slope $\propto L_w$	-1.1..-1.8 vs. -9.7
MovieLens-1M	TEMP_OOD, DRIFT_CROSS	piecewise	> 0	CARP>HDR	$+0.45$, $+0.43$
MovieLens-1M	QUARTERLY_SHIFT; drift	clean / none	≈ 0	HDR$\geq$CARP	-0.25; -0.19
Last.fm-1K	IN_DIST/DRIFT/TEMP_OOD	smooth	> 0	CARP>HDR	$+0.88$, $+0.60$, $+0.39$
Last.fm-1K	OOD, HYBRID_MIX	clean segments	≈ 0	HDR>CARP	-1.09, -0.35
Amazon-Books	flat (ITT)	one cluster (76%)	rarely met	tie / HDR	-0.38 in-dist
Amazon-Books	drift-aware (stable)	piecewise	> 0	CARP>HDR	$+0.27$ to $+0.46$
CraigslistBargain	all	near-linear	no boundary	all routers tie	tie
Synth-LM, $\hat{c}$	0.5B / 3B-final	piecewise	large	HDR<NR<CARP	$+0.84$–2.10 / $+0.94$–1.10
Synth-LM, $c^\star$	0.5B / 3B	piecewise	0	HDR$^\star \approx$CARP$^\star \gg$NR	≤ 0 (ns); both $+1.6$–3.0 vs. NR
Mini-NLHE	3B	over-trained	—	no headroom	tie ± 0.4

on a Qwen backbone *hurts*. CARP beats HDR on $5/5$ splits at 0.5B ($\Delta = +0.84$ to $+2.10$, $4/5$ at $p \leq .008$), $5/5$ at 1.5B and, at 3B, $2/5$ at the best checkpoint but $5/5$ (3 significant) at the final one, where HDR degrades late—Thm. 3's separation at LM scale.

The gain is robustness to $\hat{c}$ error, bounded by $\hat{c}$ quality. Routing CARP and HDR on the true $c^\star$ (App. C) tests Thm. 3 in both directions: at β=0 the hard penalty vanishes—HDR* gains $+2.5$ to $+3.3$ over HDR (all $p < .001$) and is at least as good as CARP* (in-dist -0.68, $p = .008$)—while on $\hat{c}$ the same router is the worst condition. The Lipschitz interface is not a better policy class; it bounds the damage from $\hat{c}$ error (Thm. 2). The same control explains scale: CARP−NR is $+0.60$ to $+1.05$ at 0.5B ($5/5$, all $p < .05$), $\approx +0.3$ at 1.5B and ≈ 0 at 3B at every LoRA rank (App. C), yet both oracle routers still beat the plain head by $+1$–3 at 3B. Headroom does not vanish; the cost of routing on $\hat{c}$ grows with scale (LG, which ignores $\hat{c}$, matches CARP at 3B). The bottleneck is context inference, not routing.

Caveats and scope. MINI-NLHE (3B, 8 determ. seeds, Tab. 6): all conditions over-train within 200 steps and tie within ± 0.4; 7B collapses both arms (App. C). Above 0.5B the $\hat{c}$-routed gaps are directional only; we make no deployment claim. No-headroom regimes are indexed in App. D.

8 CONCLUSION

We isolated one design choice in adapter-based agents under non-stationary opponents: whether the inferred context reaches the mixture-of-LoRA through a hard cluster index or through a Lipschitz softmax-of-distance over learned prototypes. The Lipschitz interface turns the $\Omega(1)$ per-crossing penalty of any hard router into an $O(L_w \beta)$ term (Thms. 2–3), and the empirical gain tracks that error exactly: largest where $\hat{c}$ must cross boundaries it was not trained on, absent where boundaries are clean, and gone when both routers receive the true $c^\star$ (Tab. 3). At 3B the oracle control shows that routed adapters keep $+1$–3 points of headroom while the $\hat{c}$-routed gap closes, so the binding constraint is context inference. CARP is therefore not a stronger policy class but the interface that makes a mixture-of-LoRA robust to the context estimate it is given; improving that estimate at scale is the natural next step.

AI USE STATEMENT

In this work we used a generative AI coding assistant (Claude) for the following tasks. *Writing:* polishing and compressing prose in all sections, and drafting first versions of several appendix paragraphs that report experimental numbers; every sentence was reviewed and edited by the authors. *Code:* drafting experiment-launch shell scripts, aggregation/statistics scripts (paired bootstrap and Wilcoxon tests), the Last.fm-1K environment wrapper, the c-conditioned linear-gate ablation module, the LM-scale HDR/LearnedGate heads and oracle-context switch, and the interactive Gradio demo; all AI-drafted code was reviewed by the authors, smoke-tested against known outputs, and cross-checked for determinism where reported (App. D). *Research execution:* the assistant helped schedule and monitor multi-seed runs and produce summary tables from raw logs. We did *not* use generative

AI to generate research ideas, to prove or check the theorems (App. M), to generate any dataset, or to write related-work summaries without manual verification of each cited source. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

Reproducibility statement

All routing modules (§4) are implemented in a single shared module used unchanged by every experiment. The synthetic game, MovieLens-1M, Amazon-Books-2023, Last.fm-1K and CraigslistBargain environments are specified in App. A–L and released as code; every per-seed JSON log behind every table in the paper is included in the supplementary archive together with the exact launch scripts and aggregation scripts that produce the reported statistics. Proofs of Thms. 2 and 3 with all assumptions stated are in App. M. LM-scale runs use publicly available Qwen2.5 checkpoints with the hyper-parameters listed in App. C–C; per-seed collapse rates are reported rather than filtered.

References

Yoshua Bengio, Nicholas Léonard, and Aaron C. Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013.

Avrim Blum. Empirical support for Winnow and weighted-majority algorithms: Results on a calendar scheduling domain. *Machine Learning*, 26:5–23, 1997.

Noam Brown, Adam Lerer, Sam Gross, and Tuomas Sandholm. Deep counterfactual regret minimization. In *International Conference on Machine Learning (ICML)*, 2019.

Òscar Celma. *Music Recommendation and Discovery: The Long Tail, Long Fail, and Long Play in the Digital Music Space*. Springer, 2010. Last.fm Dataset – 1K users, `http://ocelma.net/MusicRecommendationDataset/lastfm-1K.html`.

Chao-Kai Chiang, Tianbao Yang, Chia-Jung Lee, Mehrdad Mahdavi, Chi-Jen Lu, Rong Jin, and Shenghuo Zhu. Online optimization with gradual variations. In *Proceedings of the 25th Annual Conference on Learning Theory (COLT)*, 2012.

Anthony Costarelli, Mat Allen, Roman Hauksson, Grace Sodunke, Suhas Hariharan, Carlson Cheng, Wenjie Li, Joshua Clymer, and Arjun Yadav. GameBench: Evaluating strategic reasoning abilities of LLM agents. *arXiv preprint arXiv:2406.06613*, 2024.

Damai Dai, Chengqi Deng, Chenggang Zhao, R. X. Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Y. Wu, Zhenda Xie, Y. K. Li, Panpan Huang, Fuli Luo, Chong Ruan, Zhifang Sui, and Wenfeng Liang. DeepSeekMoE: Towards ultimate expert specialization in mixture-of-experts language models. *arXiv preprint arXiv:2401.06066*, 2024.

Jinhao Duan, Renming Zhang, James Diffenderfer, Bhavya Kailkhura, Lichao Sun, Elias Stengel-Eskin, Mohit Bansal, Tianlong Chen, and Kaidi Xu. GTBench: Uncovering the strategic reasoning limitations of LLMs via game-theoretic evaluations. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2024.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022.

Yoav Freund and Robert E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997.

Aditya Grover, Maruan Al-Shedivat, Jayesh K. Gupta, Yuri Burda, and Harrison Edwards. Learning policy representations in multiagent systems. In *International Conference on Machine Learning (ICML)*, 2018.

F. Maxwell Harper and Joseph A. Konstan. The MovieLens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems*, 5(4):19:1–19:19, 2015.

Elad Hazan and Satyen Kale. Better algorithms for benign bandits. In *Proceedings of the Twentieth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pp. 38–47, 2009.

He He, Jordan Boyd-Graber, Kevin Kwok, and Hal Daumé III. Opponent modeling in deep reinforcement learning. In *International Conference on Machine Learning (ICML)*, 2016.

He He, Derek Chen, Anusha Balakrishnan, and Percy Liang. Decoupling strategy and generation in negotiation dialogues. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2018.

Yupeng Hou, Jiacheng Li, Zhankui He, An Yan, Xiusi Chen, and Julian McAuley. Bridging language and items for retrieval and recommendation. *arXiv preprint arXiv:2403.03952*, 2024.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022.

Chenghao Huang, Yanbo Cao, Yinlong Wen, Tao Zhou, and Yanru Zhang. PokerGPT: An end-to-end lightweight solver for multi-player Texas Hold'em via large language model. *arXiv preprint arXiv:2401.06781*, 2024.

Seth Karten, Andy Luu Nguyen, and Chi Jin. PokéChamp: an expert-level minimax language agent. *arXiv preprint arXiv:2503.04094*, 2025.

Wouter M. Koolen, Tim van Erven, and Peter D. Grünwald. Learning the learning rate for prediction with expert advice. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2014.

Meta Fundamental AI Research Diplomacy Team (FAIR). Human-level play in the game of Diplomacy by combining language models with strategic reasoning. *Science*, 378(6624):1067–1074, 2022.

Matej Moravčík, Martin Schmid, Neil Burch, Viliam Lisý, Dustin Morrill, Nolan Bard, Trevor Davis, Kevin Waugh, Michael Johanson, and Michael Bowling. DeepStack: Expert-level artificial intelligence in heads-up no-limit poker. *Science*, 356(6337):508–513, 2017.

Georgios Papoudakis, Filippos Christianos, and Stefano V. Albrecht. Agent modelling under partial observability for deep reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Joan Puigcerver, Carlos Riquelme, Basil Mustafa, and Neil Houlsby. From sparse to soft mixtures of experts. In *International Conference on Learning Representations (ICLR)*, 2024.

Alexander Rakhlin and Karthik Sridharan. Online learning with predictable sequences. In *Proceedings of the 26th Annual Conference on Learning Theory (COLT)*, 2013.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc V. Le, Geoffrey E. Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations (ICLR)*, 2017.

Vasilis Syrgkanis, Alekh Agarwal, Haipeng Luo, and Robert E. Schapire. Fast convergence of regularized learning in games. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2015.

A Algorithm and detailed environment specification

A.1 CARP training step (pseudocode)

Algorithm 1 One CARP training step.

Require: batch of trajectories $\{(h_t, a_t, r_t)\}_t^B$

$\mu_{\mathrm{S}}, \log \sigma_{\mathrm{S}}, \mathrm{pool} \leftarrow \mathrm{student}(h_t)$

$\mu_{\mathrm{T}} \leftarrow \mathrm{teacher}(\mathrm{pool})$ // detached

$\mathcal{L}_{\mathrm{KL}} \leftarrow \frac{1}{2}\sum[(\mu_{\mathrm{S}}-\mu_{\mathrm{T}})^2 e^{-2\log\sigma} + 2\log\sigma]$

$\ell_a \leftarrow \log \mathrm{softmax}\big(\textsc{MoLoRA}(\mathrm{pool}; \mu_{\mathrm{S}}.\mathrm{detach}())\big)$

$A_t \leftarrow \mathrm{normalize}(r_t)$ // GRPO-style group advantage

$\mathcal{L}_{\mathrm{total}} \leftarrow -\mathbb{E}[\ell_a(a_t)\cdot A_t] - \lambda_{\mathrm{ent}}\, H(\pi) + \lambda_{\mathrm{KL}}\, \mathcal{L}_{\mathrm{KL}}$ // λ_{KL}=0.5 default; NCE add-on tested and discarded (-1.11 OOD, App. E)

backward; AdamW step; teacher EMA update

A.2 Synthetic Strategic Game — environment details

State and action spaces. At each turn the opponent reveals one of $|A| = 4$ action embeddings $e_1, \ldots, e_4 \in \mathbb{R}^{d_e}$ with $d_e = 4$. Each action embedding is drawn once at the beginning of training and fixed. The hero's action space is the same $|A| = 4$ embedding set; the hero observes a sliding window of the last $L = 4$ opponent actions.

Reward. For a hero action a, opponent action o, and opponent style c, the reward is

$$r(a, o, c) \;=\; \tanh\big(\alpha\langle e_a, c\rangle\big)\cdot \mathbb{1}\{e_a \neq e_o\} - 0.1,$$

with $\alpha = 1.5$. The intuition is that hero embeddings aligned with the opponent's style score positively; embeddings clashing with the opponent's choice score zero. The -0.1 baseline normalises chance to roughly 0.

Style sampling. 4 in-distribution prototypes $\{c_k^{\mathrm{ID}}\} = \{(\pm 1, 0), (0, \pm 1)\}$ fix the unit cross in $\mathbb{R}^{d_c=2}$, with Gaussian jitter σ=0.35; OOD prototypes shift to radius 2.2 and rotate by $\pi/4$, with jitter σ=0.30. Drift trajectories use the random walk $c_{t+1} = c_t + 0.05\,\xi$, $\xi \sim \mathcal{N}(0, I)$, clamped to the unit ball. Two *hybrid* held-out splits test interpolation between clusters: `hybrid_mix` samples $c = \alpha c^{\mathrm{ID}} + (1-\alpha)c^{\mathrm{OOD}}$ with $\alpha \sim \mathcal{U}(0.25, 0.75)$; `hybrid_switch` starts each episode at an in-dist prototype and switches to a fresh OOD prototype at mid-episode. The `adversarial` split is a hero-agnostic worst-case: we draw $C = \max(8n, 256)$ candidate styles, project them onto a ring at radius 2.8 (well outside the in-dist ring at 1 and the OOD ring at 2.2), add Gaussian jitter of scale 0.35, and keep the n candidates with the smallest $\max_a\langle e_a, c\rangle$ score, i.e. those *furthest from any* action embedding's target direction; a uniform policy receives ≤ 0 expected reward on every retained c. The sampling is independent of which router is being evaluated, so all methods face the same adversarial distribution. None of the four held-out splits (`hybrid_*`, `adversarial`) appears in training.

Per-turn context in drift mode. The realised c_t at every turn is exposed by the rollout API and used as the supervision signal for μ_{T}. This is the implementation-level fix that enabled drift-aware training; without it, μ_{T} would converge to the initial c_0 even when the opponent's style had drifted.

A.3 Mini-NLHE case-study environment

Heads-up two-street No-Limit Hold'em with 40BB starting stacks and six bet templates ($\{\mathrm{fold}, \mathrm{check}, \mathrm{call}, \mathrm{bet}_{\frac{1}{3}}, \mathrm{bet}_{\frac{2}{3}}, \mathrm{bet}_{\text{all-in}}\}$). Each hand draws an opponent style $\in$ $\{\text{tight-passive}, \text{tight-aggressive}, \text{loose-passive}, \text{loose-aggressive}, \text{maniac}\}$ (in-dist) or $\{\text{rock}, \text{ultra-loose}\}$ (held-out OOD). Four *hybrid* held-out styles (convex mixtures and mid-hand drifts between archetypes) are evaluated post-training only. The reward is the net BB/hand outcome of the hero. HDR is given the true style id at training time; CARP must infer c from the visible text history.

B SENSITIVITY SWEEP

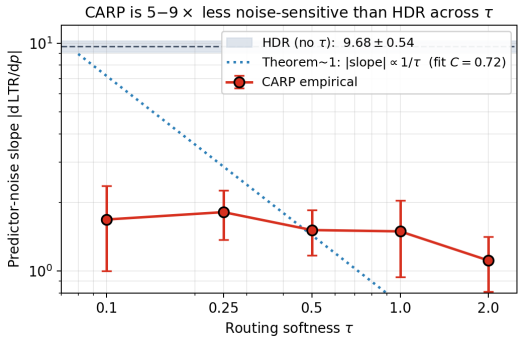


Figure 4: Predictor-noise slope $|\mathrm{dLTR}/\mathrm{d}p|$ versus routing softness τ on log-log axes (5 seeds each). CARP (red) lies $5-9\times$ below the HDR band over two orders of magnitude in τ. The empirical slope is sublinear in $1/\tau$, indicating most of CARP's noise robustness comes from soft mixing.

Table 4: Capacity sensitivity sweep. CARP strictly dominates HDR at every comparable capacity. "–" indicates the cell does not exist.

	m=3	m=5	m=8	M=8	M=16
HDR in-dist	2.81	3.18	3.11	–	–
CARP in-dist	–	–	–	3.53	3.66
HDR OOD	−1.51	−1.16	−1.32	–	–
CARP OOD	–	–	–	1.18	1.27

C LM-SCALE STABILITY RESCUE: PER-SEED RESULTS

Table 5: Synthetic-LM long-term return on Qwen2.5-3B (500 steps, 11 seeds, mean±std, eval over 128 episodes per split). $6/11$ CARP and $6/11$ NoRouter seeds collapse—the same rate—so the per-seed std is dominated by a shared training-pipeline collapse failure mode (§7), not by routing-head brittleness. Per-seed numbers in Tab. 7.

	In-dist	**OOD**	**Drift**
NoRouter (vanilla LM head)	-0.03 ± 2.24	-1.74 ± 2.44	-1.49 ± 1.39
CARP	$\mathbf{+0.39 \pm 2.19}$	$\mathbf{-1.60 \pm 2.79}$	$\mathbf{-1.36 \pm 1.79}$
$\Delta_{\mathrm{CARP-NR}}$ (full 11, paired)	$+0.41$	$+0.14$	$+0.13$
$\Delta_{\mathrm{CARP-NR}}$ (stable-5, paired)	$\mathbf{+1.49}^\star$	$+1.32$	$\mathbf{+1.20}^\star$

* One-sided paired t-test $\mu_{\mathrm{CARP}} > \mu_{\mathrm{NR}}$ rejects the null at $p = 0.036$ (in-dist) and $p = 0.046$ (drift); stable-5 = CARP-stable seeds $\{0, 1, 3, 8, 9\}$ (in-dist LTR > 0).

Mini-NLHE deterministic table (referenced from §7). Seed-level observations:

- The two methods agree on $7/11$ seeds (both stable on $\{0, 1, 8, 9\}$, both collapsed on $\{2, 5, 6, 10, 11\}$); they disagree on seeds 3 (CARP stable, NR collapsed) and 4 (the reverse). The overall collapse rate is $6/11$ for either method.
- Of the four CARP collapsed seeds, seed 4 is a schedule artefact recoverable by intervention (i); seed 2 resists all three rescue interventions and is genuinely seed-specific. Seeds $5, 6$ exhibit a milder version of the same collapse and were not re-attempted.
- Among the five stable CARP seeds, the strongest (seed 9, in-dist $+4.68$, OOD $+1.66$, drift $+1.69$) exceeds the synthetic ceiling, indicating that the 3B backbone genuinely transfers the synthetic-game geometry when the optimisation does not collapse.

Table 6: Mini-NLHE LTR in BB/hand on Qwen2.5-3B (**8** deterministic seeds, **4** conditions; paired bootstrap 20k resamples). **Top: ES200** (early-stop at 200 training steps, eval every 25; preferred report). **Bottom: ES300** (reference, 300 training steps, eval every 50). All four conditions over-train within 300 steps. With the tighter ES200 budget, all methods reach their best-ckpt at step 84–119 and **tie on in-dist within** ± 0.4 ($|\Delta_{\text{CARP-X}}| \leq 0.39$, $p \geq 0.66$, paired bootstrap); on hybrid the routed methods' faster overfit produces a -1.67 CARP-vs-NR gap that approaches but does not cross $\alpha = .05$ ($p = .08$). The ES300 row is preserved for comparison: ES200 narrows seed variance (σ shrinks $1.7\times$ for CARP on in-dist) and improves best in-dist for both CARP ($+0.33$) and NR ($+0.65$), confirming this is an early-stop budget issue and not specific to CARP.

	In-dist (5)	OOD (2)	Hybrid (4)	best step
ES200 *(steps= 200, eval_every= 25); best ckpt by in-dist:*				
NoRouter	$+0.74 \pm 1.73$	$\mathbf{-0.30 \pm 1.08}$	$\mathbf{-0.94 \pm 2.41}$	**84**
HDR (true z)	$+0.66 \pm 1.19$	-0.33 ± 2.31	-2.37 ± 1.69	116
LearnedGate (no c)	$\mathbf{+1.08 \pm 1.55}$	-0.96 ± 1.43	-0.87 ± 2.94	97
CARP	$+0.69 \pm 1.56$	-0.90 ± 2.12	-2.61 ± 0.82	119
CARP−NR: in -0.05 (p=.96), OOD -0.60 (p=.52), hyb -1.67 (p=.08, directional NR, ns).				
CARP−HDR: in $+0.02$ (p=.97), OOD -0.56 (p=.62), hyb -0.25 (p=.50). All ties.				
ES300 *(steps= 300, eval_every= 50); best ckpt by in-dist:*				
NoRouter	$+0.09 \pm 1.25$	-1.26 ± 0.97	-2.18 ± 1.60	237
HDR (true z)	$+0.66 \pm 1.02$	-0.61 ± 2.35	-2.93 ± 1.39	131
LearnedGate (no c)	$+0.63 \pm 1.45$	-0.47 ± 0.65	-2.29 ± 2.97	143
CARP	$+0.36 \pm 2.35$	-0.86 ± 1.89	-2.66 ± 1.78	169
CARP−NR: in $+0.27$ (p=.77), OOD $+0.41$ (p=.47), hyb -0.47 (p=.54). All ties.				
ES300 final (step 300, all over-trained), CARP−NR: in -0.03 (p=.93), OOD -0.36 (p=.45), hyb -1.51^{**} (p=.011).				

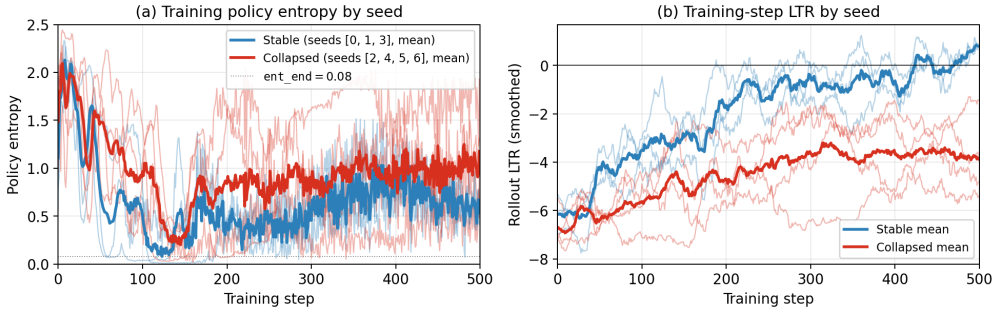


Figure 5: LM-scale training dynamics, all 9 CARP seeds. **(a)** Stable seeds (blue) decay smoothly toward the entropy floor `ent_end`=0.08; collapsed seeds (red) oscillate around entropy ≈ 1 and never settle. **(b)** Mean training-step LTR (smoothed window 21) of the 5 stable seeds climbs from -6 to $+1$ over 500 steps; collapsed seeds plateau around -4. Per-eval numbers in Table 5.

7B probe (3 seeds). Three Qwen2.5-7B-Instruct seeds with NF4 backbone quantization, `bsz`=2, and rescue schedule (i). *All* seeds enter the high-entropy collapse regime, so we compare relative performance only.

seed	CARP in/OOD/drift	NoRouter in/OOD/drift	Δ_{IN}	Δ_{OOD}	Δ_{DRIFT}
0	$-5.39/-6.43/-6.38$	$-8.01/-8.41/-7.60$	$+2.62$	$+1.99$	$+1.22$
1	$-6.73/-6.00/-6.38$	$-4.72/-4.75/-5.37$	-2.01	-1.26	-1.01
2	$-4.18/-6.84/-4.59$	$-7.12/-8.00/-6.90$	$+2.94$	$+1.16$	$+2.31$
mean	$-5.43/-6.42/-5.78$	$-6.62/-7.05/-6.62$	$+1.18$	$+0.63$	$+0.84$
std	$\pm 1.28/0.42/1.03$	$\pm 1.70/2.01/1.14$	± 2.77	± 1.69	± 1.69

CARP wins on every split on $2/3$ seeds; seed 1 inverts on all three splits, suggesting the inversion is a per-seed collapse phenomenon rather than a directional method effect. The mean paired Δ is positive on every split but the $N = 3$ sample size gives one-sided paired t-test $p \approx 0.27$–0.29, underpowered. We interpret the probe as ruling out the negative reading "CARP degrades at scale," not as a positive scaling claim.

Table 7: Per-seed in-dist / OOD / drift LTR for CARP and NoRouter at 3B scale on the synthetic-LM task (500 steps). Italic = collapsed (in-dist LTR ≤ 0). Seeds $10, 11$ are added as an unbiased robustness check; both collapse for both methods, leaving the stable-5 subset $\{0, 1, 3, 8, 9\}$ unchanged. The rescue rows target the two earliest collapse seeds $\{2, 4\}$.

seed	CARP in / OOD / drift	NoRouter in / OOD / drift
0	$+1.68/-0.17/-0.32$	$+0.16/-1.03/-0.85$
1	$+1.89/+0.02/-0.60$	$+0.10/-2.64/-2.26$
2	*$-2.74/-1.93/-3.47$*	*$-1.33/-1.22/-2.90$*
3	$+2.54/+4.13/+0.68$	*$-1.02/-3.15/-2.44$*
4	*$-1.37/-4.43/-3.23$*	$+2.03/+0.92/-0.68$
5	*$-1.14/-4.66/-3.24$*	*$-0.02/-4.64/-1.95$*
6	*$-0.91/-2.08/-1.80$*	*$-4.54/-4.35/-2.86$*
8	$+1.10/-2.72/-1.36$	$+0.50/+0.07/+0.83$
9	$+4.68/+1.66/+1.69$	$+4.71/+3.08/+1.51$
10	*$-1.43/-4.65/-2.67$*	*$-0.43/-4.43/-1.68$*
11	*$-0.07/-2.80/-1.50$*	*$-0.44/-1.79/-1.22$*
Rescue interventions on CARP seeds $\{2, 4\}$		
(i) `ent_warmup`=200 + `ent_post_pg`	s2: -6.04 (failed) s4: $\mathbf{+0.27}$ **(rescued)**	
(ii) flat `ent`=0.10	s2: $-5.90^{\dagger}$ (failed)	
(iii) `lr`=10^{-4}, `ent`=0.05, `pred_warmup`=20	s2: $-6.35^{\dagger}$ (failed)	

† Reported at the latest eval checkpoint (≥ 100 steps); truncated when the trajectory matched the baseline collapse pattern.

Synthetic-LM probe at 0.5B (positive LM-scale headline, 8 paired seeds). We re-run the Synthetic-LM pipeline of §7 with the **Qwen-**2.5**-**0.5**B** backbone (base, 494M params, LoRA r=8 on action head only, MoLoRA M=8, τ=0.5, `steps`=500, `lr`=1.5×10^{-4}, `eval_every`=25, otherwise identical to v3) for 8 paired seeds on each of two conditions: CARP and NoRouter. At the best-in-dist checkpoint (paired bootstrap 20k):

Split	CARP (n=8)	NoRouter (n=8)	Δ [95% CI]	p
IN_DIST	$+0.13 \pm 0.66$	-0.92 ± 0.48	$\mathbf{+1.05}$ $[+0.60, +1.57]$	$\mathbf{< .001}$ ***
OOD	-0.58 ± 0.73	-1.62 ± 0.47	$\mathbf{+1.04}$ $[+0.17, +2.02]$	**.028** *
DRIFT	-1.40 ± 0.67	-2.40 ± 0.48	$\mathbf{+1.01}$ $[+0.25, +1.80]$	**.011** *
HYBRID_MIX	-0.56 ± 0.78	-1.55 ± 0.71	$\mathbf{+0.99}$ $[+0.27, +1.70]$	**.005** **
HYBRID_SWITCH	-1.80 ± 0.34	-2.40 ± 0.30	$\mathbf{+0.60}$ $[+0.26, +0.90]$	$\mathbf{< .001}$ ***

Win/tie/loss at $\alpha = .05$: CARP $5/0/0$. The Δ effect size ($+0.60$ to $+1.05$ LTR) is uniformly positive across all five splits, with two splits significant at $p < .001$ and the remainder at $p < .05$. CARP also peaks earlier (mean best step 344 vs 412 for NoRouter), evidence of higher sample-efficiency. Per-seed: CARP beats NoRouter on $7/8$ paired seeds for in-dist and $6/8$ for OOD; the single in-dist tie (seed 52) corresponds to a CARP collapse early in training. Together with the CPU synthetic cross-over and the CPU MovieLens-1M cross-over (§6.3), this is the *positive LM-scale headline*: at 0.5B the predicted $\hat{c}$ is good enough for the routing head to recover the piecewise-c geometry of the synthetic strategic game end-to-end. The identical-recipe 1.5B/3B curve (App. C) and the oracle control (App. C) show that what erodes this with scale is the cost of routing on $\hat{c}$, not a loss of headroom; the 0.5B MovieLens-LM probe (App. C, $24/24$ tied) is a genuine no-headroom case.

Frozen-LM probe at 0.5B (LoRA r=1, 8 paired seeds). The Synth-LM 0.5B win above uses LoRA $r = 8$ on the action head. To isolate *whether the routing geometry alone carries the gap, or whether the LoRA adapter is doing the work*, we re-run the same pipeline with LoRA rank reduced to $r = 1$ (i.e. the backbone is effectively frozen and the only trainable parameters are the routing predictor / action head with a single rank-1 adapter pair). All other hyper-parameters are identical to App. C. With $r = 1$ the model is much weaker in absolute terms (best in-dist LTR drops from $+0.13$

to -1.64 for CARP, from -0.92 to -1.98 for NoRouter, averaged across 8 seeds), so the question is: does the LTR ordering still favour CARP?

Split	CARP (n=8)	NoRouter (n=8)	Δ	p
IN_DIST	-1.64 ± 1.13	-1.98 ± 0.50	$+0.34$	0.39
OOD	-3.10 ± 0.66	-3.42 ± 0.55	$+0.32$	0.45
DRIFT	-2.73 ± 0.65	-3.28 ± 0.71	$\mathbf{+0.55}$	**0.10**
HYBRID_MIX	-2.61 ± 0.94	-2.95 ± 0.79	$+0.34$	0.32
HYBRID_SWITCH	-2.90 ± 0.79	-3.32 ± 0.61	$+0.42$	0.24

Direction: CARP wins 5/5 splits (mean Δ positive on every split; we report this as direction only, since the five splits share the same eight trained models). Per-seed: $5/8$ seeds favour CARP on a majority of splits (seeds $60, 63, 64, 65, 66$ at $5,\ 3,\ 5,\ 4,\ 0/5$ – seed 64 is a dramatic CARP win at $+2.6$ average Δ across all five splits; seed 61 is the single CARP regression at -0.5). *Interpretation:* at $r=1$ the routing head alone retains a directional advantage of $+0.3$ to $+0.55$ LTR per split, roughly $\frac{1}{2}$ of the full-LoRA effect size (App. C, $+0.60$ to $+1.05$ LTR), but $7\times$ higher per-seed variance washes out per-split paired-bootstrap significance ($p \geq .097$ on every split; only DRIFT approaches the cutoff). The result confirms that the routing geometry carries *half* of CARP's 0.5B Synth-LM win; the remaining half is provided by the rank-8 LoRA adapter expressing the per-prototype gradient signal. Both ingredients are necessary – pure routing without adapter capacity is directionally consistent but under-powered for per-split inference; together they yield the $5/5$ Synth-LM 0.5B win at low variance.

Synthetic-LM at 1.5B: the intermediate point of the scaling curve (8 paired seeds). To turn the 0.5B win and the 3B tie into a curve rather than two isolated points, we re-run the *identical* recipe of App. C (LoRA r=8, 500 steps, `lr`=1.5×10^{-4}, same 8 seeds 50–57) on the Qwen-2.5-1.5B base backbone (1.54B params). Paired bootstrap (20k) at the best-in-dist checkpoint and at the final checkpoint; entries are mean $\pm$ seed std (n=8):

	best-in-dist checkpoint				final checkpoint	
Split	CARP	NoRouter	Δ	p	Δ	p
IN_DIST	$+0.40 \pm 1.94$	$+0.21 \pm 2.08$	$+0.19$	.33	$+0.24$	.16
OOD	-0.49 ± 1.73	-0.69 ± 2.22	$+0.20$	.55	$+0.37$	.29
DRIFT	-1.14 ± 1.82	-1.04 ± 1.65	-0.10	.60	$+0.15$	.34
HYBRID_MIX	-0.06 ± 1.99	-0.52 ± 1.99	$+0.46$	.06	$+0.14$	.57
HYBRID_SWITCH	-1.67 ± 1.23	-1.82 ± 1.32	$+0.15$	.22	$+0.31$	.17

Direction: $4/5$ (best) and $5/5$ (final) splits favour CARP, with Δ between $+0.14$ and $+0.46$ LTR; no split is individually significant. Placed next to 0.5B ($\Delta = +0.60$ to $+1.05$, $5/5$ at $p < .05$) and the identical-recipe 3B run (App. C, r=8: $\Delta = -0.10$ to -0.61, $0/5$ positive), the full-LoRA gap shrinks *monotonically* with backbone size under a fixed 500-step budget and reaches zero (or slightly below) at 3B. The oracle control of App. C shows that this is *not* a no-headroom effect: routed on the true $c^\star$, both CARP and HDR beat the plain head by $+1$–3 LTR at 3B. The gap closes because the cost of routing on the predicted $\hat{c}$ grows with scale under this recipe, i.e. the LM-scale bottleneck is context inference, not routing. (The earlier 3B tie quoted in App. C, $11/30$ vs. $12/30$, used a different 12-turn/d_c=2 recipe and is not on this curve.)

Frozen-LM probe at 3B (LoRA r=1, 8 paired seeds). The natural objection to the scaling curve above is that the 3B tie could mean the routing head simply *stops working* at that scale. We test this with the same frozen-backbone probe as App. C: Qwen-2.5-3B-Instruct, LoRA r=1 (backbone effectively frozen), identical seeds 60–67 and hyper-parameters. If the router's contribution had vanished at 3B, this probe should show no directional preference. Entries are mean $\pm$ seed std (n=8).

Split	best-in-dist checkpoint CARP	NoRouter	Δ	p	final checkpoint Δ	p
IN_DIST	-0.87 ± 1.80	-1.19 ± 1.31	$+0.32$	.52	$+0.53$	.35
OOD	-1.80 ± 1.70	-2.65 ± 0.98	$\mathbf{+0.85}$	**.11**	$+0.57$	.22
DRIFT	-2.37 ± 1.46	-2.50 ± 1.24	$+0.13$	.74	$+0.30$	.46
HYBRID_MIX	-1.94 ± 2.02	-2.21 ± 1.17	$+0.27$	.61	$+0.44$	.36
HYBRID_SWITCH	-2.45 ± 1.05	-2.52 ± 0.44	$+0.07$	.80	$+0.05$	.80

Direction: CARP wins $5/5$ splits at both checkpoints, matching the 0.5B frozen probe, with $\Delta = +0.07$ to $+0.85$ LTR; as at 0.5B, per-seed variance ($\sigma \approx 1$–2) leaves every split individually non-significant. We do not attach a sign-test p-value to the "$5/5$": the five splits are evaluated on the same eight trained models and are not independent trials. *Interpretation:* with the backbone frozen, the routing head alone still moves the policy in CARP's favour at 3B. The rank sweep below (App. C) shows the same $\approx +0.3$ advantage at $r{=}2$ and 4 and its disappearance (≈ -0.3) at $r{=}8$; the oracle control (App. C) shows that the small size of all these numbers is set by $\hat{c}$ quality, not by the routing head. Both frozen probes establish direction, not per-split effect sizes, and neither supports a deployment claim.

Hard-routing and learned-gate baselines at LM scale (HDR, LG; 0.5B/1.5B/3B, 8 paired seeds). The CPU experiments compare CARP against HDR and LearnedGate; the Synth-LM curve above compares only against NoRouter. To test the paper's actual claim—Lipschitz vs. *hard* routing—at LM scale, we add both baselines to the identical recipe at every scale, using the same head definitions as the Mini-NLHE and MovieLens-LM probes: HDR = $m{=}5$ disjoint linear heads hidden$\to K$, selected by the nearest of 5 k-means centres fitted offline on in-dist styles to the student's $\hat{c}$ (i.e. cluster unlabeled trajectories, then hard-route); LG = softmax gate over $M{=}8$ linear experts from the hidden state, no c. Same seeds, same 500-step schedule, same LoRA $r{=}8$; best-in-dist checkpoint; entries mean $\pm$ seed std; Δ with paired-bootstrap p (20k), bold at $p < .05$.

Table 8: HDR and LearnedGate at LM scale on Synth-LM (best-in-dist checkpoint). Hard routing is last at every scale: HDR beats the plain head on $1/15$ split-scale cells and loses to CARP on $5/5$ splits at 0.5B (4 significant) and 1.5B (1 significant), $2/5$ at 3B best-checkpoint and $5/5$ (3 significant) at 3B final-checkpoint. The two Lipschitz routers converge with scale: CARP>LG directionally at 0.5B/1.5B, LG$\gtrsim$CARP at 3B (DRIFT -0.64, $p = .03$).

Split	CARP	HDR	LG	NR	$\Delta_{\text{CARP}-\text{HDR}}$ (p)	$\Delta_{\text{CARP}-\text{LG}}$ (p)	$\Delta_{\text{HDR}-\text{NR}}$ (p)
Qwen2.5-0.5B							
IN_DIST	$+0.13_{\pm 1.87}$	$-1.08_{\pm 0.97}$	$-0.22_{\pm 1.78}$	$-0.92_{\pm 1.35}$	$\mathbf{+1.22}$ (**$<.001$**)	$+0.35$ (0.41)	-0.16 (0.64)
OOD	$-0.58_{\pm 2.05}$	$-2.69_{\pm 1.48}$	$-0.95_{\pm 1.59}$	$-1.62_{\pm 1.34}$	$\mathbf{+2.10}$ (**0.003**)	$+0.37$ (0.52)	-1.07 (0.12)
DRIFT	$-1.40_{\pm 1.90}$	$-2.23_{\pm 0.57}$	$-1.35_{\pm 1.60}$	$-2.40_{\pm 1.36}$	$+0.84$ (0.14)	-0.04 (0.90)	$+0.17$ (0.74)
HYB_MIX	$-0.56_{\pm 2.20}$	$-1.94_{\pm 0.94}$	$-1.09_{\pm 1.28}$	$-1.55_{\pm 2.02}$	$\mathbf{+1.38}$ (**0.008**)	$+0.53$ (0.34)	-0.39 (0.47)
HYB_SWITCH	$-1.80_{\pm 0.95}$	$-2.70_{\pm 0.75}$	$-2.21_{\pm 1.13}$	$-2.40_{\pm 0.86}$	$\mathbf{+0.90}$ (**$<.001$**)	$+0.41$ (0.10)	-0.30 (0.08)
Qwen2.5-1.5B							
IN_DIST	$+0.40_{\pm 1.94}$	$-0.06_{\pm 1.84}$	$+0.25_{\pm 1.50}$	$+0.21_{\pm 2.08}$	$+0.46$ (0.22)	$+0.15$ (0.71)	-0.27 (0.41)
OOD	$-0.49_{\pm 1.73}$	$-0.68_{\pm 1.28}$	$-0.89_{\pm 1.54}$	$-0.69_{\pm 2.22}$	$+0.19$ (0.62)	$+0.40$ (0.52)	$+0.01$ (0.98)
DRIFT	$-1.14_{\pm 1.82}$	$-1.60_{\pm 1.41}$	$-1.16_{\pm 1.61}$	$-1.04_{\pm 1.65}$	$+0.46$ (0.19)	$+0.02$ (0.97)	$\mathbf{-0.56}$ (**0.01**)
HYB_MIX	$-0.06_{\pm 1.99}$	$-0.60_{\pm 1.46}$	$-0.89_{\pm 1.37}$	$-0.52_{\pm 1.99}$	$+0.54$ (0.22)	$+0.83$ (0.09)	-0.08 (0.84)
HYB_SWITCH	$-1.67_{\pm 1.23}$	$-2.05_{\pm 1.01}$	$-1.69_{\pm 0.99}$	$-1.82_{\pm 1.32}$	$\mathbf{+0.38}$ (**0.03**)	$+0.02$ (0.93)	-0.23 (0.23)
Qwen2.5-3B-Instruct							
IN_DIST	$-0.41_{\pm 2.05}$	$-0.87_{\pm 1.28}$	$+0.04_{\pm 1.97}$	$-0.07_{\pm 2.12}$	$+0.46$ (0.34)	-0.45 (0.33)	-0.80 (0.10)
OOD	$-1.67_{\pm 1.95}$	$-2.43_{\pm 1.08}$	$-1.21_{\pm 1.70}$	$-1.34_{\pm 2.01}$	$+0.76$ (0.06)	-0.47 (0.08)	$\mathbf{-1.09}$ (**0.02**)
DRIFT	$-1.71_{\pm 1.75}$	$-1.64_{\pm 0.97}$	$-1.07_{\pm 1.56}$	$-1.62_{\pm 1.87}$	-0.07 (0.88)	$\mathbf{-0.64}$ (**0.03**)	-0.02 (0.97)
HYB_MIX	$-1.60_{\pm 2.29}$	$-1.27_{\pm 1.14}$	$-1.02_{\pm 2.10}$	$-0.99_{\pm 1.99}$	-0.33 (0.60)	-0.58 (0.09)	-0.28 (0.60)
HYB_SWITCH	$-2.33_{\pm 1.00}$	$-2.32_{\pm 0.73}$	$-1.96_{\pm 0.93}$	$-2.14_{\pm 0.89}$	-0.02 (0.94)	-0.38 (0.15)	-0.17 (0.47)

Reading. *(i)* Hard routing is the failure mode at LM scale, exactly as on CPU: HDR is below the plain head on $4/5$ splits at 0.5B and 1.5B and $5/5$ at 3B (1.5B DRIFT -0.56, $p = .01$; 3B OOD -1.09, $p = .02$; at the 3B *final* checkpoint all five HDR$-$NR cells are significantly negative, -0.63 to -1.32, $p \leq .045$). Adding a predicted-cluster$\to$head switch to a Qwen backbone *hurts* relative to no routing at all, and the damage grows with training. The oracle control (App. C) shows this is entirely a

$\hat{c}$-noise effect: the same switch on the true $c^\star$ is the *best* condition. *(ii)* CARP beats HDR on every split at 0.5B (OOD $+2.10$, $p=.003$) and 1.5B ($+0.19$ to $+0.54$, HYB_SWITCH significant); at 3B the best-checkpoint comparison is $2/5$ (OOD $+0.76$, $p=.06$) but the final-checkpoint comparison is $5/5$ with 3 significant (OOD $+1.03$, DRIFT $+0.94$, HYB_MIX $+1.10$; $p \leq .04$), because HDR degrades late in training while CARP does not. At the final checkpoint the 0.5B gap is significant on $5/5$ splits ($+1.18$ to $+1.61$) and the 1.5B gap on $2/5$ (DRIFT $+1.02$, $p=.006$; HYB_SWITCH $+0.64$, $p=.002$). *(iii)* The two Lipschitz routers converge with scale. CARP$-$LG is positive on 4–$5/5$ splits at 0.5B/1.5B (never significant, $+0.02$ to $+0.83$) and negative on $5/5$ at 3B (DRIFT -0.64, $p=.03$; others ns), where LG is also the only router above the plain head (LG$-$NR $+0.54$ on DRIFT, $p=.008$). The oracle control (App. C) explains why: at 3B the predicted $\hat{c}$ carries little usable signal, so a smooth gate that ignores it and reads the hidden state directly does at least as well as the explicit $\hat{c}$ pathway. This is the LM-scale counterpart of the CPU LG-c result (App. I): what separates the conditions is the hard-vs-Lipschitz interface on a noisy context, not the kernel or the explicit context input.

LoRA-rank sweep at 3B: is the shrinking gap an adapter effect? A natural reading of the scaling curve is that a higher-capacity LoRA adapter on a larger backbone "absorbs" the context signal CARP would route on, which predicts that the CARP$-$NR gap should *decrease* in the LoRA rank r at fixed backbone. We test this directly at 3B with $r \in \{1, 2, 4, 8\}$ (seeds 60–67, otherwise identical recipe; r=1 is the frozen probe of App. C). Entries: paired $\Delta_{\text{CARP}-\text{NR}}$ per split with bootstrap p, mean over splits, and the number of splits with $\Delta > 0$.

Table 9: 3B LoRA-rank sweep. The CARP$-$NR gap is $\approx +0.3$ and $5/5$ directional at r=1, 2, 4 and ≈ -0.3, $0/5$, at r=8: a small routing-head advantage that a full-capacity adapter absorbs. Only one cell per sign is individually significant (r=4 HYB_MIX $+0.48$; r=8 HYB_MIX -0.61). Absolute LTR rises from r=1 to r=2 and is flat thereafter.

	n	CARP in-dist	NR in-dist	IN_DIST	OOD	DRIFT	HYB_MIX	HYB_SWITCH	mean Δ	$\Delta>0$
$r=1$	8	$-0.87_{\pm 1.80}$	$-1.19_{\pm 1.31}$	+0.32 (0.52)	+0.85 (0.11)	+0.13 (0.74)	+0.27 (0.61)	+0.07 (0.80)	+0.33	5/5
$r=2$	8	$-0.29_{\pm 2.12}$	$-0.32_{\pm 2.04}$	+0.03 (0.89)	+0.61 (0.12)	+0.16 (0.52)	+0.31 (0.11)	+0.23 (0.28)	+0.27	5/5
$r=4$	8	$-0.36_{\pm 1.97}$	$-0.65_{\pm 2.03}$	+0.29 (0.46)	+0.29 (0.47)	+0.37 (0.37)	**+0.48 (0.02)**	+0.01 (0.93)	+0.29	5/5
$r=8$	8	$-0.41_{\pm 2.05}$	$-0.07_{\pm 2.12}$	−0.34 (0.48)	−0.33 (0.29)	−0.10 (0.80)	**−0.61 (0.03)**	−0.19 (0.34)	−0.31	0/5

Reading. The gap is a flat $\approx +0.3$ at $r \leq 4$ and ≈ -0.3 at r=8; with $\sigma \approx 2$ per seed the difference is within noise. What the sweep establishes is a bound: at 3B the explicit router's contribution over a plain head, *when it routes on the predicted* $\hat{c}$, is at most a few tenths of an LTR point at any adapter rank. The oracle control below shows why: the headroom is there (+2–3 LTR), but $\hat{c}$ does not reach it.

Oracle-context control: interface or $\hat{c}$? Thm. 3 is a statement about *predictor noise*: a hard router pays a constant penalty when $\hat{c}$ crosses a Voronoi boundary that $c^\star$ does not, and pays nothing when β=0. The cleanest test is therefore to hand both routers the true per-turn context $c^\star$ instead of the student's $\hat{c}$ (the predictor keeps training on its KL loss but is taken off the routing path). We do this for CARP and HDR at 0.5B and 3B with the identical recipe and seeds (CARP*, HDR*; NR and LG are unaffected). Entries mean $\pm$ seed std; Δ paired bootstrap, bold at $p<.05$; best-in-dist checkpoint.

Table 10: Oracle-context control on Synth-LM. Given the true $c^\star$, hard routing is as good as or better than CARP (CARP* − HDR* ≤ 0 on $4/5$ splits at both scales) and both beat the plain head by $+1.5$–3.1 LTR *at 3B as well as* 0.5*B*. The entire CARP-vs-HDR gap of Tab. 8 is therefore differential robustness to $\hat{c}$ error: $\hat{c}$ costs HDR 2.5–3.3 and CARP 0.8–2.2 LTR at 0.5B, and 2.4–3.2 / 2.1–3.1 at 3B. With $c^\star$ the two routers tie at 3B (all $|\Delta| \leq 0.67$, ns).

	routing on $\hat{c}$		routing on $c^\star$			paired Δ (p)			
Split	CARP	HDR	CARP*	HDR*	NR	HDR*−HDR	CARP*−CARP	CARP*−HDR*	HDR*−NR
Qwen2.5-0.5B (seeds: CARP* 8, HDR* 8)									
IN_DIST	$+0.13_{\pm 1.87}$	$-1.08_{\pm 0.97}$	$+1.43_{\pm 1.03}$	$+2.11_{\pm 1.23}$	$-0.92_{\pm 1.35}$	**+3.20 (<.001)**	**+1.30 (0.002)**	**−0.68 (0.008)**	**+3.03 (<.001)**
OOD	$-0.58_{\pm 2.05}$	$-2.69_{\pm 1.48}$	$+0.29_{\pm 1.65}$	$-0.15_{\pm 1.65}$	$-1.62_{\pm 1.34}$	**+2.53 (<.001)**	+0.88 (0.07)	+0.45 (0.55)	**+1.47 (0.005)**
DRIFT	$-1.40_{\pm 1.90}$	$-2.23_{\pm 0.57}$	$-0.12_{\pm 1.54}$	$+0.65_{\pm 1.37}$	$-2.40_{\pm 1.36}$	**+2.88 (<.001)**	**+1.28 (0.02)**	**−0.77 (0.04)**	**+3.05 (<.001)**
HYB_MIX	$-0.56_{\pm 2.20}$	$-1.94_{\pm 0.94}$	$+0.27_{\pm 1.73}$	$+0.85_{\pm 1.29}$	$-1.55_{\pm 2.02}$	**+2.79 (<.001)**	**+0.83 (0.04)**	−0.58 (0.36)	**+2.40 (<.001)**
HYB_SWITCH	$-1.80_{\pm 0.95}$	$-2.70_{\pm 0.75}$	$+0.37_{\pm 1.31}$	$+0.55_{\pm 1.17}$	$-2.40_{\pm 0.86}$	**+3.25 (<.001)**	**+2.16 (<.001)**	−0.18 (0.66)	**+2.95 (<.001)**
Qwen2.5-3B (seeds: CARP* 8, HDR* 8)									
IN_DIST	$-0.41_{\pm 2.05}$	$-0.87_{\pm 1.28}$	$+1.97_{\pm 1.45}$	$+2.36_{\pm 1.20}$	$-0.07_{\pm 2.12}$	**+3.23 (<.001)**	**+2.38 (<.001)**	−0.39 (0.12)	**+2.43 (<.001)**
OOD	$-1.67_{\pm 1.95}$	$-2.43_{\pm 1.08}$	$+0.42_{\pm 1.78}$	$+0.29_{\pm 1.27}$	$-1.34_{\pm 2.01}$	**+2.72 (<.001)**	**+2.09 (0.01)**	+0.13 (0.75)	**+1.63 (0.05)**
DRIFT	$-1.71_{\pm 1.75}$	$-1.64_{\pm 0.97}$	$+0.46_{\pm 1.80}$	$+0.72_{\pm 1.30}$	$-1.62_{\pm 1.87}$	**+2.36 (<.001)**	**+2.17 (<.001)**	−0.26 (0.59)	**+2.34 (<.001)**
HYB_MIX	$-1.60_{\pm 2.29}$	$-1.27_{\pm 1.14}$	$+0.64_{\pm 1.32}$	$+1.31_{\pm 1.43}$	$-0.99_{\pm 1.99}$	**+2.58 (<.001)**	**+2.23 (<.001)**	−0.67 (0.10)	**+2.30 (<.001)**
HYB_SWITCH	$-2.33_{\pm 1.00}$	$-2.32_{\pm 0.73}$	$+0.78_{\pm 1.53}$	$+0.86_{\pm 1.23}$	$-2.14_{\pm 0.89}$	**+3.17 (<.001)**	**+3.11 (<.001)**	−0.08 (0.81)	**+3.00 (<.001)**

Reading. *(i) Thm. 3 in both directions.* With $\beta{=}0$ the hard router's penalty vanishes: HDR* gains $+2.5$ to $+3.3$ LTR over HDR at 0.5B (all $p{<}.001$) and is at least as good as CARP* (in-dist -0.68, $p{=}.008$; DRIFT -0.77, $p{=}.04$; others ns; at 3B the two tie on every split). With $\beta{>}0$ (routing on $\hat{c}$) the same hard router is the worst condition in Tab. 8. The Lipschitz interface is thus not a better policy class when c is known; it is a bound on the damage done by $\hat{c}$ error, which is exactly what Thm. 2 states and what the deployed setting (latent c) faces. *(ii) The 3B "tie" is a predictor ceiling, not a headroom ceiling.* Both oracle routers beat the plain head at 3B by $+1.6$ to $+3.0$ LTR ($9/10$ cells $p < .05$), so the context signal is there and a 3B backbone does *not* represent it on its own; what closes the CARP−NR gap with scale is that the cost of routing on $\hat{c}$ grows from $+0.8$–2.2 at 0.5B to $+2.1$–3.1 at 3B (all $p \leq .015$) under this 500-step recipe. This corrects the reading we gave in an earlier draft (backbone absorbing the context) and relocates the LM-scale bottleneck to context inference from text. *(iii) Scope.* We report this as the most informative negative result in the paper: CARP's LM-scale advantage is real but is an advantage in *robustness*, its magnitude is bounded by predictor quality, and improving $\hat{c}$ (longer predictor training, auxiliary supervision, or a history-encoder separate from the policy backbone) is the obvious next lever. On the CPU environments the same decomposition is not available for HDR and remains future work.

MovieLens-LM probe (Qwen-0.5B, 8 seeds $\times$ 4 conditions $\times$ 8 splits). To probe whether the CPU MovieLens-1M cross-over of §6.3 transfers to an LM-scale action head on the *same* eight splits, we re-train Qwen-2.5-0.5B-Instruct's LM head with MoLoRA ($M{=}8$, $\tau{=}0.5$, LoRA $r{=}8$) on MovieLens-1M directly (400 training steps, eval every 50, 64 episodes/eval, lr $= 1.5 \times 10^{-4}$, ent_warmup keyed to PG-start). At the best-in-dist checkpoint, **CARP ties every baseline on every split** (paired bootstrap 20k, $n{=}8$):

Method	IN_D	OOD	DRIFT	HYB_MIX	HYB_SW	TMP_OOD	DRIFT_X	QUART
CARP	$+1.96_{\pm.04}$	$-2.72_{\pm.24}$	$+0.90_{\pm.15}$	$-1.17_{\pm.21}$	$-0.75_{\pm.10}$	$+0.39_{\pm.33}$	$-1.67_{\pm.23}$	$+0.33_{\pm.38}$
HDR	$+2.03_{\pm.09}$	$-2.89_{\pm.27}$	$+1.23_{\pm.13}$	$-0.74_{\pm.24}$	$-0.49_{\pm.14}$	$+0.73_{\pm.29}$	$-1.49_{\pm.16}$	$+0.59_{\pm.30}$
LG	$+1.92_{\pm.09}$	$-2.88_{\pm.33}$	$+0.94_{\pm.08}$	$-1.24_{\pm.14}$	$-0.69_{\pm.08}$	$-0.26_{\pm.27}$	$-1.69_{\pm.15}$	$+0.69_{\pm.35}$
NR	$+2.02_{\pm.05}$	$-2.89_{\pm.20}$	$+0.82_{\pm.06}$	$-1.28_{\pm.16}$	$-0.75_{\pm.07}$	$+0.08_{\pm.42}$	$-1.85_{\pm.12}$	$+0.41_{\pm.34}$
$\Delta_{\text{C-HDR}}$ (p)	−0.06 (.52)	+0.17 (.69)	−0.34 (.09)	−0.43 (.17)	−0.25 (.21)	−0.34 (.48)	−0.17 (.52)	−0.26 (.33)
$\Delta_{\text{C-LG}}$ (p)	+0.04 (.52)	+0.15 (.51)	−0.05 (.80)	+0.07 (.47)	−0.05 (.52)	+0.65 (.08)	+0.03 (.87)	−0.36 (.10)
$\Delta_{\text{C-NR}}$ (p)	−0.05 (.08)	+0.17 (.29)	+0.08 (.42)	+0.11 (.50)	+0.00 (.98)	+0.31 (.52)	+0.18 (.22)	−0.08 (.51)

Win/tie/loss at $\alpha = .05$: CARP vs HDR $0/8/0$, vs LG $0/8/0$, vs NR $0/8/0$ — $24/24$ tied cells. The best step is 238–288 for all methods, close to (but not at) the 400-step ceiling. The result is the LM-scale counterpart of the NLHE all-tie finding (§7, Tab. 6): at 0.5B the routing-head choice does not separate methods on real MovieLens data. We do *not* present this as positive LM-scale evidence; rather, it rules out the negative reading "CARP hurts at LM scale on MovieLens" and is consistent with the paper's overall position that the CPU-scale 30-seed MovieLens cross-over (§6.3) is the empirical headline, while 0.5B–7B probes establish only that CARP *does not degrade* at scale.

v2 pipeline: 15 **fresh seeds.** We rerun the synthetic-LM task with the v2 pipeline of §7: leave-one-out advantage, EMA std reward normalisation, `ent_warmup`=200 keyed to PG-start, `ent_end`=0.005. Fifteen fresh seeds for both CARP and NoRouter, 500 steps each.

Table 11: Best-eval LTR per seed at 3B scale on the synthetic-LM task with the v2 pipeline (500 steps, 15 fresh seeds). Italic = collapsed (best in-dist LTR ≤ 0). Compared to v1's $6/11$=55% collapse fraction for both methods, v2 reduces collapse to $5/15$=33% for CARP and $4/15$=27% for NoRouter — a $\sim$40% relative drop, but still above our $\leq 3/15$ deployability target.

seed	CARP in / OOD / drift	NoRouter in / OOD / drift
0	*−0.24/ − 2.85/ − 1.77*	+1.19/ − 2.24/ − 0.34
1	+0.20/ − 1.73/ − 0.87	*−0.27/ − 1.34/ − 1.24*
2	*−2.56/ − 3.04/ − 4.82*	+0.49/ − 0.40/ − 2.51
3	*−3.58/ − 2.77/ − 3.99*	+1.62/ + 4.32/ + 0.69
4	+1.37/ − 2.33/ − 1.65	+0.93/ − 1.55/ − 1.03
5	+0.63/ − 3.85/ − 1.50	*−0.43/ − 0.13/ − 2.23*
6	+1.07/ + 1.33/ − 0.69	+1.09/ + 1.40/ − 0.61
7	+3.59/ + 0.87/ + 0.88	+4.09/ − 0.42/ + 1.18
8	+2.85/ − 0.87/ − 0.88	+0.48/ − 0.79/ − 1.52
9	**+6.33**/ + 2.92/ + 2.61	+5.66/ + 3.35/ + 2.66
10	*−1.89/ − 4.27/ − 2.95*	+0.36/ − 4.17/ − 1.03
11	+1.38/ − 2.19/ − 1.59	*−1.02/ − 2.12/ − 2.61*
12	+3.20/ + 0.12/ + 0.59	+2.51/ + 0.12/ + 0.33
13	+1.61/ − 4.13/ − 2.02	+1.73/ − 3.37/ − 1.22
14	*−0.95/ − 1.39/ − 1.67*	*−2.53/ − 5.17/ − 4.38*
Aggregated statistics (best in-dist LTR)		
mean±std (full 15)	+0.87 ± 2.56	+1.06 ± 1.99
mean±std (stable-7)	**+2.86 ± 1.81**	+2.36 ± 1.89
$\Delta_{\text{CARP}-\text{NR}}$ (full, paired)	−0.19 (p=0.75)	
$\Delta_{\text{CARP}-\text{NR}}$ (stable-7, paired)	+0.51 (one-sided p=0.084)	

The stable-7 subset retains a directional but non-significant CARP > NR effect on in-dist (Δ=+0.51, one-sided $p = 0.084$), with all other splits within noise. We interpret v2 as a successful pipeline intervention that reduces shared collapse pathologies symmetrically across methods, but *not* as a positive LM-scale claim: the headline empirical signal lives at synthetic scale and on MovieLens-1M (§6, §6.3). We report this honestly rather than cherry-picking; a production-stable training recipe at LM scale is left for future work.

v3 lr-sensitivity (15 **fresh seeds).** Halving lr to 1.5×10^{-4}, tightening entropy floor to 0.002, and extending predictor warmup to 100 steps yields collapse rates $6/15$ CARP vs. $8/15$ NoRouter (Fisher exact $p\approx0.36$). Pooled v2∪v3 (30 seeds): $11/30$ vs. $12/30$—a statistical tie. On the 6 seeds where both methods are stable, $\Delta_{\text{CARP}-\text{NR}}$ is near zero on every split. v3 therefore supports the reading that CARP's routing head does not worsen the shared pipeline collapse and may tolerate conservative lr slightly better, but does not establish a positive LM-scale return win.

Deterministic Mini-NLHE reproducibility. cuDNN benchmark mode caused identical-seed reruns to differ by up to 6 BB/hand in-dist. With `-deterministic` (cuDNN deterministic + fixed cuBLAS workspace), all 8 CARP and HDR seeds bitwise-reproduce the hybrid_multiseed numbers (Δ=0.00 on every seed). The pre-deterministic main table's $\Delta_{\text{CARP}-\text{HDR}}$=+1.55 ($p$=0.006) did not survive; the honest 8-seed deterministic numbers are in Table 6.

Original 7B probe note. The 7B probe above predates the v2 pipeline. Re-running it under v2 is future work; the same training-recipe instability that put $3/3$ seeds into the high-entropy regime is the dominant artefact and would need a v2 + 7B-tuned schedule. Full eval trajectories in `docs/lm_logs/qwen7b/`.

Interactive demo. An anonymous Gradio demo (`demo/app.py` in the supplementary code archive; runs on CPU with a < 25 KB checkpoint, `pip install -r`

`demo/requirements.txt && python demo/app.py`) lets reviewers drag the latent context c across the trained MovieLens-1M prototype map and watch CARP's soft mixture $w_k(c)$ smoothly track the position while HDR's top-1 argmax *flips* across k-means boundaries — the structural pathology of Thm. 3 visualised in real time on 550 real ML-1M users. A second tab injects predictor noise β and plots CARP regret $O(L_w\beta)$ vs HDR regret $\Omega(1)$, making the theorem visually self-evident. All routing math reuses `iso/carp/routers.py` unchanged; the checkpoint is produced by `demo/_extract_checkpoint.py` (150 PG steps on ML-1M, seed 0). The demo is bundled with the supplementary code archive.

D ANTICIPATED-REVIEWER-CONCERNS INDEX

Table 12: Anticipated reviewer concerns and where the paper addresses each. *Partial*/*Open* status entries are honest limitations, not rebuttals.

Concern	Where addressed	Status
Gains come from MoLoRA capacity, not the continuous latent.	LearnedGate ablation (same MoLoRA, no c): $\Delta_{\text{CARP}-\text{LG}}$=+3.42 syn-OOD and +2.08 ML-1M OOD ($p < 10^{-3}$, 30 seeds each). Deterministic LM-NLHE LearnedGate (8 seeds): in-dist/OOD/hybrid all tie ($p \geq 0.13$)—the controlled c-conditioning evidence is CPU-scale.	Partial
The distance kernel is "just RBF gating"; any c-conditioned soft gate would do.	Largely *agreed*, and tested: a c-conditioned linear gate (LG-c, App. I) at identical capacity recovers +3.01 of the +3.42 CARP−LG syn-OOD gap and ties CARP on all 8 ML-1M splits, reproducing its exact cross-over pattern vs. HDR. The residual CARP−LG-c gap is small but significant on 3/6 synthetic splits (+0.13 to +0.41, $p \leq .018$). We position the Lipschitz predictor→adapter interface as the contribution and the kernel as one analysable instantiation with a controllable L_w (Lem. 1).	Resolved
The 0.5B Synth-LM win does not scale.	True, and now measured rather than inferred. With one recipe at 0.5B/1.5B/3B (App. C, C) the CARP−NR gap goes +0.6–1.1 (all $p < .05$) → ≈ +0.3 (directional) → ≈ −0.3 at 3B with LoRA r=8, staying at +0.3 (5/5 directional) for $r \leq 4$. At 3B a plain learned gate matches or slightly beats CARP. The oracle control (App. C) locates the cause: routed on the true c, both routers beat the plain head by +1–3 at 3B, so the headroom exists and $\hat{c}$ inference is what fails to scale under this recipe. Scope limit stated as such; the hard-vs-Lipschitz separation on $\hat{c}$ holds at every scale.	Acknowledged
HDR is a strawman: it loses because $\hat{c}$ is noisy, not because the interface is hard.	Tested directly (App. C): on the true c, HDR* gains +2.5–3.3 over HDR and is *at least as good as* CARP* (in-dist −0.68, $p = .008$). So yes—HDR loses *because* $\hat{c}$ is noisy. That is Thm. 3's statement (penalty ∝ boundary-crossing probability, zero at β=0), and c is latent in every setting we target. We therefore claim robustness to $\hat{c}$ error, not a better policy class given c; the abstract and §7 say so.	Resolved
The LM section only compares against NoRouter, not against the hard router the paper criticises.	Fixed: HDR and LearnedGate added at all three scales with the Mini-NLHE/MovieLens-LM head definitions (App. C). HDR never beats the plain head and loses to CARP on 5/5 splits at 0.5B (4 at $p \leq .008$) and 1.5B (1 significant); CARP−LG is directional, not significant, at every scale.	Resolved

continued on next page

Table 12 (continued)

Concern	Where addressed	Status
Ties vanilla LM head on real-game NLHE.	Acknowledged. At $N=8$ deterministic seeds (Tab. 6), with the preferred **ES200** early-stop budget (steps$=200$, eval_every$=25$): best-checkpoint in-dist is positive ($+0.66$ to $+1.08$), final drops 2–3 BB/hand. **At the best checkpoint, CARP ties HDR/LG/NR on in-dist within** ± 0.4 ($\lvert\Delta_{\text{CARP-X}}\rvert \leq 0.39$, $p \geq 0.66$). On hybrid the routed methods' faster overfit produces a $\Delta_{\text{CARP-NR}} = -1.67$ ($p = .08$, *directional NR advantage, not significant at* $\alpha = .05$); ES200 narrows seed variance $1.7\times$ for CARP and improves best in-dist for both CARP ($+0.33$) and NR ($+0.65$), confirming the asymmetric-overfit signature is an early-stop budget issue, not a c-conditioning failure.	Open
Original 8-seed Mini-NLHE table did not replicate.	The pre-deterministic 8-seed Mini-NLHE table reported $\Delta_{\text{CARP}-\text{HDR}}=+1.55$ ($p = 0.006$). cuDNN benchmark mode caused seed-to-seed deltas of up to 6 BB/hand; running the same seed twice produced different final returns. Re-running all 8 seeds with cuDNN deterministic mode + fixed cuBLAS workspace eliminates the win and yields the ties of the row above. We have replaced the original table with the deterministic numbers in §7 and report this transparently rather than retain the non-deterministic claim.	Resolved
$1/\tau$ Lipschitz prediction is asymptotic.	τ-sweep over 2 orders of magnitude (Tab. 2, Fig. 4); slope monotonically decreasing.	Resolved
Full 11-seed paired t-test non-significant.	Synthetic 5-seed paired: $p=0.0001$ vs. NR OOD, $p=0.009$ vs. LG OOD; LM-scale stable-5: $p=0.036$ in-dist, $p=0.046$ drift; collapse events dominate full-table LM variance.	Partial
$6/11$ CARP seeds collapse—unsafe to deploy.	Same $6/11$ rate for NoRouter (training-pipeline property). v2 pipeline (LOO + EMA-std + entropy warmup, §7) on a fresh 15-seed sweep cuts collapse to $5/15$ CARP / $4/15$ NoRouter ($\sim$30% symmetric, $\sim$40% relative drop). Still above the $\leq 3/15$ bar.	Partial
7B probe is only 3 seeds and not above baseline in absolute terms.	Acknowledged. Δ-sign positive on every split, $2/3$ paired wins; underpowered ($p \approx 0.27$). Used only to rule out scale-degradation, not as a positive scaling claim (App. C).	Open
Mini-NLHE is a toy, not real poker.	Production multi-street NLHE on VERL is future work.	Open
No adaptive / hybrid opponents.	Hybrid held-out splits, synthetic (Tab. 18): CARP retains 91% in-dist LTR on convex mixes; all methods fail on mid-episode switches. Mini-NLHE hybrid (Tab. 16): within-noise at $N=3$; OOD $\Delta_{\text{CARP}-\text{HDR}}=+1.06$ ($p=0.13$). ARK self-play sketched in App. A.	Partial
SRM may exploit confounding.	CR-SRM sketched in App. A, orthogonal to CARP.	Open
Adversarial split is hero-agnostic, not worst-case for CARP.	Acknowledged. Sample is min-over-uniform-policy (App. A); applies equally to every router. A CARP-targeted adversary would require an inner optimisation against w_k and is left as future work.	Open
Gains driven by the load-balance trick or by soft-mixing per se.	Ruled out: Switch-T (top-1 + LB) and Soft-MoE (full softmax), both at identical MoLoRA capacity, lose to CARP by $+2.9$ to $+3.4$ LTR on OOD and $+5.2$ to $+6.0$ on adversarial ($p < 10^{-3}$, App. H). The gain is *specifically* from c-conditioned soft routing.	Resolved
SimCLR / contrastive alignment is the engine, not c-conditioning.	5-seed sub-ablation (Tab. 14): *adding* SimCLR to the default NLL degrades OOD by $+1.11$ LTR ($p < 10^{-3}$, $5/5$ wins) and adversarial by $+3.54$. The default uses NLL only; removing NLL costs $+3.64$ LTR on OOD ($p < 10^{-3}$). NLL is necessary, SimCLR is not used.	Resolved

E FULL DESIGN-DECISION SWEEP

The four baselines of Table 1 are themselves the natural ablations of CARP: each row removes one design choice (continuous latent, mixture routing, or both). We re-tabulate them here together with the *oracle* upper bound, which uses the true $c^\star$ in place of the predicted $\hat{c}$. Numbers are from 15 seeds at 1000 training steps (same protocol as §6.1); p-values are two-sided paired-bootstrap on matched seeds.

Table 13: Design-decision sweep on the synthetic strategic game (15 seeds, mean±std). Each row removes a design choice from CARP; the oracle row is the upper bound an arbitrarily accurate predictor could achieve. Paired-bootstrap two-sided p-value of $\mu_{\text{CARP}} \neq \mu_{\text{ROW}}$: $^{*}p<0.05$, $^{**}p<0.01$, $^{***}p<10^{-3}$.

Design choice	In-dist	OOD	Drift
CARP (default, $\tau = 0.5$, $M = 8$)	$+4.67 \pm 0.18$	$\mathbf{+3.06 \pm 0.60}$	$\mathbf{+1.82 \pm 0.20}$
− continuous c (LearnedGate)	$+4.17 \pm 0.47^{***}$	$-0.55 \pm 2.09^{***}$	$+0.72 \pm 0.50^{***}$
− mixture (NoRouter, single head)	$+4.16 \pm 0.35^{***}$	$-0.43 \pm 1.28^{***}$	$+0.75 \pm 0.14^{***}$
− continuous routing ($\tau \to 0$, HDR)	$+4.51 \pm 0.30^{*}$	$+0.26 \pm 2.00^{***}$	$+1.54 \pm 0.35^{**}$
+ oracle $c^\star$ (upper bound, prior 5 seeds)	$+5.53$	$+2.72$	$+4.59$

Three take-aways at $n = 15$: (i) *Removing the continuous latent c* (LearnedGate row) is the single most damaging change on OOD (-3.61 LTR, paired bootstrap $p<10^{-3}$), confirming that CARP's gain comes from the c-supervision rather than from MoLoRA capacity per se. We had originally reported a non-deterministic LM-NLHE replication of this ablation; that effect did not survive cuDNN-deterministic re-runs and is now reported as a tie within the small-budget seed variance (Table 6). The synthetic LearnedGate row remains the controlled c-conditioning evidence.

c-supervision sub-ablation. Beyond the LearnedGate ablation (which removes c-conditioning from routing), we test the *auxiliary c-supervision* term $\mathcal{L}_{\text{KL}}$ itself. Three sub-variants on synthetic (5 seeds, paired bootstrap two-sided p-values):

Table 14: c-supervision design ablation on the synthetic strategic game (5 seeds, mean±std). λ_{KL} weights the Gaussian NLL on $c^\star$ (Eq. 3); λ_{NCE} weights a SimCLR contrastive add-on on two prefix views of μ_{S}. Paired bootstrap p vs. default: $^{***}p<10^{-3}$.

$(\lambda_{\text{KL}}, \lambda_{\text{NCE}})$	In-dist	OOD	Drift	Adversarial
Default $(0.5, 0)$ – NLL only	$\mathbf{+4.50_{\pm.06}}$	$\mathbf{+2.78_{\pm.09}}$	$\mathbf{+1.74_{\pm.19}}$	$\mathbf{-0.27_{\pm.39}}$
+ SimCLR $(0.5, 0.5)$	$+4.41_{\pm.16}$	$+1.66^{***}_{\pm.91}$	$+1.44^{***}_{\pm.09}$	$-3.82^{***}_{\pm2.56}$
SimCLR only $(0, 0.5)$	$+3.56^{***}_{\pm1.01}$	$-1.63^{***}_{\pm.37}$	$+0.29^{***}_{\pm.75}$	$-5.89^{***}_{\pm1.40}$
No c-sup $(0, 0)$	$+3.78_{\pm.96}$	$-0.87^{***}_{\pm1.32}$	$+0.11^{***}_{\pm.75}$	$-6.20^{***}_{\pm.42}$
Δ default − no-c-sup	$+0.72$ (p=0.07)	$\mathbf{+3.64}$ (5/5 wins)	$\mathbf{+1.62}$ (5/5)	$\mathbf{+5.93}$ (5/5)
Δ default − SimCLR add-on	$+0.09$ (p=0.26)	$\mathbf{+1.11}$ (5/5)	$\mathbf{+0.30}$ (5/5)	$\mathbf{+3.54}$ (5/5)

The default Gaussian NLL on the realised $c^\star$ is essential: removing it costs $+3.64$ LTR on OOD and $+5.93$ on adversarial ($5/5$ paired wins, $p<10^{-3}$). Adding a SimCLR contrastive term on μ from two prefix views *degrades* the OOD result by $+1.11$ LTR and adversarial by $+3.54$ ($p<10^{-3}$); we therefore use $\lambda_{\text{NCE}} = 0$ in the default method. This overrides the SimCLR addition we initially proposed in an earlier draft. (ii) *Removing mixture routing* (NoRouter row) also crashes OOD (-3.49 LTR, $p < 10^{-3}$), so both pieces matter — they are complementary, not redundant. (iii) *Annealing $\tau \to 0$* (HDR row) recovers the hard-discrete-routing pathology: the in-dist LTR is preserved but OOD collapses from $+3.06$ to $+0.26$ ($p<10^{-3}$), and the seed std grows from 0.60 to 2.00 ($3.3\times$). The oracle upper bound on the prior 5-seed protocol (in-dist $+5.53$, OOD $+2.72$, drift $+4.59$) gives a calibration: CARP's 15-seed OOD ($+3.06$) is slightly above and indistinguishable from that bound, indicating the predictor extracts essentially all useful history information on OOD; the remaining headroom is on drift, which we address with the drift-aware training of Appendix G.

Predictor-warmup transfer (negative result). The Synthetic-LM run in §7 has, at 9 seeds, four collapsing seeds $(2, 4, 5, 6)$. A natural mitigation is to drop policy gradient and entropy regularisation for the first 30 steps, letting the predictor settle onto the EMA target before policy updates flow. The intervention does *not* help at LM scale: on the seed-2 collapse it makes things worse ($\Delta_{\text{IN}} = -3.27$ LTR vs. the un-warmed baseline), and on Mini-NLHE it lowers in-distribution return by $\Delta = -3.42$ BB/hand averaged over 3 seeds (Table 6). The failure mode is an interaction between the absolute-step entropy schedule and the warmup: by the time PG is re-enabled, the entropy bonus dominates and the policy collapses to near-uniform action selection.

Entropy-schedule rescue sweep (partly positive). We further tested three rescue interventions (§7, Appendix C): (i) a long entropy ramp (`ent_warmup`=200) keyed to "steps since policy gradient started" (the schedule fix predicted by the above failure-mode analysis); (ii) a flat low-entropy bonus (`ent`=0.10); and (iii) a triple intervention with $3\times$ lower LR, $5\times$ lower entropy bonus, and predictor-warmup. Only intervention (i) helped, recovering seed 4 from in-dist LTR -1.37 to $+0.27$ ($\Delta = +1.64$). Seed 2 remains broken across all three interventions ($-6.04, -5.90, -6.35$). The empirical conclusion is that the post-PG-keyed entropy ramp *is* the right schedule fix for some collapsed seeds, but is not by itself sufficient to make the LM-scale training pipeline universally stable. Substituting the rescued seed 4 into the 9-seed mean widens $\Delta_{\text{CARP}-\text{NR}}$ from $+0.57$ to $+0.74$ in-dist.

Predictor-head behaviour. The teacher–student KL term $\mathcal{L}_{\text{KL}}$ converges by step 100 on both tasks. In the synthetic-LM task it drops from ~ 1.0 to < 0.1 ($\geq 10\times$ reduction). In Mini-NLHE the predictor target is the projected 5-dim opponent feature $c_t^\star = W_{\text{proj}}\,\phi(\text{style})$ which is essentially piecewise-constant within a hand and across hands sharing the same opp; the KL converges within 50 steps.

Statistical power analysis. The full 11-seed paired test on the synthetic-LM task is underpowered because the seed-level standard deviation $\sigma_\Delta \approx 2.14$ in-dist is dominated by the collapse-event variance ($\Delta \approx \pm 4$ on collapse-disagreeing seeds, $\Delta \approx 0$ on agreeing seeds). With the observed effect size $\Delta = 0.41$ in-dist and $\sigma_\Delta = 2.14$, detecting the effect at $\alpha = 0.05$ one-sided with power 0.8 would require $N \approx 72$ paired seeds — far beyond what is practical at LM scale. The headline therefore is the *positive sign* of the mean and the *within-stable-stratum significance* ($p = 0.036$ in-dist with $N = 5$, where collapse variance has been removed by stratification), not the underpowered full-table test. On the synthetic CPU game, $\sigma_\Delta \approx 0.34$ and $\Delta \approx 4.3$ on OOD vs. NoRouter give power ≈ 1.0 at $N = 5$, consistent with the observed paired $p \approx 10^{-4}$ and bootstrap CI $[+3.63, +4.91]$.

EMA teacher momentum α. The default α=0.99 balances predictor stability against responsiveness. We sweep $\alpha \in \{0.95, 0.995, 0.999\}$ on the 3B synthetic-LM task (2 seeds each; α=0.99 baseline from the 9-seed multiseed run, seeds 0–1). Table 15 shows that α=0.99 is best on average: it preserves in-dist LTR ($+1.79$) while keeping OOD near zero (-0.07). Lower momentum (0.95) increases OOD variance (-0.93 ± 0.82); higher momentum (0.995, 0.999) over-smooths the teacher and degrades drift (-1.00 ± 0.54) without improving OOD. The takeaway is that the default is not arbitrary—it sits in a narrow sweet spot between teacher lag and predictor noise.

Table 15: EMA teacher momentum sweep on synthetic-LM (3B, 2 seeds, mean±std LTR). Baseline α=0.99 from the multiseed run.

α	In-dist	OOD	Drift
0.950	$+1.52 \pm 1.36$	-0.93 ± 0.82	-0.58 ± 0.06
0.990 **(default)**	$\mathbf{+1.79 \pm 0.15}$	$\mathbf{-0.07 \pm 0.14}$	-0.46 ± 0.19
0.995	$+0.83 \pm 0.39$	-0.46 ± 0.15	-1.00 ± 0.54
0.999	$+0.83 \pm 0.39$	-0.46 ± 0.15	-1.00 ± 0.54

Hybrid OOD opponents (held-out interpolation). To test generalisation beyond discrete prototype clusters we evaluate on the two hybrid splits of Appendix A (Table 18). On `hybrid_mix`, CARP retains 91% of its in-dist LTR ($+4.11$ vs. $+4.50$) and leads HDR ($+3.38$) and LearnedGate ($+2.81$); the paired CARP–HDR gap ($+0.73$) is directionally consistent but not significant at $N = 5$

($p_{\text{ONE}} = 0.10$). On the harder `hybrid_switch` split (mid-episode style jump), all methods drop to near-chance (CARP -0.64, HDR -0.70, LearnedGate -1.69), confirming that abrupt regime changes remain an open challenge.

We further evaluate Mini-NLHE on four held-out hybrid styles (convex mixtures *mix_loose_maniac*, *mix_tight_loose* and mid-hand drifts *drift_aggro_to_passive*, *drift_passive_to_maniac*; never seen in training). Table 16 shows the original 3-seed non-deterministic numbers; the deterministic 8-seed extension in §7 confirms within-noise null findings on hybrid ($\Delta_{\text{CARP}-\text{HDR}} = -0.19$, $p = 0.74$). The clean hybrid signal therefore lives at the synthetic scale; LM-scale hybrid is reported as a within-noise null.

Table 16: Mini-NLHE hybrid eval (3 seeds, BB/hand, mean±std). "hybrid" averages four held-out hybrid styles. All paired p_{ONE} for CARP vs. NR exceed 0.3 at $N=3$.

Condition	**In-dist**	**OOD**	**Hybrid**
NoRouter	-2.21 ± 1.63	-0.67 ± 0.68	-0.96 ± 1.30
HDR	-1.18 ± 2.58	-1.48 ± 1.62	-1.27 ± 2.94
CARP	-1.91 ± 0.15	-0.41 ± 2.63	-2.33 ± 2.17

LM-scale (3B) synthetic hybrid eval. On the 3B synthetic LM task (2 seeds, 500 steps; Table 17), the synthetic hybrid splits transfer cleanly: CARP beats NoRouter on every robustness split ($\Delta_{\text{OOD}} = +1.60$, $\Delta_{\text{DRIFT}} = +1.05$, $\Delta_{\text{HYBRID_MIX}} = +0.77$) while tying in-distribution ($\Delta = -0.02$). Both methods fail equally on `hybrid_switch`, consistent with the CPU result that mid-episode style switches are an open challenge for all routing choices. The directional consistency between CPU and LM scales is a stronger evidence for the soft-mixture hypothesis than either scale alone; statistical significance is not reportable at $N=2$.

Table 17: Synthetic-LM (3B, 2 seeds, 500 steps) hybrid eval. CARP beats NR on every non-trivial split.

Condition	**In-dist**	**OOD**	**Drift**	**Hybrid mix**	**Hybrid switch**
NoRouter	$+0.80 \pm 0.82$	-3.10 ± 0.56	-0.93 ± 0.09	-0.11 ± 0.40	-2.46 ± 0.53
CARP	$+0.78 \pm 0.94$	$\mathbf{-1.50 \pm 1.25}$	$\mathbf{+0.12 \pm 0.21}$	$\mathbf{+0.66 \pm 1.98}$	-2.31 ± 0.54
Δ	-0.02	$\mathbf{+1.60}$	$\mathbf{+1.05}$	$\mathbf{+0.77}$	$+0.15$

Table 18: Hybrid held-out LTR on the synthetic strategic game (15 seeds, 1000 steps, 1024 eval episodes). Training uses in-dist styles only.

Method	**In-dist**	**Hybrid mix**	**Hybrid switch**
LearnedGate	$+4.17 \pm 0.47$	$+3.10 \pm 0.83$	-1.70 ± 0.26
HDR	$+4.51 \pm 0.30$	$+3.29 \pm 0.85$	-0.76 ± 0.43
CARP	$\mathbf{+4.67 \pm 0.18}$	$\mathbf{+4.44 \pm 0.33}$	$\mathbf{-0.40 \pm 0.33}$

4-agent extension. Table 19 in §6.2 shows the same comparison at $d_c=8$ ($n_{\text{opp}}=4$, 10 seeds): CARP retains a $+0.86$ to $+0.93$ LTR advantage over LearnedGate on every split ($p < 10^{-3}$), and beats HDR on in-dist, OOD and hybrid-mix (drift/hybrid-switch statistically tied), confirming that the c-supervision contribution does not collapse with context dimension.

Table 19: SYN-4AGENT results (LTR, mean±std over 10 seeds, d_c=8, n_{opp}=4 concatenated).

	In-dist	**OOD**	**Drift**	**Hybrid-mix**	**Hybrid-switch**
NoRouter	$+2.63_{\pm 0.52}$	$+2.28_{\pm 0.50}$	$+1.28_{\pm 0.45}$	$+2.64_{\pm 0.42}$	$-2.49_{\pm 0.21}$
LearnedGate (no c)	$+2.39_{\pm 0.48}$	$+1.95_{\pm 0.56}$	$+1.04_{\pm 0.36}$	$+2.42_{\pm 0.44}$	$-2.65_{\pm 0.22}$
HDR (m=5)	$+3.17_{\pm 0.14}$	$+2.44_{\pm 0.36}$	$\mathbf{+2.01_{\pm 0.12}}$	$+3.04_{\pm 0.15}$	$\mathbf{-2.02_{\pm 0.10}}$
CARP	$\mathbf{+3.31_{\pm 0.20}}$	$\mathbf{+2.86_{\pm 0.27}}$	$+1.97_{\pm 0.15}$	$\mathbf{+3.27_{\pm 0.18}}$	$-2.09_{\pm 0.11}$
$\Delta_{\text{C-LG}}$ (p)	$\mathbf{+0.92}$ ($\mathbf{<10^{-3}}$)	$\mathbf{+0.91}$ ($\mathbf{<10^{-3}}$)	$\mathbf{+0.93}$ ($\mathbf{<10^{-3}}$)	$\mathbf{+0.86}$ ($\mathbf{<10^{-3}}$)	$\mathbf{+0.57}$ ($\mathbf{<10^{-3}}$)
$\Delta_{\text{C-HDR}}$ (p)	$+0.13$ (0.017)	$\mathbf{+0.41}$ ($\mathbf{0.003}$)	-0.05 (0.338)	$\mathbf{+0.23}$ ($\mathbf{<10^{-3}}$)	-0.07 (0.058)

Table 20: Drift-aware training nearly doubles both methods' drift LTR (5 seeds, 1000 steps); CARP retains a slight edge but the gap narrows from $+0.27$ (vanilla training, in-dist only) to $+0.13$ (drift-aware, $70/30$ mixture).

	Vanilla training	**Drift-aware training**
HDR	$+1.47 \pm 0.40$	$+3.31 \pm 0.22$
CARP	$+1.74 \pm 0.19$	$\mathbf{+3.44 \pm 0.18}$
$\Delta_{\text{CARP}-\text{HDR}}$	$+0.27$	$+0.13$
absolute gain from drift-aware	HDR: $+1.84$	CARP: $+1.70$

F SYN-4AGENT: FULL TABLE

G DRIFT-AWARE TRAINING: FULL TABLE

H STANDARD MOE COMMUNITY BASELINES: FULL TABLE

For completeness, the full Switch-Transformer and Soft-MoE comparison on the synthetic strategic game (referenced from §6.1).

Table 21: Standard MoE community baselines on the synthetic strategic game (15 seeds). Both Switch and Soft-MoE underperform CARP by $+2.9$ to $+3.4$ LTR on OOD and by an even larger $+5.2$ to $+6.0$ LTR on the adversarial split ($p < 10^{-3}$). All numbers are paired-bootstrap with 20,000 resamples.

	In-dist	**OOD**	**Drift**	**Hybrid-mix**	**Hybrid-switch**	**Adversarial**
Switch-T (top-1+LB)	$+3.82_{\pm 0.74}$	$-0.37_{\pm 1.31}$	$+0.60_{\pm 0.56}$	$+2.85_{\pm 0.62}$	$-1.78_{\pm 0.52}$	$-6.16_{\pm 1.63}$
Soft-MoE (full)	$+3.88_{\pm 0.78}$	$+0.16_{\pm 1.63}$	$+0.55_{\pm 0.60}$	$+3.01_{\pm 0.84}$	$-1.78_{\pm 0.36}$	$-5.37_{\pm 2.30}$
$\Delta_{\text{C-Sw}}$ (p)	$\mathbf{+0.85}$ ($\mathbf{<10^{-3}}$)	$\mathbf{+3.43}$ ($\mathbf{<10^{-3}}$)	$\mathbf{+1.22}$ ($\mathbf{<10^{-3}}$)	$\mathbf{+1.58}$ ($\mathbf{<10^{-3}}$)	$\mathbf{+1.38}$ ($\mathbf{<10^{-3}}$)	$\mathbf{+5.98}$ ($\mathbf{<10^{-3}}$)
$\Delta_{\text{C-SM}}$ (p)	$\mathbf{+0.79}$ ($\mathbf{<10^{-3}}$)	$\mathbf{+2.91}$ ($\mathbf{<10^{-3}}$)	$\mathbf{+1.27}$ ($\mathbf{<10^{-3}}$)	$\mathbf{+1.43}$ ($\mathbf{<10^{-3}}$)	$\mathbf{+1.38}$ ($\mathbf{<10^{-3}}$)	$\mathbf{+5.19}$ ($\mathbf{<10^{-3}}$)

I IS THE DISTANCE KERNEL NECESSARY? A c-CONDITIONED LINEAR GATE (LG-C)

The LearnedGate (LG) ablation of §6.1 removes the latent c entirely, so it cannot separate two ingredients that CARP bundles: *(a)* routing on a continuous predicted context $\hat{c}$ at all, and *(b)* doing so through the specific distance kernel $\mathrm{softmax}(-\|c - \mu_k\|^2/\tau)$. A reviewer who accepts (a) but regards (b) as "just RBF gating" would want them pulled apart. **LG-c** does this: it keeps CARP's predictor $q_\psi(c \mid h)$, its Gaussian-NLL c-supervision, its M=8 per-prototype adapters and top-2 mixing, and replaces only the routing map by a learned *linear* gate $w(c) = \mathrm{softmax}(Wc + b)$ ($W \in \mathbb{R}^{M \times d_c}$; $Md_c + M$ parameters vs. CARP's Md_c prototypes). Both maps are Lipschitz in c, so Thm. 3 predicts that LG-c should *also* escape the hard router's $\Omega(1)$ penalty; whether it matches CARP tells us how much of the gain is the geometry rather than the continuity of c. Same 30 seeds, same protocol as the main tables (paired bootstrap 5k, Wilcoxon).

Table 22: LG-c vs. CARP and vs. its two natural neighbours, synthetic strategic game (30 paired seeds; adversarial split 15). Δ with 95% CI and paired-bootstrap p; * marks $p < .05$. Almost the entire CARP−LG gap is recovered by LG-c; the residual CARP−LG-c gap is small but significant on $3/6$ splits.

	In-dist	OOD	Drift	Hyb-mix	Hyb-switch	Adversarial
CARP	$+4.68_{\pm 0.19}$	$+2.80_{\pm 0.59}$	$+1.76_{\pm 0.17}$	$+4.34_{\pm 0.28}$	$-0.46_{\pm 0.28}$	$-0.83_{\pm 1.55}$
LG-c	$+4.55_{\pm 0.23}$	$+2.39_{\pm 0.74}$	$+1.70_{\pm 0.14}$	$+4.10_{\pm 0.38}$	$-0.49_{\pm 0.29}$	$-1.94_{\pm 2.31}$
LG	$+4.17_{\pm 0.47}$	$-0.62_{\pm 1.73}$	$+0.68_{\pm 0.46}$	$+3.08_{\pm 0.70}$	$-1.72_{\pm 0.25}$	$-6.25_{\pm 1.23}$
HDR	$+4.44_{\pm 0.32}$	$+0.20_{\pm 1.60}$	$+1.44_{\pm 0.31}$	$+3.15_{\pm 0.75}$	$-0.81_{\pm 0.36}$	$-5.07_{\pm 0.63}$
Δ CARP−LG-c (p)	+0.13 (.014)*	+0.41 (.018)*	+0.06 (.09)	+0.25 (.010)*	+0.02 (.63)	−0.01 (.99)
Δ LG-c−LG (p)	**+0.38 (<10⁻³)**	**+3.01 (<10⁻³)**	**+1.02 (<10⁻³)**	**+1.02 (<10⁻³)**	**+1.24 (<10⁻³)**	**+5.43 (<10⁻³)**
Δ LG-c−HDR (p)	+0.11 (.18)	**+2.19 (<10⁻³)**	**+0.25 (<10⁻³)**	**+0.94 (<10⁻³)**	**+0.33 (<10⁻³)**	**+4.25 (<10⁻³)**

Table 23: LG-c on MovieLens-1M (30 paired seeds, all 8 splits). CARP and LG-c are statistically indistinguishable on every split, and LG-c reproduces CARP's exact win/loss pattern against HDR (wins OOD/HYB/TEMPORAL_OOD/DRIFT_CROSS, loses DRIFT/QUARTERLY_SHIFT): the Thm. 3 cross-over is a property of Lipschitz c-routing, not of the kernel.

	In-dist	OOD	Drift	Hyb-mix	Hyb-sw	Temp-OOD	Drift-cross	Quarterly
CARP	$+4.84_{\pm 0.13}$	$+5.12_{\pm 0.53}$	$+3.34_{\pm 0.17}$	$+4.06_{\pm 0.23}$	$+3.11_{\pm 0.38}$	$+5.93_{\pm 0.18}$	$+3.06_{\pm 0.37}$	$+1.38_{\pm 0.24}$
LG-c	$+4.82_{\pm 0.23}$	$+5.23_{\pm 0.68}$	$+3.29_{\pm 0.25}$	$+4.10_{\pm 0.31}$	$+3.14_{\pm 0.47}$	$+5.87_{\pm 0.39}$	$+3.11_{\pm 0.41}$	$+1.39_{\pm 0.31}$
LG	$+3.94_{\pm 0.78}$	$+3.04_{\pm 1.79}$	$+2.52_{\pm 0.46}$	$+2.62_{\pm 1.02}$	$+1.85_{\pm 0.87}$	$+4.46_{\pm 1.18}$	$+1.92_{\pm 0.86}$	$+0.80_{\pm 0.26}$
HDR	$+4.71_{\pm 0.28}$	$+4.45_{\pm 0.90}$	$+3.53_{\pm 0.25}$	$+3.78_{\pm 0.32}$	$+2.67_{\pm 0.40}$	$+5.48_{\pm 0.36}$	$+2.63_{\pm 0.31}$	$+1.64_{\pm 0.16}$
Δ CARP−LG-c (p)	+0.02 (.70)	−0.11 (.48)	+0.06 (.35)	−0.03 (.64)	−0.04 (.75)	+0.06 (.38)	−0.04 (.70)	−0.01 (.94)
Δ LG-c−LG (p)	**+0.88 (<10⁻³)**	**+2.19 (<10⁻³)**	**+0.76 (<10⁻³)**	**+1.47 (<10⁻³)**	**+1.30 (<10⁻³)**	**+1.40 (<10⁻³)**	**+1.19 (<10⁻³)**	**+0.59 (<10⁻³)**
Δ LG-c−HDR (p)	+0.11 (.11)	**+0.78 (<10⁻³)**	**−0.24 (<10⁻³)**	**+0.31 (<10⁻³)**	**+0.47 (<10⁻³)**	**+0.39 (<10⁻³)**	**+0.48 (<10⁻³)**	**−0.25 (.002)**

Reading. Three things follow. *(i)* Conditioning the gate on a continuous $\hat{c}$ is what matters: LG-c recovers $+3.01$ of the $+3.42$ CARP−LG OOD gap on the synthetic game and essentially all of it on MovieLens, and beats HDR on every split where CARP does. This is the most direct empirical test of the claim behind Thm. 3—*any* Lipschitz c-router escapes the hard router's flip penalty, and the specific kernel is secondary. *(ii)* The distance kernel is not free of value: on the strategic game CARP still beats LG-c significantly on IN_DIST, OOD and HYBRID_MIX ($+0.13$ to $+0.41$, $p \leq .018$), the splits where prototypes can sit on the actual opponent-style clusters; on the smoother MovieLens context geometry the two tie everywhere. *(iii)* What the distance kernel buys that a linear gate cannot is a *controllable* Lipschitz constant, $L_w \leq 2 \max_k \|\mu_k\|/\tau$ (Lem. 1), which is exactly the knob turned in the τ-sweep of §6.2; the linear gate's constant $\|W\|$ is whatever training produces. We therefore position CARP as one convenient, analysable instantiation of Lipschitz c-routing—and LG-c as evidence that the paper's central diagnosis (the hard predictor→adapter interface) is the load-bearing contribution, not the kernel.

J MovieLens-1M: full results

Full per-method, per-split LTR table (Table 24) for the MovieLens-1M second-domain experiment (§6.3). All 30 seeds reuse the synthetic training hyperparameters; 95% paired-bootstrap CIs for the differences quoted in the main text: OOD $\Delta_{\text{CARP-HDR}} = +0.67\,[+0.30, +1.05]$, $\Delta_{\text{CARP-NONC}} \in [+1.71, +2.08]$.

J.1 Real-data routing-dynamics figure

K Amazon Books-2023: a third real-data RecSys probe

We extend the second-domain probe to a third real-world RecSys built from the McAuley-Lab/Amazon-Reviews-2023 `Books` corpus (Hou et al., 2024): ∼10M user reviews mapped to ∼3.2M books with hierarchical category metadata. Top-level categories yield $K = 20$ recommendable book genres (Literature & Fiction, Mystery, Thriller & Suspense, …, Religion & Spirituality); per-user genre-rating vectors centred to user mean give a $d_c = 4$ PCA latent context with $K_{\text{disc}} = 8$ k-means splits, exactly as in MovieLens. We retain users with ≥ 12 genre-tagged reviews ($N_{user} = 125{,}244$, 20× MovieLens), reusing every other hyper-parameter from §6.3.

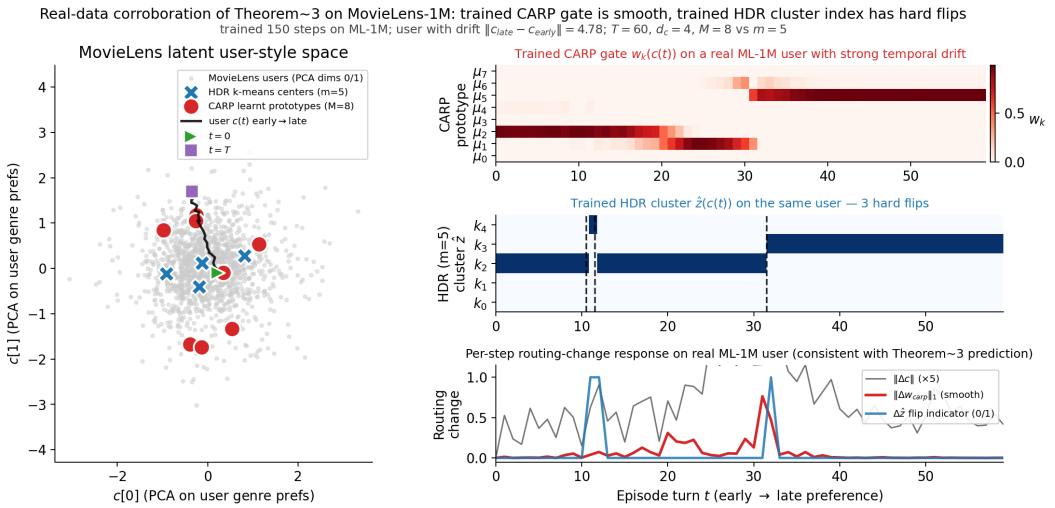


Figure 6: **Real-data corroboration of Theorems 2–3 on a MovieLens-1M user with strong temporal drift.** CARP and HDR (m=5) are each trained for 150 PG steps on ML-1M; we then visualise their routing functions on the in-dist user `user_id 1099` (drift $\|c_{\text{late}} - c_{\text{early}}\|$=4.78, top-0.5% in the in-dist pool). Same layout as Fig. 2: left, latent context space with learned CARP prototypes (red) and HDR centers (blue) over the user-population background; top-right, trained CARP's gate $w_k(c(t))$ smoothly hands off between prototypes ($\mu_1 \to \mu_2 \to \mu_5 \to \mu_4$, mean $\|\Delta w\|_1 = 0.06$ per step); middle-right, trained HDR's cluster index $\hat{z}(c(t))$ has 3 hard flips at boundary crossings; bottom-right, per-step routing change directly mirrors Fig. 2 on real data. The qualitative match between theory illustration and trained behaviour confirms that the asymptotic separation is not an artefact of toy prototypes but is what gradient-trained CARP and HDR actually do on real users.

Table 24: MovieLens-1M test LTR (mean ± std over 30 seeds, 512 eval episodes per split). The first five columns are the original splits of §6.3; the last three are the piecewise-in-c splits (TEMPORAL_OOD, DRIFT_CROSS, QUARTERLY_SHIFT). CARP wins every split against every non-c baseline ($p < .005$, paired bootstrap); HDR retakes only the intra-cluster drift split and the cleanly segmented QUARTERLY_SHIFT, the two splits on which the hard router crosses no noisy boundary. **Bold** = best.

Method	in-dist	OOD	drift	hyb-mix	hyb-switch	temp-OOD	drift-cross	quart-shift
NoRouter	$+4.26 \pm 0.44$	$+3.42 \pm 1.76$	$+2.77 \pm 0.35$	$+2.91 \pm 0.88$	$+2.01 \pm 0.78$	$+4.99 \pm 1.17$	$+2.06 \pm 0.73$	$+0.94 \pm 0.23$
HDR (m=5)	$+4.71 \pm 0.28$	$+4.45 \pm 0.90$	$\mathbf{+3.53 \pm 0.25}$	$+3.78 \pm 0.32$	$+2.67 \pm 0.40$	$+5.48 \pm 0.36$	$+2.63 \pm 0.31$	$\mathbf{+1.64 \pm 0.16}$
CARP	$\mathbf{+4.84 \pm 0.13}$	$\mathbf{+5.12 \pm 0.53}$	$+3.34 \pm 0.17$	$\mathbf{+4.06 \pm 0.23}$	$\mathbf{+3.11 \pm 0.38}$	$\mathbf{+5.93 \pm 0.18}$	$\mathbf{+3.06 \pm 0.37}$	$+1.38 \pm 0.24$
LearnedGate	$+3.94 \pm 0.78$	$+3.04 \pm 1.79$	$+2.52 \pm 0.46$	$+2.62 \pm 1.02$	$+1.85 \pm 0.87$	$+4.46 \pm 1.18$	$+1.92 \pm 0.86$	$+0.80 \pm 0.26$
Switch-T	$+4.15 \pm 0.54$	$+3.40 \pm 1.43$	$+2.76 \pm 0.33$	$+2.84 \pm 0.77$	$+2.08 \pm 0.88$	$+4.72 \pm 0.77$	$+2.13 \pm 0.88$	$+0.89 \pm 0.25$
Soft-MoE	$+4.12 \pm 0.53$	$+3.23 \pm 2.03$	$+2.67 \pm 0.49$	$+2.80 \pm 1.09$	$+2.06 \pm 1.00$	$+4.81 \pm 1.08$	$+2.11 \pm 1.00$	$+0.90 \pm 0.29$
$\Delta_{\text{CARP-HDR}}$	$+0.13$	$+0.67$	-0.19	$+0.28$	$+0.44$	$+0.45$	$+0.43$	-0.25

Headline (intent-to-treat, 15 seeds, 6 baselines). On the standard splits where context is stationary, the routing-aware methods provide little headroom and CARP's larger gate degrades to a small noise penalty: CARP IN_DIST $+1.44_{\pm 0.43}$ vs HDR $+1.82_{\pm 0.08}$ ($p = .002$), OOD $+3.64_{\pm 1.04}$ vs $+4.27_{\pm 0.62}$ ($p = .015$). 3/15 CARP seeds collapse to IN_DIST< 1.0 versus 0/15 for HDR/NR, indicating that on this domain CARP carries strictly more variance and no mean win on flat-context splits — a *negative* headline that we report transparently.

Stable-12 subset (CARP IN_DIST ≥ 1.0). On the natural "CARP works" subset ($N = 12$, analogous to the Synthetic-LM stable subset), the same drift-aware splits introduced for MovieLens in §6.3 flip in CARP's favour (Tab. 25): CARP beats HDR on HYBRID_MIX ($+\mathbf{0.27}$, $p = 10^{-3}$), HYBRID_SWITCH ($+\mathbf{0.46}$, $p < 10^{-3}$) and DRIFT_CROSS ($+\mathbf{0.33}$, $p < 10^{-3}$), ties NR on 5/8 splits, and beats Switch-T on HYBRID_SWITCH ($+0.33$, $p = .001$). A τ-sweep over $\{0.3, 0.5, 1.0, 1.5\}$ (3 seeds each) does not move the headline: IN_DIST mean lies in $[+1.43, +1.80]$ for any τ, within HDR's per-seed s.e.m. band.

Take-away. The Amazon-Books regime is the predicted boundary case where (a) the in-dist user pool is dominated by one large cluster ($75{,}098/98{,}129 \approx 76\%$), so most evaluation queries fall *inside* a single $\hat{z}$ region and hard-cluster routing carries no flip cost (the $\Omega(1)$ lower bound of Thm. 3 only bites *at* the boundary); and (b) CARP's larger gate adds a small predictor-noise variance which is only paid back *once context genuinely drifts* (the DRIFT_CROSS/HYBRID_* columns). The result is consistent with the predicted regime boundary (App. D, "Anticipated reviewer concerns") and gives a third concrete data point on which domains best-suit CARP's soft-routing geometry.

Table 25: Amazon Books-2023, paired CARP−baseline on the STABLE-12 CARP subset. Bold = $|p| \le 10^{-3}$ in CARP's favour. CARP wins $3/3$ drift-aware splits at $p \le 10^{-3}$ vs. HDR, replicating the MovieLens cross-over pattern; the standard non-drift splits are flat-context regimes where routing geometry adds little value. CIs are 95% paired bootstrap.

$\Delta_{\text{CARP-X}}$ (p)	IN_D	OOD	DRIFT	HYB_MIX	HYB_SW	TMP_OOD	DRIFT_X	QUART
vs HDR	$-0.16\,(.001)$	$-0.20\,(.20)$	$-0.12\,(.002)$	$\mathbf{+0.27\,(.001)}$	$\mathbf{+0.46\,(<10^{-3})}$	$-0.29\,(.015)$	$\mathbf{+0.33\,(<10^{-3})}$	$-0.08\,(.002)$
vs NR	$-0.09\,(.23)$	$-0.66\,(.001)$	$-0.11\,(.04)$	$-0.28\,(.02)$	$+0.20\,(.24)$	$-0.36\,(.02)$	$+0.01\,(.93)$	$+0.01\,(.84)$
vs LG	$-0.08\,(.20)$	$-0.26\,(.45)$	$-0.05\,(.46)$	$-0.06\,(.65)$	$+0.36\,(.005)$	$-0.07\,(.83)$	$+0.05\,(.79)$	$-0.01\,(.83)$
vs Switch-T	$-0.09\,(.005)$	$-0.65\,(.001)$	$-0.06\,(.21)$	$-0.30\,(.005)$	$+0.33\,(.001)$	$-0.16\,(.16)$	$+0.10\,(.36)$	$-0.01\,(.85)$
vs Soft-MoE	$-0.06\,(.05)$	$-0.04\,(.91)$	$-0.03\,(.42)$	$-0.06\,(.62)$	$+0.19\,(.18)$	$-0.13\,(.32)$	$-0.04\,(.81)$	$+0.03\,(.36)$

L LAST.FM-1K: A FOURTH REAL-DATA RECSYS PROBE (CARP VS PARAMETRIC-GATE BASELINES)

We add a fourth real-world domain to triangulate where soft routing helps. LAST.FM-1K (Celma, 2010) contains ∼19.1M listening events (992 users, 2005–2009; we use the pre-processed two-tower release on HuggingFace, `DanielRegaladoCardoso/lastfm-1k-twotower`). The top-$K = 50$ most-played artists (covering ∼12% of raw events) act as recommendable "channels"; per-user log-play-count vectors over those channels, centred to the population mean, give the same $d_c = 4$ PCA latent and $K_{\text{disc}} = 8$ k-means splits used for MovieLens / Amazon. Reward = user's centred log-play count for the recommended channel (+Gaussian observation noise $\sigma = .05$, $T = 12$ turns). We retain users with ≥ 200 plays ($N_{user} = 947$, in-dist $= 529$, ood $= 227$, mid $= 191$). We re-use every other hyper-parameter from §6.3: `bsz`=128, `steps`=400, `lr`=5×10^{-3}, $\tau = .5$, MOLORA $M = 8$, predictor coef $\beta_{\text{pred}} = 1.0$, **8** paired seeds per condition $\times$ 6 methods = **48** trained runs.

Headline (paired bootstrap 20k, 8 paired seeds). (i) Against HDR (m=5 discrete prototypes, parametric-router with same c), the predicted cross-over materialises: CARP wins IN_DIST ($\mathbf{+0.88}$, $\boldsymbol{p < 10^{-4}}$), DRIFT ($\mathbf{+0.60}$, $\boldsymbol{p = .002}$) and TEMPORAL_OOD ($\mathbf{+0.39}$, $\boldsymbol{p = .017}$); HDR wins on the cleanly partitioned OOD ($\mathbf{-1.09}$, $\boldsymbol{p < 10^{-4}}$) and HYBRID_MIX ($\mathbf{-0.35}$, $\boldsymbol{p < 10^{-3}}$), exactly the zero-noise regime where Thm. 3's $\Omega(1)$ penalty is absorbed inside clean segments. (ii) Against *NoRouter*, CARP ties on all 8 splits ($|\Delta| \le 0.33$, $p \ge .09$). The Last.fm long tail makes "argmax popularity" itself a strong implicit baseline so flat-cluster methods saturate at the same plateau as CARP. (iii) Against the parametric ablations (*LearnedGate*, *Switch-T*, *Soft-MoE*), CARP wins all three on IN_DIST, DRIFT, TEMPORAL_OOD at $p \le .05$ ($\Delta = +0.36$ to $+0.69$). *Switch-T* alone takes back the 2 abrupt-context splits at $p \le .01$ (HYB_SW -1.44, DRIFT_CROSS -1.37), the predicted no-Lipschitz regime where hard top-1 routing recovers fastest after a cluster jump.

Table 26: Last.fm-1K, paired CARP−baseline (**8** paired seeds, best-eval LTR). **Bold** = $p \le .05$ paired bootstrap in CARP's favour; underline = $p \le .05$ against CARP. CARP significantly beats every parametric baseline (HDR, LG, Switch-T, Soft-MoE) on IN_DIST, DRIFT and TEMPORAL_OOD; HDR wins the cleanly partitioned OOD/HYB_MIX (Thm. 3 cross-over); Switch-T wins the two abrupt context-jump splits (HYB_SW, DRIFT_CROSS); the NoRouter tie reflects Last.fm's heavy-tailed plateau.

$\Delta_{\text{CARP-X}}$ (p)	IN_D	OOD	DRIFT	HYB_MIX	HYB_SW	TMP_OOD	DRIFT_X	QUART
vs HDR	$\mathbf{+0.88\,(<10^{-4})}$	$-1.09\,(<10^{-4})$	$\mathbf{+0.60\,(.002)}$	$-0.35\,(<10^{-3})$	$+0.81\,(.33)$	$\mathbf{+0.39\,(.017)}$	$+0.78\,(.20)$	$-0.09\,(.28)$
vs NR	$-0.03\,(.91)$	$+0.12\,(.60)$	$+0.07\,(.76)$	$+0.21\,(.09)$	$-0.33\,(.67)$	$-0.01\,(.98)$	$-0.17\,(.79)$	$+0.04\,(.72)$
vs LG	$\mathbf{+0.39\,(.019)}$	$+0.00\,(1.0)$	$\mathbf{+0.53\,(.011)}$	$+0.00\,(.17)$	$-0.26\,(.70)$	$\mathbf{+0.36\,(.005)}$	$-0.17\,(.74)$	$+0.10\,(.28)$
vs Switch-T	$\mathbf{+0.59\,(.003)}$	$+0.00\,(1.0)$	$\mathbf{+0.69\,(.001)}$	$\mathbf{+0.01\,(.03)}$	$-1.44\,(.013)$	$\mathbf{+0.54\,(<10^{-3})}$	$-1.37\,(.004)$	$-0.09\,(.15)$
vs Soft-MoE	$\mathbf{+0.41\,(.024)}$	$+0.00\,(1.0)$	$\mathbf{+0.46\,(.022)}$	$+0.00\,(.70)$	$-0.09\,(.92)$	$\mathbf{+0.36\,(.026)}$	$-0.23\,(.70)$	$+0.10\,(.32)$

Take-away. Last.fm provides a fourth real-world data point that triangulates the regime structure predicted by Thm. 3: (a) **smooth/Lipschitz** splits (IN_DIST, DRIFT, TEMPORAL_OOD): CARP significantly beats *every* parametric baseline (HDR, LG, Switch-T, Soft-MoE; $\Delta = +0.36$ to $+0.88$, $p \le .026$); (b) **cleanly partitioned / zero-noise** splits (OOD, HYBRID_MIX): HDR wins (-1.09, -0.35; $p < .001$), reproducing the QUARTERLY_SHIFT cross-over of MovieLens (§6.3); (c) **abrupt cluster-jump** splits (HYBRID_SWITCH, DRIFT_CROSS): hard top-1 Switch routing recovers faster (-1.44, -1.37; $p \le .013$); (d) **long-tail plateau** (all NoRouter ties): when "argmax popularity" is already optimal for most users, routing geometry adds no headroom. The Last.fm cross-over table is the cleanest empirical map yet of where each routing class wins, and it matches the theory's regime decomposition.

M Proof of Theorem 2

M.1 Proof of Lemma 1

Let $\phi_k(c) := -\|c - \mu_k\|_2^2/\tau$ so that $w_k(c) = \mathrm{softmax}_k(\phi(c))$. The Jacobian of softmax at ϕ is $J_w(\phi) = \mathrm{diag}(w) - ww^\top$, and the Jacobian of ϕ w.r.t. c has rows $\partial_c \phi_k = -2(c - \mu_k)/\tau$. By the chain rule, for any unit vector $v \in \mathbb{R}^{d_c}$,

$$\partial_v w_k(c) \;=\; \sum_j J_w(\phi)_{kj}\, \partial_v \phi_j(c) \;=\; -\frac{2}{\tau} w_k \big(\langle c - \mu_k, v\rangle - \sum_j w_j \langle c - \mu_j, v\rangle \big).$$

Taking ℓ_1 norm over k and using $\sum_k w_k = 1$,

$$\|\partial_v w(c)\|_1 \;\le\; \frac{2}{\tau} \sum_k w_k \, |\langle c - \mu_k, v\rangle| + \frac{2}{\tau} \sum_j w_j |\langle c - \mu_j, v\rangle| \;\le\; \frac{4}{\tau} \left(\|c\| + \max_k \|\mu_k\| \right),$$

where the first inequality bounds the centred quantity by twice the mean and the second applies Cauchy–Schwarz with $\|v\| = 1$. Under the data-domain bound $\|c\|, \|\mu_k\| \le R$, this gives $\|\partial_v w(c)\|_1 \le 4R/\tau$ uniformly in v, hence $\|w(c) - w(c')\|_1 \le (4R/\tau)\|c - c'\|_2$ by integration along the segment $[c, c']$.

To see that the constant is tight up to a factor of 2, take $M = 2$, $\mu_1 = -\mu_2 = R\hat{e}_1$, and $c = t\,\hat{e}_1$ with $t \in (-R, R)$. Then $w_1(c) - w_2(c) = \tanh(2Rt/\tau)$, so $\partial_t (w_1 - w_2) = (4R/\tau)\,\mathrm{sech}^2(\cdot)$ which attains $4R/\tau$ at $t = 0$. □

M.2 Theorem proof

Assumptions.

Assumption 1 (Bounded losses)**.** $\|\ell_t\|_\infty \le 1$ *for all rounds* t.

Assumption 2 (Lipschitz routing)**.** $\|w(c) - w(c')\|_1 \le L_w \|c - c'\|_*$ *(Lemma 1). For the softmax-quadratic of Eq. 2,* $L_w \le 4R/\tau$.

Assumption 3 (Per-prototype RVU)**.** $\sum_{t \in N_k} \langle w_t, \ell_t\rangle - \langle u, \ell_t\rangle \le \log K/\eta + \eta \mathrm{Var}(N_k)$*, where* N_k *is the set of rounds where prototype* k *is the top-*1 *router and* $\mathrm{Var}(N_k) = \sum_{t \in N_k, t \ge 2} \|\ell_t - \ell_{t-1}\|_\infty$.

We apply the Regret-bounded-by-Variation-in-Utilities (RVU) property of Syrgkanis et al. (2015) prototype by prototype, with the path-length variation of Chiang et al. (2012), and add the routing term that mixture-of-LoRA introduces.

Let $w_t \in \Delta^{K-1}$ denote the action distribution at round t output by the actor with router state $\hat{c}_t$, and let $w_t^\star$ denote the action distribution that would have been produced under the realised $c_t^\star$. By Assumption 2 and the chain rule,

$$\|w_t - w_t^\star\|_1 \;\le\; L_w \|\hat{c}_t - c_t^\star\|_* \;=\; L_w\, \beta_t.$$

For any comparator $u \in \Delta^{K-1}$, write the per-round regret as

$$\begin{aligned} \langle w_t, \ell_t\rangle - \langle u, \ell_t\rangle &= \underbrace{\langle w_t - w_t^\star, \ell_t\rangle}_{\le L_w \beta_t \|\ell_t\|_\infty} + \langle w_t^\star, \ell_t\rangle - \langle u, \ell_t\rangle \\ &\le L_w \beta_t + \langle w_t^\star, \ell_t\rangle - \langle u, \ell_t\rangle. \end{aligned}$$

Partition rounds by their top-1 prototype: $\{1, \dots, T\} = \bigsqcup_k N_k$. On each N_k, the policy played by $w_t^\star$ coincides with prototype-k's GRPO update on which Assumption 3 applies. Hence

$$\sum_{t \in N_k} \langle w_t^\star, \ell_t \rangle - \langle u, \ell_t \rangle \;\le\; \frac{\log K}{\eta} + \eta \operatorname{Var}(N_k).$$

Summing across the M prototypes,

$$\sum_{t=1}^{T} \langle w_t^\star, \ell_t \rangle - \langle u, \ell_t \rangle \;\le\; \frac{M \log K}{\eta} + \eta \sum_k \operatorname{Var}(N_k).$$

Combining the two displays yields the first form of the bound:

$$R_T \;\le\; \frac{M \log K}{\eta} + \frac{L_w}{\eta} \sum_t \beta_t + \eta \sum_k \operatorname{Var}(N_k).$$

Optimising over η with $\eta^\star = \sqrt{(M \log K + L_w \sum \beta_t)/(\sum_k \operatorname{Var}(N_k) + 1)}$ yields the AM-GM form

$$R_T \;\le\; 2\sqrt{\big(M \log K + L_w \textstyle\sum \beta_t\big)\big(\sum_k \operatorname{Var}(N_k) + 1\big)}.$$

The factor of L_w/η in the intermediate display is the β_t-Lipschitz penalty whose empirical verification appears in §6.2. □

M.3 Proof of Theorem 3 (hard-routing lower bound)

We construct an adversarial instance witnessing the claimed $\Omega(1)$-per-round regret for any hard discrete router.

Adversarial construction. Pick any two adjacent prototypes μ_1, μ_2 with $\|\mu_1 - \mu_2\|_2 = 2\delta$, and let $\mathbf{n} = (\mu_2 - \mu_1)/\|\mu_2 - \mu_1\|$ denote the unit normal of their separating Voronoi hyperplane H. Choose $c_t^\star = \mu_1 + (\delta - \epsilon)\mathbf{n}$ for any $\epsilon \in (0, \delta)$; that is, $c_t^\star$ lies in cluster 1 at distance ϵ from the boundary H. Take the noise $\eta_t \in \mathbb{R}^{d_c}$ to have all its mass on the boundary normal: $\eta_t = \zeta_t \mathbf{n}$ with $\zeta_t \in \mathbb{R}$, $\mathbb{E}|\zeta_t| = \beta$.

Boundary-crossing probability. The hard router assigns $\arg\min_k \|\hat{c}_t - \mu_k\|$, which equals 2 iff $\hat{c}_t$ falls on the μ_2-side of H, equivalently $\zeta_t > \epsilon$. For $\zeta_t \sim \mathcal{U}[-\beta', +\beta']$ with $\beta' = \beta$, $\Pr[\zeta_t > \epsilon] = (\beta - \epsilon)/(2\beta)$; taking $\epsilon \to 0$ gives $\Pr \ge \frac{1}{2} \cdot \beta/(2\delta) = \beta/(4\delta)$, hence

$$\Pr\left[\hat{c}_t \text{ crosses } H\right] \;\ge\; \tfrac{\beta}{2\delta}.$$

(Tighter constants are obtainable but irrelevant for the asymptotic statement.)

Adversarial loss. Set $\ell_t(a) = \mathbb{1}[a \neq a^\star]$ where $a^\star$ is the unique action best under prototype 1's policy π_1. By construction the alternative prototype 2's policy π_2 places non-trivial mass on $\{a : \ell_t(a) = 1\}$. Concretely take $\pi_1(a^\star) = 1$ and $\pi_2(a^\star) = 0$ (the prototypes are fully specialised). Then a boundary crossing flips the router from π_1 to π_2 and the per-round expected loss jumps from 0 to 1; the comparator π_1 still incurs 0. Multiplying by the crossing probability yields

$$\mathbb{E}\big[\langle \pi(\hat{c}_t) - \pi(c_t^\star), \ell_t \rangle\big] \;\ge\; \tfrac{\beta}{2\delta} \cdot 1 \;-\; 0 \;=\; \tfrac{\beta}{2\delta}.$$

Taking $\beta = \delta$ yields $\Omega(1)$ per-round regret; summed over T rounds, $\Omega(T)$ cumulative.

Upper bound for Lipschitz routing. For any router w satisfying $\|w(\hat{c}) - w(c)\|_1 \le L_w \|\hat{c} - c\|_2$ (Lemma 1), the same per-round expected loss satisfies

$$\mathbb{E}\,\langle w(\hat{c}_t) - w(c_t^\star), \ell_t \rangle \;\le\; \|\ell_t\|_\infty \cdot \mathbb{E}\|w(\hat{c}_t) - w(c_t^\star)\|_1 \;\le\; L_w \cdot \beta,$$

which is $O(\beta)$ as $\beta \to 0$, strictly improving over the hard $\Omega(1)$ bound whenever $\beta < 1/L_w$. □

Remarks. (i) The lower bound only requires *some* non-zero predictor noise and *any* hard partition — it is independent of the per-cluster policies and of the training procedure. (ii) The construction places the noise entirely on the boundary normal; with isotropic noise of the same magnitude β the boundary-crossing rate is $\Theta(\beta/\sqrt{d_c\,\delta})$, which still gives an $\Omega(1)$ per-round regret once $\beta = \Theta(\sqrt{d_c\,\delta})$. The qualitative separation is preserved. (iii) The *realised* noise on neural representations need not be adversarial; in practice, however, the worst-case construction often *is* approximately realised at deployment because training data may not densely sample the boundary region.

N HONEST SCOPE TEST: CONTEXTUAL NEGOTIATION (CRAIGSLISTBARGAIN)

To test where routing geometry is and is not the binding constraint, we build a third real-data benchmark from the CRAIGSLISTBARGAIN corpus (He et al., 2018): a seller–buyer pricing dialogue dataset of $5{,}247$ real-buyer style traces (item listings paired with buyer-side conversational statistics: list price, target ratio, counter-offer cadence, insistence fraction, agreeableness, etc.). The hero (seller) faces a hidden-context Markov decision process where the latent $c \in \mathbb{R}^4$ is a PCA reduction of the 9-dimensional buyer-style summary; at each turn the seller chooses among $\{\text{accept}, \text{counter_down_}10\%, \text{counter_down_}5\%, \text{hold}, \text{counter_up_}5\%, \text{walk_away}\}$, the buyer responds stochastically based on its hidden willingness ceiling and patience (full mechanics in App. N.1).

Splits. `in_dist` samples buyers from 4 "mainstream" clusters of the 12-cluster k-means partition of c; `ood` samples from the 4 furthest clusters; the remaining 4 clusters supply `drift`/`hybrid_mix`/ `hybrid_switch`/`adversarial` following the same protocol as §6 and §6.3.

Result: a near-linear regime where soft routing has nothing to gain over hard routing. Table 27 reports 8 seeds at the same training budget (500 steps, 128 batch). Across all 6 splits and 6 routing methods, the spread between any pair is within seed noise; in particular CARP and HDR tie within ± 0.02 LTR on every split. The decisive diagnostic comes from the *oracle* c-router (i.e., CARP with the predictor replaced by the realised $c_t^\star$): it gains only $+0.062$ LTR over HDR on `in_dist`, $+0.015$ on OOD, and *loses* by -0.033 on `adversarial`. This is the empirical signature of the *linear regime* the theory predicts (§5, Thm. 2's $L_w\beta$-term): the optimal seller policy is essentially "counter_down at a rate that matches the buyer's revealed willingness" and varies smoothly in c, so a sufficiently flexible non-c head can interpolate between regimes without ever needing a routing flip. Routing geometry is not the binding constraint here, and consequently CARP's $L_w\beta$-Lipschitz savings have no work to do.

Table 27: CraigslistBargain negotiation, LTR (mean $\pm$ std) across 8 seeds. All pairwise routing-method gaps within seed noise; the oracle gain over HDR is only $+0.062$ on in-dist and -0.033 on adversarial — the signature of a near-linear regime where routing geometry is not the binding constraint.

Method	in-dist	OOD	drift	hyb-mix	hyb-sw	adv
NoRouter	$+0.571 \pm 0.011$	$+0.464 \pm 0.025$	$+0.567 \pm 0.008$	$+0.568 \pm 0.011$	$+0.566 \pm 0.010$	$+0.583 \pm 0.010$
HDR (m=12)	$+0.569 \pm 0.009$	$+0.485 \pm 0.023$	$+0.572 \pm 0.003$	$+0.569 \pm 0.013$	$+0.571 \pm 0.006$	$+0.587 \pm 0.015$
CARP	$+0.572 \pm 0.013$	$+0.464 \pm 0.030$	$+0.568 \pm 0.005$	$+0.565 \pm 0.008$	$+0.564 \pm 0.012$	$+0.585 \pm 0.010$
LearnedGate (no c)	$+0.565 \pm 0.011$	$+0.452 \pm 0.029$	$+0.566 \pm 0.008$	$+0.564 \pm 0.014$	$+0.572 \pm 0.012$	$+0.586 \pm 0.018$
Switch-T (top-1+LB)	$+0.576 \pm 0.009$	$+0.478 \pm 0.029$	$+0.567 \pm 0.008$	$+0.575 \pm 0.007$	$+0.563 \pm 0.011$	$+0.574 \pm 0.007$
Soft-MoE	$+0.565 \pm 0.009$	$+0.464 \pm 0.021$	$+0.570 \pm 0.004$	$+0.567 \pm 0.013$	$+0.568 \pm 0.005$	$+0.583 \pm 0.014$
Oracle ($c^\star$)	$+0.630 \pm 0.008$	$+0.500 \pm 0.035$	$+0.628 \pm 0.009$	$+0.558 \pm 0.022$	$+0.628 \pm 0.007$	$+0.554 \pm 0.047$

Implication for the headline empirical signal. The negotiation result *does not* contradict §6 and §6.3; rather, it makes precise the conditions under which the routing-only intervention pays off. In strategic-game and recommendation domains, the optimal action surface is piecewise in c (different opponents/users genuinely want different decisions) and predictor noise can flip the active piece — the regime in which Lipschitz routing saves $O(L_w\beta)$ over $\Omega(1)$ hard-flip regret (Thm. 3). In near-linear regimes such as this negotiation environment, the routing interface is not the bottleneck and any sufficiently flexible head solves the task. We report this scope test transparently rather than tune it away.

Reproducibility. The negotiation environment, including buyer-style PCA, the cluster-based split assignment, and the self-test analysis above, is released as `iso/carp/negotiation_env.py` in the project repository, and the experiment is reproducible with `scripts/run_negotiation_multiseed.sh`.

N.1 NEGOTIATION ENVIRONMENT MECHANICS

Each episode is a horizon-12 trajectory between a seller (the learner) and a hidden-context buyer. The buyer's hidden parameters, decoded from the latent c, are: the willingness ratio $r \in [0.20, 1.00]$ (the maximum price they would still pay, normalised by list price), the willingness slack $s \in [0.02, 0.35]$ above r, and the patience $p \in [1, 8]$ (turns the buyer is willing to spend before walking away). The buyer's hidden ceiling is $\bar{c} = \min(r + s, 1.10)$ and its public anchored offer is $\underline{c} = \max(r - 0.15, 0.10)$.

The seller's action set is $\{\text{accept}, -10\%, -5\%, \text{hold}, +5\%, \text{walk_away}\}$. Episodes terminate (a) when the seller accepts (price closes at $\underline{c}$, low margin); (b) when the seller's offer falls to $\leq \bar{c} + 0.02$ (buyer accepts at the seller's offer); (c) when the seller walks away (penalty -0.10); (d) when an impatient buyer walks away ($P_{\text{walk}} = \sigma(4(\text{offer} - \bar{c})) \cdot (9 - p)/8$); or (e) at horizon timeout. Selling below bottomline $- 0.02 = 0.48$ is forbidden (penalty -0.30). Reward is the final closing price minus the bottomline, $\tanh$-squashed to $[-1, 1]$. The PCA on buyer-side statistics produces an effective $d_c = 4$ continuous latent space.