When Policies Cannot Be Retrained: A Unified Closed-Form View of Post-Training Steering in Offline Reinforcement Learning

Elias Hossain* **Mohammad Jahid Ibna Basher** **Ivan Garibay**
Ozlem Garibay **Niloofar Yousefi***

Complex Adaptive Systems Laboratory
Department of Industrial Engineering and Management Systems
University of Central Florida
Orlando, FL, USA

{mdelias.hossain,niloofar.yousefi}@ucf.edu

*Corresponding authors: Elias Hossain and Niloofar Yousefi

Abstract

Offline RL can learn effective policies from fixed datasets, but deployment objectives often shift after training, while the trained actor may be frozen due to data, cost, or governance constraints. We study deployment-time adaptation of frozen offline actors using Product-of-Experts (PoE) composition with a goal-conditioned prior. Our main practical finding is graceful degradation rather than universal improvement: with degraded or random priors, precision-weighted PoE remains anchored to the frozen actor, whereas additive and prior-only adaptation often collapse, and a dataset-only KL-budget rule offers a non-oracle way to choose the composition strength. We also clarify that, for diagonal-Gaussian actors and priors, PoE with coefficient α induces the same deterministic policy as KL-regularized adaptation with $\beta = \alpha/(1-\alpha)$, with posterior covariance differences cancelling under deterministic action selection. Across D4RL MuJoCo, AntMaze, and Adroit, results reveal an actor-competence ceiling: composition can help or preserve a competent frozen actor, can also hurt one that is near its ceiling, and cannot recover when the actor itself fails. Swept CQL/IQL-guided and SF/GPI baselines are competitive in some settings, but do not eliminate this regime boundary. Thus, PoE/KL-Reg is best viewed as an actor-anchored safety layer for deployment-time objective shift, not as a general return-maximization method, with effectiveness governed by actor competence rather than knob choice.[1]

1 Introduction

Offline reinforcement learning (RL) [Levine et al., 2020] trains policies from previously collected data without environment interaction [Chenreddy and Delage, 2026, Park et al., 2025, Zhang et al., 2026]. Deployment objectives often change after training, but existing offline RL methods address such changes during training through conservative value estimation or behavior-regularized optimization [Chen et al., 2023, Li et al., 2023, Guan et al., 2024]. A practically important setting remains less explored: the actor has already been trained and approved, the objective is revised afterward, and modifying the actor is not feasible. This frozen-actor setting arises when data access is limited, retraining is expensive, or any policy change requires governance or revalidation.

[1] Code: https://github.com/eliashossain001/poe_composition

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

We consider *actor-preserving deployment-time adaptation*: steering a frozen offline actor toward a revised objective while remaining conservative relative to the validated policy. We combine a fixed actor $\pi_\theta(a \mid s)$ with a goal-conditioned deployment prior $\rho_\phi(a \mid s, g)$ through a Product-of-Experts (PoE) refinement [Welling, 2007]. The refined policy solves an entropy-regularized objective and, for diagonal-Gaussian actors and priors, admits a precision-weighted closed form. As a clarifying unification in the frozen-actor deployment setting we make explicit that this closed form is algebraically the same deterministic policy as the KL-regularized frozen-actor adaptation rule under $\beta = \alpha/(1-\alpha)$; the underlying Gaussian KL-barycenter algebra is standard, but the two rules have been treated as distinct mechanisms in prior work, and we organize the empirical analysis around this identity.

Empirically, we study this composition on four D4RL MuJoCo task settings, namely HalfCheetah, Hopper, and Walker2d medium and Hopper medium-expert (3,900 rollout episodes, plus 3,900 for a ten-seed replication; 95% bootstrap CIs) [Fu et al., 2020, Todorov et al., 2012], six harder cells, and AntMaze diagnostics. The central finding is *graceful degradation*: under degraded or random priors, precision-weighted composition stays close to the frozen actor while prior-only and additive adaptation collapse. An α sweep then exposes an *actor-competence ceiling*: composition is bounded by the competence of the frozen actor rather than by the choice of knob.

Contributions.

- **Closed-form unification.** For diagonal-Gaussian actors and priors, PoE refinement $\pi_{\text{ref}} \propto \pi_\theta^\alpha \rho_\phi^{1-\alpha}$ and KL-regularized adaptation with $\beta = \alpha/(1-\alpha)$ give the same deterministic policy; their covariances differ by the scalar $1/\alpha$, which cancels under deterministic action selection. The Gaussian algebra is standard; we make the identity explicit for frozen-actor deployment, where the two rules have been treated as separate mechanisms.
- **Robustness under degraded priors.** Under noisy or random priors, PoE/KL-Reg stay anchored to the frozen actor, while additive and prior-only adaptation collapse. A dataset-only KL-budget rule chooses α without rollouts.
- **Boundary characterization.** Composition helps, ties, or hurts in 4/4/4 of the 12 headline cells (ten seeds). An α sweep on six harder cells and AntMaze diagnostics shows an *actor-competence ceiling*: medium-expert is HURT in 8 or 9 of 9 cells at every α, and steering cannot lift a zero-success actor.
- **Baselines and theory.** Swept CQL/IQL-guided and SF/GPI baselines are competitive in places but do not remove this boundary. Theorem 2 serves only as a qualitative gain–deviation decomposition, not as a predictive bound.

2 Related Work

Offline reinforcement learning. Offline RL controls distributional mismatch between the behavior and learned policies [Kumar et al., 2020, Fujimoto et al., 2019, Kostrikov et al., 2021], and advantage-weighted regression / AWAC [Peng et al., 2019, Nair et al., 2020] give the closed-form KL-regularized update $\pi(a \mid s) \propto \pi_{\text{behav}}(a \mid s)\, \exp(A(s,a)/\beta)$ as the canonical Gaussian trust-region reference, where π_{behav} denotes the behavior policy and $\beta > 0$ is the regularization temperature (we use π_{behav} here to avoid clashing with our use of β throughout the rest of the paper for the KL-regularization coefficient). All of these adapt at training time; our setting holds the actor fixed and adapts only at deployment.

Transfer with changing reward. Successor features and generalized policy improvement handle transfer across reward weightings [Barreto et al., 2017, 2018], forward–backward representations learn at training time a universal representation from which policies for new rewards are obtained zero-shot [Touati and Ollivier, 2021, Touati et al., 2023], and goal-conditioned RL conditions policies or value functions on a goal at training time [Chane-Sane et al., 2021, Liu et al., 2022]. These require committing to a transfer-oriented pipeline before deployment; we instead start from a fixed, already-validated actor that was not trained for transfer and adapt only at deployment through a separate goal-conditioned prior.

Multiplicative composition of policies. Product-of-Experts combines distributions multiplicatively [Hinton, 2002]. In RL, energy-based max-entropy policies [Haarnoja et al., 2017] and composable deep RL [Haarnoja et al., 2018] use multiplicative composition of Gaussian policies with the closed-form precision-weighted mean and covariance we reuse; decoding-time composition of base and expert models has been studied for language generation [Liu et al., 2021]. We do not introduce PoE composition; the contribution is to make its identity with the KL-regularized frozen-actor update explicit and to use it to organize an empirical characterization.

3 Problem Setup

Let $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, r, \gamma)$ be a discounted MDP and let $g \in \mathcal{G}$ denote a deployment goal that modifies the effective reward via $r_g(s, a)$. In our D4RL experiments, $g \in \mathbb{R}^3$ weights three per-step reward components recovered from the environment reward:

$$r_g(s, a) = g \cdot rc(s, a), \qquad rc(s, a) = \big(r^{\text{fwd}}(s, a),\ c^{\text{ctrl}}(a),\ 1\big), \tag{1}$$

where r^{fwd} is the forward-progress reward, $c^{\text{ctrl}}(a) = k\|a\|^2 \geq 0$ is the magnitude of the environment's control cost ($k = 0.1$ for HalfCheetah, 10^{-3} for Hopper and Walker2d), and the last channel is the alive indicator; the MuJoCo reward is $r = r^{\text{fwd}} - c^{\text{ctrl}}$, plus the alive bonus for Hopper and Walker2d. The deployment objective is $J_g(\pi) = \mathbb{E}_{\tau\sim\pi}[\sum_t \gamma^t r_g(s_t, a_t)]$. We are given a fixed offline dataset $\mathcal{D} = \{(s_i, a_i, r_i, s'_i, d_i)\}_{i=1}^N$, pretrain an actor $\pi_\theta(a \mid s)$, and freeze its parameters; at deployment we allow no new interaction, no new data collection, and no actor retraining. The problem is to construct a deployment policy that remains conservative with respect to π_θ while adapting to the updated g. We use V_g^π, Q_g^π, A_g^π for goal-conditioned value/action-value/advantage, and $d^\pi(s) = (1-\gamma)\sum_t \gamma^t \Pr(s_t = s \mid \pi)$ for discounted state occupancy. *Scope:* we study goal shifts within this space of dataset-recoverable reward components and require the deployed policy to stay within the support of the validated actor; objectives that need behavior outside that support, such as reverse motion (Appendix O.5), are out of scope by design. Transition shift is treated structurally in Appendix A.

4 Deployment-Time Policy Refinement via Product-of-Experts

We adapt a frozen offline actor at deployment time by composing it with a goal-conditioned prior, modifying action selection without touching actor parameters.

4.1 PoE Refinement

Let $\pi_\theta(a \mid s)$ be the frozen actor and $\rho_\phi(a \mid s, g)$ a goal-conditioned deployment prior. The refined deployment policy is

$$\pi_{\text{ref}}(a \mid s, g) = \frac{\pi_\theta(a \mid s)^\alpha \rho_\phi(a \mid s, g)^{1-\alpha}}{\sum_{a' \in \mathcal{A}} \pi_\theta(a' \mid s)^\alpha \rho_\phi(a' \mid s, g)^{1-\alpha}}, \tag{2}$$

where $\alpha \in [0, 1]$ controls the adaptation–conservatism tradeoff: $\alpha \to 1$ recovers π_θ, and $\alpha \to 0$ recovers ρ_ϕ restricted to actor-supported actions. The form is actor-preserving in finite action spaces: $\pi_\theta(a \mid s) = 0$ implies $\pi_{\text{ref}}(a \mid s, g) = 0$ for every $\alpha > 0$.

4.2 Goal-Conditioned Prior

The prior maps (s, g) to a distribution over actions (or the parameters of a Gaussian density in continuous control), and is trained *only* on the same offline dataset $\mathcal{D}$ as the actor, with no deployment interaction. Per batch we (i) z-score and $\tanh$-rescale the reward-component vector $rc(s, a)$, (ii) sample a sharp goal g from a mixture of one-hot-like and Dirichlet draws, (iii) compute the goal-weighted score $g \cdot \widetilde{rc}(s_i, a_i)$, and (iv) partition the top-k and bottom-k transitions into (s^+, a^+) and (s^-, a^-). The prior is trained to minimize

$$\mathcal{L}_\rho(\phi) = -\mathbb{E}_{(s^+,a^+)}\big[\log \rho_\phi(a^+ \mid s^+, g)\big] + \lambda_{\text{con}}\, \mathbb{E}\Big[\big[m - (\log \rho_\phi(a^+ \mid s^+, g) - \log \rho_\phi(a^- \mid s^-, g))\big]_+\Big], \tag{3}$$

with defaults m=1, λ_{con}=1, top/bottom fraction 0.2 (full hyperparameters in Appendix P). The first term is goal-weighted BC on high-score transitions; the second pushes high-score actions to higher log-density than low-score actions *under the same sampled goal*, giving the prior a directional response to g. A pure BC prior would ignore g, and a pure ranking loss could drift off the data support; the combination keeps the prior on dataset actions, which the degradation analysis relies on, while making it goal-directional. The prior only reweights dataset actions and never queries a critic. Defaults were chosen from a small grid search; in our experience headline-package returns are qualitatively stable to nearby choices of $(m, \lambda_{\text{con}}, \text{top/bottom})$ and the sharp-vs-Dirichlet goal mixture, but a systematic sensitivity sweep is left to future work. The construction needs only a scalar deployment score per transition, not a privileged decomposition: without reward components, a hand-specified deployment reward, a learned reward model, or critic-derived scores (the critic-prior variant, Appendix O.13) can replace $g \cdot rc$. This keeps actor pretraining and deployment-time adaptation modular.

4.3 Continuous-Action Gaussian Form

For diagonal-Gaussian $\pi_\theta(\cdot \mid s) = \mathcal{N}(\mu_\theta, \Sigma_\theta)$ and $\rho_\phi(\cdot \mid s, g) = \mathcal{N}(\mu_\phi, \Sigma_\phi)$, the PoE-refined policy is Gaussian with precision and mean

$$\Sigma_{\text{ref}}^{-1} = \alpha\,\Sigma_\theta^{-1} + (1-\alpha)\,\Sigma_\phi^{-1}, \qquad \mu_{\text{ref}} = \Sigma_{\text{ref}}\big(\alpha\,\Sigma_\theta^{-1}\mu_\theta + (1-\alpha)\,\Sigma_\phi^{-1}\mu_\phi\big),$$

a closed-form precision-weighted combination of actor and prior.

Equivalence with KL-regularized adaptation. The KL-regularized update $\arg\min_\pi \mathbb{E}_\pi[-\log \rho_\phi] + \beta\,\mathrm{KL}(\pi\|\pi_\theta)$ reduces to PoE in four steps:

(i) $\pi^* \propto \pi_\theta\,\rho_\phi^{1/\beta}$, (ii) $\Lambda^{\text{KL-Reg}}(\beta) = \Sigma_\theta^{-1} + \frac{1}{\beta}\Sigma_\phi^{-1}$,

(iii) $\beta = \frac{\alpha}{1-\alpha} \Rightarrow \Lambda^{\text{KL-Reg}}(\beta) = \frac{1}{\alpha}\Lambda_{\text{ref}}^{\text{PoE}}(\alpha)$, (iv) $\mu^{\text{KL-Reg}} = \mu_{\text{ref}}$, $\Sigma_{\text{ref}}^{\text{PoE}} = \frac{1}{\alpha}\Sigma^{\text{KL-Reg}}$.

The natural parameters are scalar multiples, so the means coincide exactly and the covariances differ by the global scalar $1/\alpha = (1+\beta)/\beta$, which cancels under deterministic action selection. Appendix C audits the identity on dataset states and rollouts.

Notation. $\alpha \in [0, 1]$ is the actor weight in PoE; $\beta = \alpha/(1-\alpha) \in [0, \infty)$ is the KL weight on the frozen actor. The grid $\alpha \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$ corresponds to $\beta \in \{0.111, 0.429, 1, 2.333, 9\}$. $\alpha \to 1$ ($\beta \to \infty$) returns the frozen actor; $\alpha \to 0$ ($\beta \to 0$) returns the prior on actor support.

5 Theoretical Analysis

We include two theorems for structural interpretation only; main-body empirical claims rest on rollouts, not on these bounds. Theorem 1 is the KL-barycenter identity tying α to a weighted KL objective; Theorem 2 decomposes return into a one-step gain and a TV deviation penalty (loose at standard γ, see Appendix D). Both are stated in the finite-action setting; a Wasserstein continuous-action sketch and additional guarantees are in Appendix A.

5.1 Variational Characterization

PoE refinement is the unique maximizer of an entropy-regularized objective trading off attraction to actor and prior, on the effective common support $\Gamma_{s,g} = \{a : \pi_\theta(a \mid s) > 0,\ \rho_\phi(a \mid s, g) > 0\}$.

Theorem 1 (Variational optimality)**.** *Fix s, g and assume a finite action space. Optimization is restricted to distributions supported on $\Gamma_{s,g}$ (actions outside receive zero mass in any maximizer; proof in Appendix A.1). Then the unique maximizer of*

$$\mathcal{F}_{s,g}(\pi) = \mathbb{E}_{a\sim\pi}[\alpha \log \pi_\theta(a \mid s) + (1-\alpha)\log \rho_\phi(a \mid s, g)] + \mathcal{H}(\pi) \tag{4}$$

is the PoE policy in Eq. (2); equivalently $\pi_{\text{ref}} = \arg\min_\pi[\alpha\,\mathrm{KL}(\pi\|\pi_\theta) + (1-\alpha)\mathrm{KL}(\pi\|\rho_\phi)]$.

The Gaussian closed form of Sec. 4.3 is the precision-space solution, and the KL-regularized equivalence (Sec. 4.3) follows directly. In finite action spaces the identity preserves actor support for every $\alpha > 0$.

5.2 Conservative Improvement Relative to the Frozen Actor

Let $\bar{A}_g^{\pi_\theta}(s) = \mathbb{E}_{a\sim\pi_{\mathrm{ref}}(\cdot|s,g)}[A_g^{\pi_\theta}(s,a)]$ be the one-step refined-policy advantage under the frozen actor's value function, and $\delta_\pi = \sup_s \mathrm{TV}(\pi_{\mathrm{ref}}(\cdot \mid s,g), \pi_\theta(\cdot \mid s))$.

Theorem 2 (Conservative improvement decomposition)**.** *Assume bounded rewards. Then*

$$J_g(\pi_{\mathrm{ref}}) - J_g(\pi_\theta) \;\geq\; \tfrac{1}{1-\gamma}\,\mathbb{E}_{s\sim d^{\pi_\theta}}[\bar{A}_g^{\pi_\theta}(s)] - \frac{2\gamma}{(1-\gamma)^2}\,\varepsilon_A\,\delta_\pi, \quad \varepsilon_A = \sup_s |\bar{A}_g^{\pi_\theta}(s)|. \tag{5}$$

At $\gamma = 0.99$ the penalty coefficient $\frac{2\gamma}{(1-\gamma)^2} = 19{,}800$; with large worst-case total variation, the resulting lower bound can be vacuous. We therefore do not use Eq. (5) to predict rollout gains or classify cells. Its qualitative reading is that one-step gain must be weighed against policy deviation. HELP cells pair a useful gain with small deviation (Hopper medium G_2, Walker2d medium G_3); HURT cells pair a small gain with deviations that leave a narrow region of reliable behavior (the G_1 speed cells and medium-expert, Sec. 7.5); FROZEN cells have both terms small (HalfCheetah medium). This is an interpretation, not a theoretical explanation of the observed labels. Proofs appear in Appendix A.

6 Experimental Setup

We use D4RL continuous-control tasks [Fu et al., 2020]. The headline package covers HalfCheetah, Hopper, and Walker2d medium and Hopper medium-expert; the harder-regime package (Sec. 7.5, Appendix I) covers the medium-replay and medium-expert variants, plus AntMaze umaze and medium-play as competence diagnostics. For each environment, a diagonal-Gaussian actor is trained by behavioral cloning [Fujimoto et al., 2019] and frozen, a separate goal-conditioned Gaussian prior is trained with Eq. (3), and at deployment the two are composed in closed form (Sec. 4.3). All packages share the rollout and reporting protocol below.

Evaluation protocol. All main-body results are based on real MuJoCo simulator rollouts. We run five seeds and five episodes per seed, giving 25 episodes per method–setting–goal cell and 3,900 headline rollout episodes in total. All headline methods execute deterministic Gaussian mean actions; any stochastic-sampling result is reported only as a diagnostic and is not used for the main claims (Appendix B). Deployment goals weight the components of Eq. (1): $G_1 = (1, 0.1, 0.1)$ (speed), $G_2 = (0.5, 0.5, 0.5)$ (balanced), and $G_3 = (0.1, 1, 0.1)$ (control-weighted). Because c^{ctrl} enters with a positive weight, G_3 rewards rather than penalizes control effort; it serves as a third reward direction distinct from speed.[2] The same decomposition is used for prior training and rollouts, so the goal-weighted rollout return $\widehat{J}_g(\pi) = \sum_t g \cdot rc_t$ is aligned with the prior objective. We also report D4RL-normalized return and the rollout-state average $\mathrm{KL}(\pi(\cdot \mid s,g) \,\|\, \pi_\theta(\cdot \mid s))$.

Methods. We compare Frozen Actor, Prior Only, Additive Mix with $\lambda = 0.5$, KL-Reg, PoE, and critic-guided baselines. KL-Reg is evaluated at $\beta \in \{0.111, 0.429, 1.000, 2.333, 9.000\}$, and PoE is evaluated at $\alpha \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$ with the matched relation $\beta = \alpha/(1-\alpha)$. We also include an AWR/AWAC-style first-order critic-gradient update,

$$\mu_\theta(s) + \frac{\sigma_\theta^2(s)}{\beta}\nabla_a Q_g(s, \mu_\theta(s)),$$

swept over $\beta \in \{0.5, 1.0, 3.0\}$. CQL- and IQL-guided critic baselines are swept over the same β grid; details are reported in Appendices H and H.1.

Statistical reporting. For each cell, we report 95% bootstrap confidence intervals over the 25 rollout episodes, seed-matched paired differences with 95% bootstrap confidence intervals, and rliable-style probability of improvement [Agarwal et al., 2021], resampled from per-seed summaries. Throughout the main body, "return" refers to real MuJoCo rollout return rather than an off-policy diagnostic score.

[2] The submitted version called G_3 "efficient". That label was incorrect; the camera-ready renames it without changing any result.

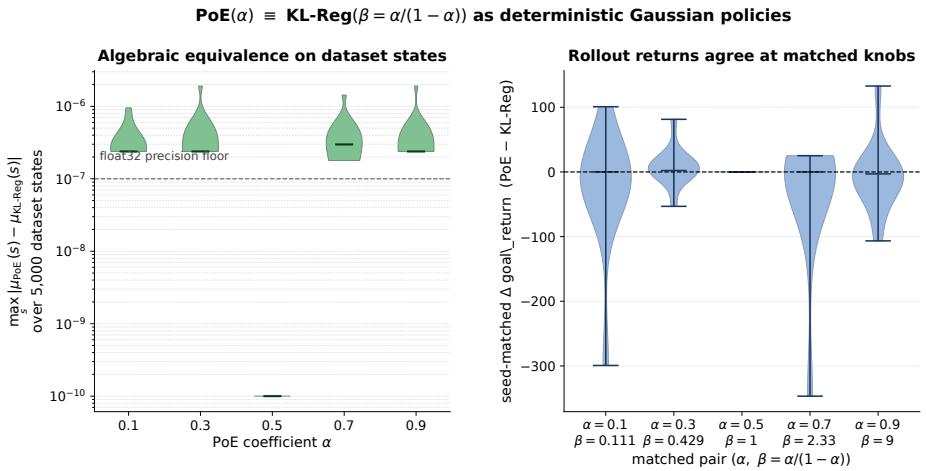

Figure 1: Empirical verification that $\mathrm{PoE}(\alpha)$ and $\mathrm{KL\text{-}Reg}(\beta = \alpha/(1-\alpha))$ are the same deterministic Gaussian policy. **Left**: per-state $|\mu_{\mathrm{PoE}} - \mu_{\mathrm{KL\text{-}Reg}}|$ over 5,000 dataset states per α, across four D4RL task settings and three deployment goals. Violin scale is logarithmic; the dashed line marks float-32 machine precision. **Right**: seed-matched rollout goal-weighted-return difference (PoE minus matched KL-Reg) at each matched α, pooled across all 12 setting-by-goal cells. The residual spread away from $\alpha = 0.5$ is MuJoCo chaotic amplification of float-precision action jitter, not a policy-level difference.

7 Results

We report three rollout findings (Figures 1, 2, and 3): (i) PoE and KL-Reg are the same deterministic policy under $\beta = \alpha/(1-\alpha)$; (ii) with a reliable prior, composition helps on some cells, ties the frozen actor on others, and hurts on the rest; (iii) under degraded priors, precision-weighted composition anchors to the frozen actor while additive and prior-only adaptation collapse, which is the strongest practical evidence in the paper.

7.1 Empirical Equivalence of PoE and KL-Regularized Steering

We empirically verify the identity in Sec. 4.3. With $\beta = \alpha/(1-\alpha)$, PoE and KL-Reg produce the same deterministic Gaussian mean; their covariance differs only by a global scalar and does not affect mean-action rollouts. Across 5,000 dataset states per environment and all $\alpha \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$, $|\mu_{\mathrm{PoE}} - \mu_{\mathrm{KL\text{-}Reg}}| \leq 2 \times 10^{-6}$. At $\alpha = 0.5, \beta = 1.0$, MuJoCo rollouts give bit-identical returns and episode lengths in all 12 cells; other matched settings have paired 95% CIs including zero. Thus, PoE and KL-Reg are a single actor-anchored steering family in our deterministic deployment setting. Figure 1 summarizes the audit; details are in Appendix C.

7.2 Headline Rollout Results

Figure 2 shows the original five-seed mean goal-weighted return with 95% bootstrap confidence intervals for each method family and each setting–goal cell. For Additive, KL-Reg, and PoE, we report the best operating point within each family according to goal-weighted return. Exact per-cell values are provided in Appendix O.11.

We label each cell with a seed-matched classifier (Appendix G): with ϵ the half-width of the frozen actor's 95% CI, a cell is HELP if the best composition mean exceeds the frozen-actor mean by more than ϵ, HURT if it falls below by more than ϵ, and FROZEN otherwise. The ten-seed replication gives 4/4/4 (five seeds: 4/5/3). HalfCheetah medium is FROZEN on all three goals. In Hopper medium G_2, Hopper medium-expert G_2 and G_3, and Walker2d medium G_3, the best composition method improves over the frozen actor by $+144$, $+55$, $+110$, and $+55$ (HELP). The three speed-dominant G_1 cells are HURT: the frozen actor outperforms every composition method by 291, 878, and 2,030. With ten seeds, Hopper medium G_3 moves from FROZEN to a small HURT ($\Delta = -11.0$ versus $\epsilon = 6.3$). These rounded deltas are five-seed values; ten-seed values are in Appendix E. With this

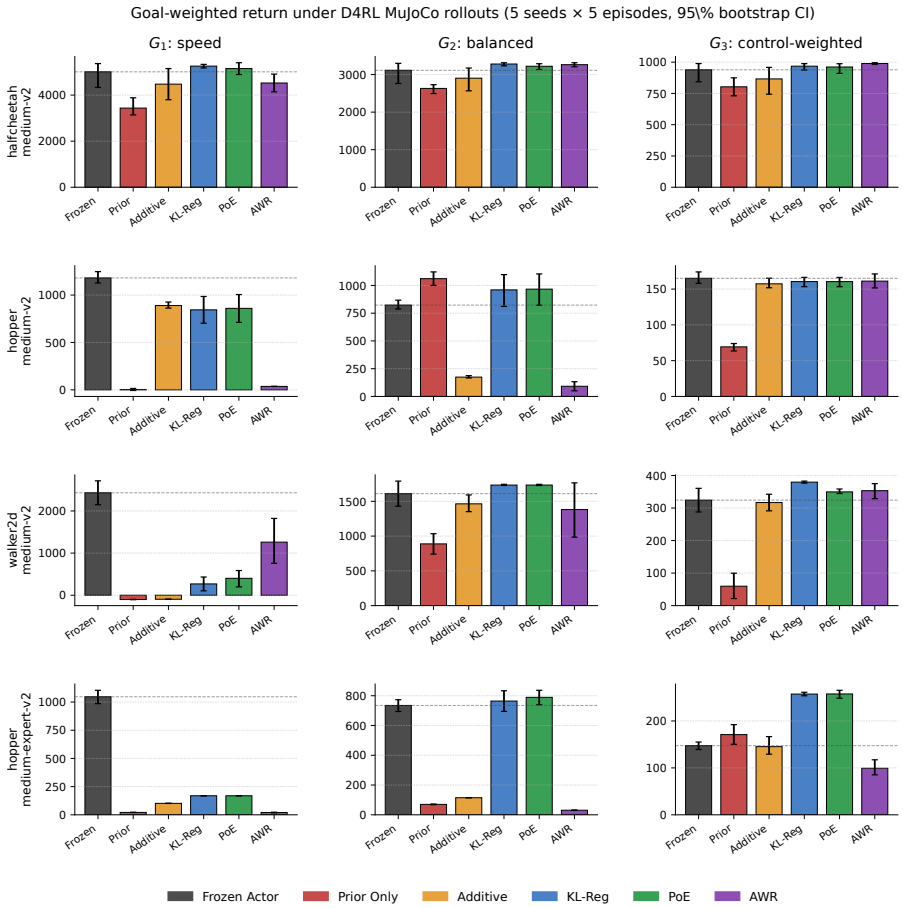

Figure 2: Headline D4RL MuJoCo rollout comparison. Bars show mean returns, and error bars show 95% bootstrap confidence intervals over 5 seeds and 5 episodes per seed, giving 25 episodes per cell. Rows denote environments and columns denote deployment goals. For each composition family, the plotted result is the best-in-family operating point based on goal-weighted return.

prior, redirecting an actor near its velocity ceiling destabilizes it; with an AWR-trained prior these three cells are not HURT (Sec. 7.4).

7.3 Critic-Gradient (First-Order) and Critic-Guided Baselines

A first-order critic-gradient baseline in the style of AWR/AWAC replaces the refinement prior with a step along $\nabla_a Q_g(s, \mu_\theta(s))$ (Appendix H). It is not AWR proper, which refits the policy by weighted regression and thus retrains the actor; it isolates whether a raw critic gradient can substitute for a density anchor at deployment. It improves over Frozen in only 3 of 12 cells (HalfCheetah G_2, G_3, Walker2d G_3), falls more than 100 return points below Frozen in 7, and collapses close to Prior Only in 5 Hopper and Walker2d G_1 cases. The fitted-Q signal is useful mainly near the data support: once the step moves $\mu_\theta(s)$ off the support ridge, Hopper- and Walker2d-style agents often early-terminate. This agrees with Park et al. [2024], who find gradient-based policy extraction effective when it is behavior-constrained; density-anchored composition supplies such an anchor without retraining. CQL- and IQL-guided updates swept over $\beta \in \{0.5, 1.0, 3.0\}$ (Appendix H.1) obtain, at the best β per cell, HELP/FROZEN/HURT splits of $5/7/0$ and $6/6/0$ against $4/5/3$ for PoE/KL-Reg (all five-seed; PoE/KL-Reg is $4/4/4$ with ten seeds). Critic-guided methods thus avoid the three large HURT cases but add only one or two HELP cells, require auxiliary critic training, and do not inherit the closed-form equivalence (Sec. 4.3) or the prior-degradation robustness (Sec. 7.4).

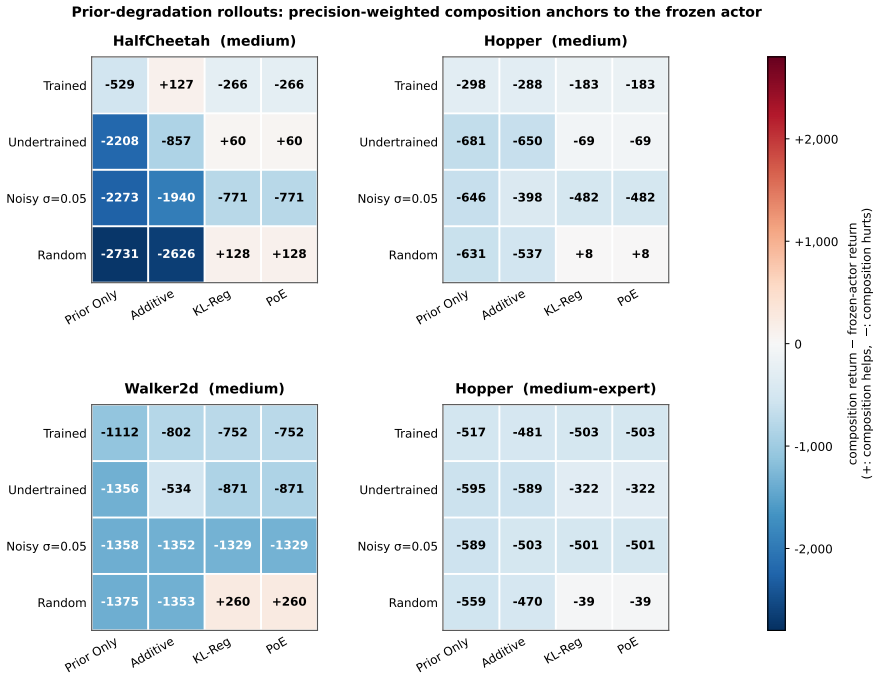

Figure 3: Prior-degradation rollouts. Each cell is the mean goal-weighted return of a composition family minus the frozen-actor return on the same environment, averaged across goals. Red (positive): composition helps. Blue (negative): composition hurts. All cells use a single fixed operating point, PoE at $\alpha = 0.5$ (matched KL-Reg at $\beta = 1.0$) and Additive at $\lambda = 0.5$, rather than a per-cell best α. PoE and matched KL-Reg are the same Gaussian update (Sec. 7.1). A random prior can sit above the trained prior because it is ignored, not because it is useful: a diffuse prior has low precision, so the precision-weighted posterior stays on the frozen actor (in one dimension with $\alpha = 0.5$ and actor $\mathcal{N}(0, 1)$, a prior $\mathcal{N}(1, 0.1^2)$ moves the mean to 0.99 but a prior $\mathcal{N}(1, 10^2)$ only to 0.0099; in our rollouts the mean KL to the frozen actor is 0.61 under the random prior versus 10.4 under the trained prior), whereas a confident trained prior moves the action and pays the steering cost on goals where the actor is near its ceiling. The informative comparison is within a row: under the same random prior, Prior Only and Additive collapse while PoE/KL-Reg do not. Per-goal values: Fig. 5 (Appendix).

7.4 Robustness to Degraded Deployment Priors

We evaluate the four task settings and three goals under four prior variants (the trained checkpoint, *undertrained*, *noisy* with Gaussian parameter noise at $\sigma = 0.05$, and *random*) with three seeds and three episodes per cell. Figure 3 reports the gap in goal-weighted return relative to the frozen actor, averaged across goals.

Additive and prior-only deployment collapse as the prior degrades, while precision-weighted composition stays near the frozen actor. Under *random* prior: Prior Only 86.3, Additive 164.2, Frozen 1,410.5, PoE/KL-Reg 1,499.8; under *noisy* ($\sigma = 0.05$): 193.8/362.0/–/639.6; under *undertrained*: 200.4/753.3/–/1,110.0 (per-cell numbers: Appendix O.6). A degraded prior has low precision, so the posterior mean shrinks toward the frozen actor; additive mixing carries damaged means through. Precision-weighted composition is an actor-anchored *safety layer* under untrustworthy priors, not a return-maximization rule. To choose α without hindsight, a KL-budget rule (Appendix O.4) selects the strongest steering, i.e., the smallest α, whose mean dataset-state KL from the frozen actor stays within a budget κ; it needs only the dataset, the actor, and the prior, and no rollouts. On `halfcheetah-medium-v2`, scored with the off-policy diagnostic evaluator rather than MuJoCo rollouts, the selection loss relative to the hindsight-best α is at most 0.020 across $\kappa \in \{0.1, 0.3, 0.5, 1.0\}$ and all three goals, shrinks as the budget loosens, and is zero at $\kappa = 1.0$, where the selected α coincides with the oracle. Tighter budgets select more conservative operating points (e.g., $\alpha = 0.8$

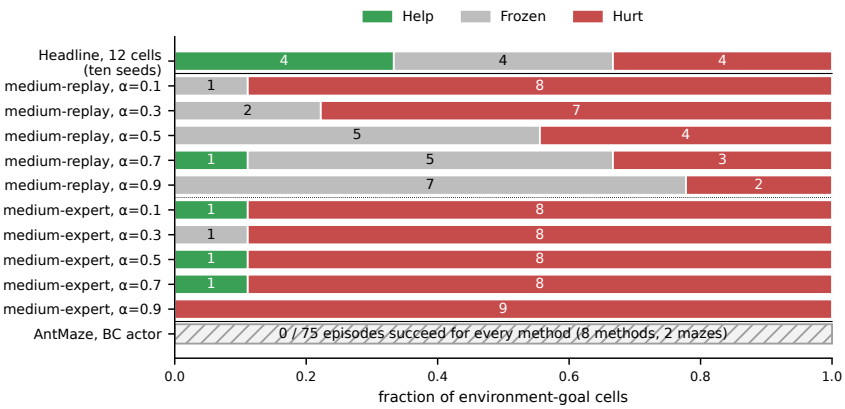

Figure 4: Where steering helps. HELP/FROZEN/HURT verdicts (CI-half-width classifier, Appendix G) for the 12 headline cells (ten seeds) and for every harder-regime cell–α combination (9 cells per row); bar labels are cell counts (Table 7). Medium-replay moves toward FROZEN as α grows, medium-expert stays HURT at every α, and with a zero-success BC actor every AntMaze method stays at zero.

versus an oracle of 0.05 at $\kappa = 0.1$). The budget κ is therefore a deployment design choice that trades goal-directed gain for proximity to the validated actor, not a quantity tuned to the oracle.

Prior construction. With a goal-conditioned prior trained by advantage-weighted regression on dataset actions instead of Eq. (3) (Appendix M), the headline split changes from 4/5/3 to 4/8/0: no cell is HURT, and Hopper medium-expert G_1 moves from -878 to $+117$, although Prior Only with that prior still collapses there. The HURT cells thus reflect a near-ceiling actor combined with the contrastive prior, not the actor alone.

7.5 Harder Regimes and the Actor-Competence Ceiling

To examine whether the HELP/FROZEN/HURT boundary depends on regime difficulty or α, we swept $\alpha \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$ across six harder locomotion settings: the medium-replay and medium-expert variants of HalfCheetah, Hopper, and Walker2d, totaling $5{,}850$ episodes. Equivalence persists: per-cell $|\mu_{\text{PoE}} - \mu_{\text{KL-Reg}}|$ is exactly 0 at $\alpha = 0.5$ in all 18 cells. Figure 4 summarizes the verdicts (counts in Table 7, Appendix J). Medium-replay remains α-responsive, with 7 of 9 cells becoming Frozen ties at $\alpha = 0.9$. Medium-expert is largely insensitive: 8 of 9 cells remain HURT for $\alpha \in \{0.1, 0.3, 0.5, 0.7\}$, and all 9 are HURT at $\alpha = 0.9$. A representative case is Walker2d medium-expert, where G_1 Frozen reaches 4920, while PoE collapses to $\{399, 72, 62, 63, 78\}$ across the tested α values. We interpret this as an *actor-competence ceiling*. When the frozen actor already lies near the locomotion ceiling, even small precision-weighted shifts can move the deployed action off the high-support ridge. Thus, increasing α does not provide reliable steering headroom; with the contrastive prior, the failure is driven by the geometry of the medium-expert support ridge rather than by the specific knob value. Per-cell values are reported in Appendix J.

AntMaze, graded competence, and Adroit. On AntMaze umaze and medium-play, frozen BC actors that never reach the goal (the umaze actor was trained on a 2% data subsample) solve 0 of 75 episodes, and every deployment method inherits this zero (Appendix K). With six umaze actors of 39–67% success, PoE/KL-Reg retain 24–69% success while Prior Only collapses to 0%, but composition never improves on the frozen actor (FROZEN for three actors, HURT for three); contrary to the monotone trend we had anticipated, its effect becomes more negative as actor success rises. On Adroit (Appendix L), composition preserves a moderately competent pen actor (FROZEN on all goals) while prior-only and additive steering are HURT, and no method succeeds with an incompetent hammer actor. An SF/GPI baseline [Barreto et al., 2017, 2018] is competitive on halfcheetah but fragile on hopper and walker2d (Appendix N).

8 Discussion

Three readings summarize the evidence: PoE and KL-Reg are one rule under $\beta = \alpha/(1-\alpha)$; the central practical value is *graceful degradation* under degraded priors; and the regime boundary is set by actor competence rather than by α (Fig. 4), with AntMaze as its extreme; for near-ceiling actors it also depends on how the prior is trained. Critic-guided baselines (AWR; CQL/IQL swept over β) and SF/GPI sharpen this boundary rather than overturn it: at their best β, CQL/IQL never HURT on the 12 main cells but HELP on only 5–6, and they do not inherit the closed-form equivalence or the prior-degradation robustness. When a reliable critic is affordable, validated, and its β can be tuned, CQL-guided steering is a strong option. We do not recommend it as the default for governed deployment: its best β was chosen per cell from realized returns and we know of no rollout-free selector for it; single-β operation is fragile (IQL at $\beta = 1.0$ collapses to about 5 return on all three harder-regime hopper-medium-expert cells, Appendix I); it inherits critic error directly (Critic-Greedy saturates both instability metrics under critic corruption, Appendix O.14); and a critic in the action path is heavier to revalidate than a one-knob reweighting.

When does steering fail? A dataset-only diagnostic. The actor-competence ceiling is empirical rather than a universal guarantee. We evaluated a dataset-state sensitivity score S, the mean KL divergence between the composed policy at $\alpha = 0.5$ and the frozen actor, computed in closed form without rollouts. On the 12 main cells, the three five-seed HURT cells score highest (minimum HURT 5.06; maximum non-HURT 4.96; Spearman -0.77, $p = 0.003$, between S and the return deltas). This suggests a procedure that uses only the dataset, actor, and prior: abstain when S exceeds a prespecified threshold, otherwise steer within a dataset-state KL budget, and retrain if the actor itself is incompetent. S measures how far steering moves a validated actor, not return; a large deviation could help a poor actor, although AntMaze shows reweighting cannot rescue such an actor. The separation is retrospective and imperfect: with ten-seed labels Hopper medium G_3 ($S = 1.57$) becomes HURT, and on the 18 harder-regime cells (Appendix F) S tracks the deltas (Spearman -0.64, $p = 0.005$; pooled over all 30 cells -0.67, $p < 10^{-4}$) and scores the three medium-expert G_1 HURT cells highest, but HURT and non-HURT cells overlap (5 of the 8 HURT cells rank in the top 8). S therefore ranks steering risk but gives no threshold that transfers across regimes. Three further dataset-only indicators are consistent with the boundary: actor–prior log-density mismatch on dataset states, the actor's normalized-return position (saturation), and near-zero offline success (AntMaze).

9 Conclusion

We studied deployment-time adaptation of frozen offline RL actors when objectives change but retraining is infeasible. For diagonal-Gaussian actors and priors, PoE refinement and KL-regularized adaptation are one actor-anchored rule with α as a conservatism knob. The rule degrades gracefully rather than improving universally: it stays anchored under degraded priors where additive and prior-only adaptation collapse, it can preserve or improve a reliable actor but cannot recover a weak one, and whether it helps or hurts a near-ceiling actor depends on the prior. PoE/KL-Reg is therefore best viewed as an actor-anchored safety layer for objective shift, not a universal return-maximization method.

Limitations. The closed form and the equivalence assume diagonal-Gaussian actors and priors; for mixture or diffusion policies the product has no closed form, so extending the policy class changes the method itself, not only the analysis. The formal guarantees are mainly finite-action, with continuous-action sketches in Appendix A.10. The PoE $\equiv$ KL-Reg identity holds at the deterministic mean used in our rollouts; under stochastic sampling the two rules have different posterior covariances and therefore different stochastic KLs. Composition hurts near-ceiling actors (3 of 12 main cells with five seeds, 4 with ten; most medium-expert cells at every α). The ten-seed replication covers the headline package only; the harder-regime package uses five seeds. The KL-budget selector is evaluated with the off-policy diagnostic evaluator, and the dataset-only sensitivity score was validated retrospectively on 12 cells. Evaluation covers D4RL locomotion, AntMaze, and two Adroit tasks, all with BC-trained actors and linear reward-component goal shifts; the graded-competence and Adroit studies use one fixed prior per task, the AWR-prior comparison covers only the 12 headline cells, and the AntMaze umaze prior was trained on a 2% data subsample. Future work should address richer policy classes, stochastic rollouts, and calibrated pre-deployment diagnostics.

Acknowledgments and Disclosures

Funding. This research received no specific grant funding from public, commercial, or not-for-profit funding agencies.

Competing Interests. The authors declare no competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron C Courville, and Marc Bellemare. Deep reinforcement learning at the edge of the statistical precipice. *Advances in neural information processing systems*, 34:29304–29320, 2021.

André Barreto, Will Dabney, Rémi Munos, Jonathan J Hunt, Tom Schaul, Hado P Van Hasselt, and David Silver. Successor features for transfer in reinforcement learning. *Advances in neural information processing systems*, 30, 2017.

Andre Barreto, Diana Borsa, John Quan, Tom Schaul, David Silver, Matteo Hessel, Daniel Mankowitz, Augustin Zidek, and Remi Munos. Transfer in deep reinforcement learning using successor features and generalised policy improvement. In *International Conference on Machine Learning*, pages 501–510. PMLR, 2018.

Elliot Chane-Sane, Cordelia Schmid, and Ivan Laptev. Goal-conditioned reinforcement learning with imagined subgoals. In *International conference on machine learning*, pages 1430–1440. PMLR, 2021.

Liting Chen, Jie Yan, Zhengdao Shao, Lu Wang, Qingwei Lin, Saravanakumar Rajmohan, Thomas Moscibroda, and Dongmei Zhang. Conservative state value estimation for offline reinforcement learning. *Advances in Neural Information Processing Systems*, 36:35064–35083, 2023.

Abhilash Reddy Chenreddy and Erick Delage. Epistemic robust offline reinforcement learning. *arXiv preprint arXiv:2604.07072*, 2026.

Justin Fu, Aviral Kumar, Ofir Nachum, George Tucker, and Sergey Levine. D4RL: Datasets for deep data-driven reinforcement learning. *arXiv preprint arXiv:2004.07219*, 2020.

Scott Fujimoto, David Meger, and Doina Precup. Off-policy deep reinforcement learning without exploration. In *International conference on machine learning*, pages 2052–2062. PMLR, 2019.

Jiayi Guan, Li Shen, Ao Zhou, Lusong Li, Han Hu, Xiaodong He, Guang Chen, and Changjun Jiang. PoCE: Primal policy optimization with conservative estimation for multi-constraint offline reinforcement learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26243–26253, 2024.

Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. Reinforcement learning with deep energy-based policies. In *International conference on machine learning*, pages 1352–1361. PMLR, 2017.

Tuomas Haarnoja, Vitchyr Pong, Aurick Zhou, Murtaza Dalal, Pieter Abbeel, and Sergey Levine. Composable deep reinforcement learning for robotic manipulation. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 6244–6251. IEEE, 2018.

Geoffrey E Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002.

Ilya Kostrikov, Ashvin Nair, and Sergey Levine. Offline reinforcement learning with implicit Q-learning. *arXiv preprint arXiv:2110.06169*, 2021.

Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative Q-learning for offline reinforcement learning. *Advances in neural information processing systems*, 33:1179–1191, 2020.

Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. 2020. URL https://arxiv.org/abs/2005.01643.

Hao Li, Xiao-Hu Zhou, Xiao-Liang Xie, Shi-Qi Liu, Zhen-Qiu Feng, Xiao-Yin Liu, Mei-Jiang Gui, Tian-Yu Xiang, De-Xing Huang, Bo-Xian Yao, et al. Crop: Conservative reward for model-based offline policy optimization. *arXiv preprint arXiv:2310.17245*, 2023.

Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A Smith, and Yejin Choi. DExperts: Decoding-time controlled text generation with experts and anti-experts. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6691–6706, 2021.

Minghuan Liu, Menghui Zhu, and Weinan Zhang. Goal-conditioned reinforcement learning: Problems and solutions. *arXiv preprint arXiv:2201.08299*, 2022.

Ashvin Nair, Abhishek Gupta, Murtaza Dalal, and Sergey Levine. AWAC: Accelerating online reinforcement learning with offline datasets. *arXiv preprint arXiv:2006.09359*, 2020.

Jongchan Park, Mingyu Park, and Donghwan Lee. Pretraining a shared q-network for data-efficient offline reinforcement learning. *arXiv preprint arXiv:2505.05701*, 2025.

Seohong Park, Kevin Frans, Sergey Levine, and Aviral Kumar. Is value learning really the main bottleneck in offline RL? In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Xue Bin Peng, Aviral Kumar, Grace Zhang, and Sergey Levine. Advantage-weighted regression: Simple and scalable off-policy reinforcement learning. *arXiv preprint arXiv:1910.00177*, 2019.

Emanuel Todorov, Tom Erez, and Yuval Tassa. MuJoCo: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pages 5026–5033. IEEE, 2012.

Ahmed Touati and Yann Ollivier. Learning one representation to optimize all rewards. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

Ahmed Touati, Jérémy Rapin, and Yann Ollivier. Does zero-shot reinforcement learning exist? In *International Conference on Learning Representations*, 2023.

Max Welling. Product of experts. *Scholarpedia*, 2(10):3879, 2007.

Songyuan Zhang, Oswin So, HM Ahmad, Eric Yang Yu, Matthew Cleaveland, Mitchell Black, and Chuchu Fan. Reform: Reflected flows for on-support offline rl via noise manipulation. *arXiv preprint arXiv:2602.05051*, 2026.

Contents

1 Introduction **1**

2 Related Work **2**

3 Problem Setup **3**

4 Deployment-Time Policy Refinement via Product-of-Experts **3**

4.1 PoE Refinement 3

4.2 Goal-Conditioned Prior 3

4.3 Continuous-Action Gaussian Form 4

5 Theoretical Analysis **4**

5.1 Variational Characterization 4

5.2 Conservative Improvement Relative to the Frozen Actor 5

6 Experimental Setup **5**

7 Results **6**

7.1 Empirical Equivalence of PoE and KL-Regularized Steering 6

7.2 Headline Rollout Results 6

7.3 Critic-Gradient (First-Order) and Critic-Guided Baselines 7

7.4 Robustness to Degraded Deployment Priors 8

7.5 Harder Regimes and the Actor-Competence Ceiling 9

8 Discussion **10**

9 Conclusion **10**

Supplementary Material **13**

A Theoretical Analysis (Extended) **16**

A.1 Proof of Theorem 1 16

A.2 Proof of Proposition 1 17

A.3 Auxiliary Lemmas for Policy Improvement 17

A.4 Proof of Theorem 2 18

A.5 Proof of Corollary 1 18

A.6 Auxiliary Lemma for Transition Shift 18

A.7 Proof of Theorem 3 19

A.8 Proof of Corollary 2 19

A.9 Proof of Corollary 3 20

A.10 Sketch for Proposition 2 and Corollary 4 20

B Evaluation Protocol and Metrics (Diagnostic Only) **21**
B.1 Protocol and Evaluators . . . 22
B.2 Diagnostic Metrics . . . 22
B.3 Goal-Conditioned Critic Training . . . 22
B.4 Policy Construction Versus Evaluation . . . 22

C Equivalence Audit: PoE(α) $\equiv$ KL-Reg($\beta = \alpha/(1-\alpha)$) **23**

D Conservative-Improvement Bound: Empirical Diagnostic **23**

E Ten-Seed Replication and Verdict Sensitivity **24**

F Dataset-Only Sensitivity Score on the Harder Regimes **24**

G Help / Frozen / Hurt Boundary Classification **24**

H First-Order Critic-Gradient Baseline (AWR/AWAC-Style) **25**
H.1 CQL/IQL-Guided Critic Baselines . . . 26

I Harder-Regime Rollout Package **27**

J Harder-Regime α Sweep **29**

K AntMaze Diagnostic **30**

L Adroit Manipulation **31**

M AWR-Trained Prior **32**

N Successor-Feature / GPI Baseline **33**

O Additional Results **34**
O.1 Note on Adaptation Efficiency . . . 34
O.2 Multi-Environment Goal Responsiveness . . . 34
O.3 Exact Pareto Operating Points . . . 34
O.4 Adaptive-α Selection . . . 35
O.5 Support Mismatch and Contradictory Goals . . . 35
O.6 Robustness to Deployment-Prior Degradation . . . 35
O.7 Actor-Quality Sensitivity . . . 36
O.8 Extended Baseline Comparison under the Diagnostic Evaluator . . . 36
O.9 Extended Distribution Shift Experiments . . . 38
O.10 Validated Multi-Environment Exact Comparisons . . . 38
O.11 Detailed Single-Environment D4RL Results . . . 38
O.12 Additional Multi-Environment D4RL Comparison Tables . . . 40
O.13 Extended Ablation Studies . . . 41

O.14 Policy Stability under Critic Degradation 41
O.15 Support Preservation Diagnostic 42
O.16 Q-value Analysis of Selected Actions 43
O.17 Additional Benchmark Analysis on Frozen-Actor Goal Adaptation 43

P Implementation Details and Reproducibility **44**
P.1 Training and Evaluation Setup 44
P.2 Inference-Time Procedure 44
P.3 Reproducibility Manifest 44

Q AI Usage Statement **45**

A Theoretical Analysis (Extended)

A.1 Proof of Theorem 1

Fix a state s and goal g, and abbreviate

$$p_a := \pi(a \mid s), \quad u_a := \pi_\theta(a \mid s), \quad v_a := \rho_\phi(a \mid s, g).$$

Because logarithms are only defined on positive mass, we work on the effective support

$$\Gamma_{s,g} := \{a \in \mathcal{A} : \ u_a > 0, \ v_a > 0\}.$$

For actions outside $\Gamma_{s,g}$, the weighted KL objective is either infinite or the Product-of-Experts numerator is zero, so such actions receive zero mass in the optimizer. It is therefore sufficient to optimize over probability vectors supported on $\Gamma_{s,g}$.

The per-state objective is

$$\mathcal{F}_{s,g}(p) = \sum_{a \in \Gamma_{s,g}} p_a\big(\alpha \log u_a + (1-\alpha)\log v_a\big) - \sum_{a \in \Gamma_{s,g}} p_a \log p_a,$$

subject to $p_a \geq 0$ and $\sum_{a \in \Gamma_{s,g}} p_a = 1$.

Introduce a Lagrange multiplier λ for the simplex constraint:

$$\mathcal{L}(p, \lambda) = \sum_{a \in \Gamma_{s,g}} p_a\big(\alpha \log u_a + (1-\alpha)\log v_a\big) - \sum_{a \in \Gamma_{s,g}} p_a \log p_a + \lambda \left(\sum_{a \in \Gamma_{s,g}} p_a - 1 \right).$$

Differentiating with respect to p_a gives

$$\frac{\partial \mathcal{L}}{\partial p_a} = \alpha \log u_a + (1-\alpha)\log v_a - (\log p_a + 1) + \lambda.$$

Setting the derivative to zero yields

$$\log p_a = \alpha \log u_a + (1-\alpha)\log v_a + \lambda - 1.$$

Exponentiating,

$$p_a = \exp(\lambda - 1)\, u_a^\alpha v_a^{1-\alpha}.$$

Because $\sum_{a \in \Gamma_{s,g}} p_a = 1$, the normalizing constant is

$$\exp(\lambda - 1) = \left(\sum_{a' \in \Gamma_{s,g}} u_{a'}^\alpha v_{a'}^{1-\alpha} \right)^{-1}.$$

Therefore,

$$p_a = \frac{u_a^\alpha v_a^{1-\alpha}}{\sum_{a' \in \Gamma_{s,g}} u_{a'}^\alpha v_{a'}^{1-\alpha}},$$

which is exactly the Product-of-Experts policy on the effective support, with zero mass outside that support:

$$\pi_{\text{ref}}(a \mid s, g) = \frac{\pi_\theta(a \mid s)^\alpha \rho_\phi(a \mid s, g)^{1-\alpha}}{\sum_{a'} \pi_\theta(a' \mid s)^\alpha \rho_\phi(a' \mid s, g)^{1-\alpha}}.$$

To prove the equivalent KL form, consider

$$\alpha\, \mathrm{KL}(\pi \| \pi_\theta) + (1-\alpha)\, \mathrm{KL}(\pi \| \rho_\phi).$$

Expanding both KL terms over $\Gamma_{s,g}$ yields

$$\begin{aligned} &\alpha \sum_{a \in \Gamma_{s,g}} \pi(a \mid s) \log \frac{\pi(a \mid s)}{\pi_\theta(a \mid s)} + (1-\alpha) \sum_{a \in \Gamma_{s,g}} \pi(a \mid s) \log \frac{\pi(a \mid s)}{\rho_\phi(a \mid s, g)} \\ &= \sum_{a \in \Gamma_{s,g}} \pi(a \mid s) \log \pi(a \mid s) - \sum_{a \in \Gamma_{s,g}} \pi(a \mid s)\big(\alpha \log \pi_\theta(a \mid s) + (1-\alpha)\log \rho_\phi(a \mid s, g)\big). \end{aligned}$$

This is exactly $-\mathcal{F}_{s,g}(\pi)$ up to the simplex constraint. Hence maximizing $\mathcal{F}_{s,g}$ is equivalent to minimizing the weighted KL objective. Uniqueness follows from strict concavity of entropy on the simplex. □

Proposition 1 (Interpolation and support preservation)**.** *Fix a state s and goal g, and recall the effective common support $\Gamma_{s,g} = \{a : \pi_\theta(a \mid s) > 0,\ \rho_\phi(a \mid s, g) > 0\}$. For every $\alpha > 0$, $\pi_{\mathrm{ref}}(a \mid s, g) = 0$ whenever $\pi_\theta(a \mid s) = 0$, so PoE refinement preserves actor support. Moreover, restricted to $\Gamma_{s,g}$:*

1. *As $\alpha \to 1$, $\pi_{\mathrm{ref}}(\cdot \mid s, g)$ converges to $\pi_\theta(\cdot \mid s)$ on $\Gamma_{s,g}$. If additionally $\mathrm{supp}(\pi_\theta(\cdot \mid s)) \subseteq \mathrm{supp}(\rho_\phi(\cdot \mid s, g))$, the limit recovers $\pi_\theta(\cdot \mid s)$ everywhere.*
2. *As $\alpha \to 0$, $\pi_{\mathrm{ref}}(\cdot \mid s, g)$ converges to $\rho_\phi(\cdot \mid s, g)$ restricted to $\mathrm{supp}(\pi_\theta(\cdot \mid s))$ and renormalized; this reduces to $\rho_\phi(\cdot \mid s, g)$ when $\mathrm{supp}(\rho_\phi(\cdot \mid s, g)) \subseteq \mathrm{supp}(\pi_\theta(\cdot \mid s))$.*

For diagonal-Gaussian actor and prior the supports coincide with $\mathbb{R}^{d_a}$, so both inclusion conditions hold automatically.

A.2 Proof of Proposition 1

Recall that

$$\pi_{\mathrm{ref}}(a \mid s, g) \propto \pi_\theta(a \mid s)^\alpha \rho_\phi(a \mid s, g)^{1-\alpha}.$$

As $\alpha \to 1$, the exponent on ρ_ϕ tends to zero. For every action $a \in \Gamma_{s,g}$ (where both $\pi_\theta(a \mid s) > 0$ and $\rho_\phi(a \mid s, g) > 0$),

$$\pi_\theta(a \mid s)^\alpha \rho_\phi(a \mid s, g)^{1-\alpha} \to \pi_\theta(a \mid s).$$

After normalization, the refined policy converges to $\pi_\theta(\cdot \mid s)$ restricted to $\Gamma_{s,g}$ and renormalized. If additionally $\mathrm{supp}(\pi_\theta(\cdot \mid s)) \subseteq \mathrm{supp}(\rho_\phi(\cdot \mid s, g))$ (so that every actor-supported action lies in $\Gamma_{s,g}$ and the renormalization is trivial), then

$$\lim_{\alpha \to 1} \pi_{\mathrm{ref}}(\cdot \mid s, g) = \pi_\theta(\cdot \mid s).$$

Without this inclusion the limit may differ from π_θ at actions a with $\pi_\theta(a \mid s) > 0$ but $\rho_\phi(a \mid s, g) = 0$, where $\rho_\phi(a \mid s, g)^{1-\alpha} = 0$ for every $\alpha \in (0, 1)$.

As $\alpha \to 0$, support preservation still holds for every $\alpha > 0$. Thus the limit is the prior restricted to actor support:

$$\lim_{\alpha \to 0} \pi_{\mathrm{ref}}(a \mid s, g) = \frac{\rho_\phi(a \mid s, g)\, \mathbf{1}\{\pi_\theta(a \mid s) > 0\}}{\sum_{a'} \rho_\phi(a' \mid s, g)\, \mathbf{1}\{\pi_\theta(a' \mid s) > 0\}}.$$

In the special case where

$$\mathrm{supp}(\rho_\phi(\cdot \mid s, g)) \subseteq \mathrm{supp}(\pi_\theta(\cdot \mid s)),$$

this reduces to

$$\lim_{\alpha \to 0} \pi_{\mathrm{ref}}(\cdot \mid s, g) = \rho_\phi(\cdot \mid s, g).$$

For support preservation, if $\pi_\theta(a \mid s) = 0$, then

$$\pi_\theta(a \mid s)^\alpha = 0 \qquad \text{for every } \alpha > 0,$$

so the numerator in Eq. (2) is zero and therefore

$$\pi_{\mathrm{ref}}(a \mid s, g) = 0.$$

□

A.3 Auxiliary Lemmas for Policy Improvement

We first state two standard ingredients.

Lemma 1 (Performance difference lemma)**.** *For any two stationary policies π and π' and deployment goal g,*

$$J_g(\pi') - J_g(\pi) = \frac{1}{1-\gamma} \mathbb{E}_{s \sim d^{\pi'}} \mathbb{E}_{a \sim \pi'(\cdot \mid s)} \left[A_g^\pi(s, a) \right].$$

Proof. This is the standard performance difference lemma specialized to the goal-indexed reward r_g. It follows by telescoping the Bellman equations for V_g^π along trajectories generated by π'. □ □

Lemma 2 (Occupancy difference bound)**.** *Let*

$$\delta_\pi = \sup_s \mathrm{TV}(\pi'(\cdot \mid s), \pi(\cdot \mid s)).$$

Then

$$\|d^{\pi'} - d^{\pi}\|_1 \le \frac{2\gamma}{1-\gamma}\,\delta_\pi.$$

Proof. This is a standard discounted occupancy perturbation bound obtained from the fixed-point equations for discounted visitation together with contraction and kernel perturbation arguments. □ □

A.4 Proof of Theorem 2

Starting from Lemma 1,

$$J_g(\pi_{\mathrm{ref}}) - J_g(\pi_\theta) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi_{\mathrm{ref}}}}\left[\bar{A}_g^{\pi_\theta}(s)\right].$$

Add and subtract the same quantity under d^{π_θ}:

$$\begin{aligned} J_g(\pi_{\mathrm{ref}}) - J_g(\pi_\theta) = \frac{1}{1-\gamma}&\mathbb{E}_{s\sim d^{\pi_\theta}}\left[\bar{A}_g^{\pi_\theta}(s)\right] \\ &+ \frac{1}{1-\gamma}\left(\mathbb{E}_{s\sim d^{\pi_{\mathrm{ref}}}}\left[\bar{A}_g^{\pi_\theta}(s)\right] - \mathbb{E}_{s\sim d^{\pi_\theta}}\left[\bar{A}_g^{\pi_\theta}(s)\right]\right). \end{aligned}$$

By definition of ε_A,

$$\left|\bar{A}_g^{\pi_\theta}(s)\right| \le \varepsilon_A \quad \text{for all } s.$$

Therefore,

$$\left|\mathbb{E}_{s\sim d^{\pi_{\mathrm{ref}}}}\left[\bar{A}_g^{\pi_\theta}(s)\right] - \mathbb{E}_{s\sim d^{\pi_\theta}}\left[\bar{A}_g^{\pi_\theta}(s)\right]\right| \le \varepsilon_A \|d^{\pi_{\mathrm{ref}}} - d^{\pi_\theta}\|_1.$$

Applying Lemma 2,

$$\|d^{\pi_{\mathrm{ref}}} - d^{\pi_\theta}\|_1 \le \frac{2\gamma}{1-\gamma}\,\delta_\pi.$$

Combining the inequalities gives

$$J_g(\pi_{\mathrm{ref}}) - J_g(\pi_\theta) \ge \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi_\theta}}\left[\bar{A}_g^{\pi_\theta}(s)\right] - \frac{2\gamma}{(1-\gamma)^2}\,\varepsilon_A\,\delta_\pi.$$

□

Corollary 1 (Sufficient condition for improvement)**.** *If the expected local advantage term in Theorem 2 dominates the conservatism penalty, then PoE improves over the frozen actor.*

A.5 Proof of Corollary 1

The corollary follows immediately from Theorem 2. If

$$\mathbb{E}_{s\sim d^{\pi_\theta}}\left[\bar{A}_g^{\pi_\theta}(s)\right] > \frac{2\gamma}{1-\gamma}\,\varepsilon_A\,\delta_\pi,$$

then the lower bound on

$$J_g(\pi_{\mathrm{ref}}) - J_g(\pi_\theta)$$

is strictly positive. □

A.6 Auxiliary Lemma for Transition Shift

Lemma 3 (State occupancy shift under kernel perturbation)**.** *Assume*

$$\sup_{s,a} \mathrm{TV}(P_{\mathrm{train}}(\cdot \mid s,a), P_{\mathrm{deploy}}(\cdot \mid s,a)) \le \varepsilon_P.$$

Then for any stationary policy π,

$$\|d^{\pi}_{\mathrm{deploy}} - d^{\pi}_{\mathrm{train}}\|_1 \le \frac{2\gamma}{1-\gamma}\,\varepsilon_P.$$

Proof. Fix a policy π and define the policy-induced state kernels

$$P^{\pi}_{\text{train}}(s' \mid s) = \sum_a \pi(a \mid s) P_{\text{train}}(s' \mid s, a), \quad P^{\pi}_{\text{deploy}}(s' \mid s) = \sum_a \pi(a \mid s) P_{\text{deploy}}(s' \mid s, a).$$

By convexity of total variation,

$$\sup_s \text{TV}(P^{\pi}_{\text{train}}(\cdot \mid s), P^{\pi}_{\text{deploy}}(\cdot \mid s)) \le \varepsilon_P.$$

Now write the discounted occupancy equations:

$$d^{\pi}_{\text{train}} = (1-\gamma)\mu + \gamma (P^{\pi}_{\text{train}})^{\top} d^{\pi}_{\text{train}}, \qquad d^{\pi}_{\text{deploy}} = (1-\gamma)\mu + \gamma (P^{\pi}_{\text{deploy}})^{\top} d^{\pi}_{\text{deploy}},$$

where μ is the initial state distribution. Subtracting,

$$d^{\pi}_{\text{deploy}} - d^{\pi}_{\text{train}} = \gamma (P^{\pi}_{\text{deploy}})^{\top} (d^{\pi}_{\text{deploy}} - d^{\pi}_{\text{train}}) + \gamma \big((P^{\pi}_{\text{deploy}})^{\top} - (P^{\pi}_{\text{train}})^{\top} \big) d^{\pi}_{\text{train}}.$$

Taking ℓ_1 norms and using that stochastic kernels are nonexpansive in ℓ_1,

$$\|d^{\pi}_{\text{deploy}} - d^{\pi}_{\text{train}}\|_1 \le \gamma \|d^{\pi}_{\text{deploy}} - d^{\pi}_{\text{train}}\|_1 + \gamma \big\| (P^{\pi}_{\text{deploy}})^{\top} - (P^{\pi}_{\text{train}})^{\top} \big\|_{1\to 1} \|d^{\pi}_{\text{train}}\|_1.$$

Because $\|d^{\pi}_{\text{train}}\|_1 = 1$ and the operator norm difference is bounded by $2\varepsilon_P$, we obtain

$$\|d^{\pi}_{\text{deploy}} - d^{\pi}_{\text{train}}\|_1 \le \gamma \|d^{\pi}_{\text{deploy}} - d^{\pi}_{\text{train}}\|_1 + 2\gamma \varepsilon_P.$$

Rearranging yields

$$\|d^{\pi}_{\text{deploy}} - d^{\pi}_{\text{train}}\|_1 \le \frac{2\gamma}{1-\gamma} \varepsilon_P.$$

□ □

Theorem 3 (Robustness to deployment transition shift)**.** *Under bounded train-to-deploy transition shift, the return of any policy differs between the train and deployment domains by at most* $\frac{2\gamma R_{\max}}{(1-\gamma)^2} \varepsilon_P$.

A.7 Proof of Theorem 3

For any policy π,

$$J^{\text{deploy}}_g(\pi) - J^{\text{train}}_g(\pi) = \frac{1}{1-\gamma} \left(\mathbb{E}_{s\sim d^{\pi}_{\text{deploy}},\, a\sim\pi}[r_g(s,a)] - \mathbb{E}_{s\sim d^{\pi}_{\text{train}},\, a\sim\pi}[r_g(s,a)] \right).$$

By bounded rewards,

$$\left| \mathbb{E}_{s\sim d^{\pi}_{\text{deploy}},\, a\sim\pi}[r_g(s,a)] - \mathbb{E}_{s\sim d^{\pi}_{\text{train}},\, a\sim\pi}[r_g(s,a)] \right| \le R_{\max} \|d^{\pi}_{\text{deploy}} - d^{\pi}_{\text{train}}\|_1.$$

Applying Lemma 3,

$$\left| J^{\text{deploy}}_g(\pi) - J^{\text{train}}_g(\pi) \right| \le \frac{R_{\max}}{1-\gamma} \cdot \frac{2\gamma}{1-\gamma} \varepsilon_P = \frac{2\gamma R_{\max}}{(1-\gamma)^2} \varepsilon_P.$$

□

Corollary 2 (Deployment-side improvement bound)**.** *Combining Theorem 3 with Theorem 2 yields a deployment-side lower bound on the refined policy's improvement over the frozen actor.*

A.8 Proof of Corollary 2

By Theorem 3,

$$J^{\text{deploy}}_g(\pi_{\text{ref}}) \ge J^{\text{train}}_g(\pi_{\text{ref}}) - \frac{2\gamma R_{\max}}{(1-\gamma)^2} \varepsilon_P$$

and

$$J^{\text{deploy}}_g(\pi_\theta) \le J^{\text{train}}_g(\pi_\theta) + \frac{2\gamma R_{\max}}{(1-\gamma)^2} \varepsilon_P.$$

Subtracting,

$$J^{\text{deploy}}_g(\pi_{\text{ref}}) - J^{\text{deploy}}_g(\pi_\theta) \ge J^{\text{train}}_g(\pi_{\text{ref}}) - J^{\text{train}}_g(\pi_\theta) - \frac{4\gamma R_{\max}}{(1-\gamma)^2} \varepsilon_P.$$

Now apply Theorem 2 to the train-side difference to obtain the stated result. □

Corollary 3 (Plug-in prior stability)**.** *Assume the estimated log-prior is uniformly close to the true log-prior on the effective common support, i.e.* $\sup_{s,a,g} |\log \widehat{\rho}(a \mid s, g) - \log \rho_\phi(a \mid s, g)| \le \varepsilon_n(\delta)$. *Then the plug-in refined policy* $\widehat{\pi}_{\mathrm{ref}}$ *satisfies*

$$\mathcal{F}_{s,g}(\pi_{\mathrm{ref}}) - \mathcal{F}_{s,g}(\widehat{\pi}_{\mathrm{ref}}) \le 2(1-\alpha)\,\varepsilon_n(\delta), \qquad \mathrm{KL}\big(\widehat{\pi}_{\mathrm{ref}}(\cdot \mid s, g) \,\|\, \pi_{\mathrm{ref}}(\cdot \mid s, g)\big) \le 2(1-\alpha)\,\varepsilon_n(\delta),$$

uniformly in (s, g). *The first inequality is a variational-objective gap; the second follows because the entropy-regularized objective* $\mathcal{F}_{s,g}$ *in Eq.* (4) *is a negative KL up to an additive constant, with* $\mathcal{F}_{s,g}(\pi_{\mathrm{ref}}) - \mathcal{F}_{s,g}(\pi) = \mathrm{KL}(\pi \| \pi_{\mathrm{ref}})$ *for any* π *supported on* $\Gamma_{s,g}$.

A.9 Proof of Corollary 3

Fix (s, g) and define

$$\widehat{\mathcal{F}}_{s,g}(\pi) = \mathbb{E}_{a\sim\pi}[\alpha \log \pi_\theta(a \mid s) + (1-\alpha) \log \widehat{\rho}(a \mid s, g)] + \mathcal{H}(\pi).$$

By assumption,

$$\sup_{s,a,g} |\log \widehat{\rho}(a \mid s, g) - \log \rho_\phi(a \mid s, g)| \le \varepsilon_n(\delta).$$

Hence for any policy π,

$$\left|\widehat{\mathcal{F}}_{s,g}(\pi) - \mathcal{F}_{s,g}(\pi)\right| \le (1-\alpha)\,\varepsilon_n(\delta).$$

Since π_{ref} maximizes $\mathcal{F}_{s,g}$ and $\widehat{\pi}_{\mathrm{ref}}$ maximizes $\widehat{\mathcal{F}}_{s,g}$,

$$\begin{aligned}\mathcal{F}_{s,g}(\pi_{\mathrm{ref}}) - \mathcal{F}_{s,g}(\widehat{\pi}_{\mathrm{ref}}) = {} & \big(\mathcal{F}_{s,g}(\pi_{\mathrm{ref}}) - \widehat{\mathcal{F}}_{s,g}(\pi_{\mathrm{ref}})\big) \\ & + \big(\widehat{\mathcal{F}}_{s,g}(\pi_{\mathrm{ref}}) - \widehat{\mathcal{F}}_{s,g}(\widehat{\pi}_{\mathrm{ref}})\big) \\ & + \big(\widehat{\mathcal{F}}_{s,g}(\widehat{\pi}_{\mathrm{ref}}) - \mathcal{F}_{s,g}(\widehat{\pi}_{\mathrm{ref}})\big).\end{aligned}$$

The middle term is nonpositive because $\widehat{\pi}_{\mathrm{ref}}$ maximizes $\widehat{\mathcal{F}}_{s,g}$. Therefore,

$$\mathcal{F}_{s,g}(\pi_{\mathrm{ref}}) - \mathcal{F}_{s,g}(\widehat{\pi}_{\mathrm{ref}}) \le 2(1-\alpha)\varepsilon_n(\delta).$$

For the KL form, note that on $\Gamma_{s,g}$ the entropy-regularized objective satisfies

$$\mathcal{F}_{s,g}(\pi) = -\,\mathrm{KL}\big(\pi \,\|\, \pi_{\mathrm{ref}}(\cdot \mid s, g)\big) + \log Z_{s,g},$$

because $\pi_{\mathrm{ref}} \propto \exp(\alpha \log \pi_\theta + (1-\alpha) \log \rho_\phi)$ is the unique maximizer. Hence $\mathcal{F}_{s,g}(\pi_{\mathrm{ref}}) - \mathcal{F}_{s,g}(\widehat{\pi}_{\mathrm{ref}}) = \mathrm{KL}(\widehat{\pi}_{\mathrm{ref}} \,\|\, \pi_{\mathrm{ref}})$, and the previous inequality gives

$$\mathrm{KL}\big(\widehat{\pi}_{\mathrm{ref}}(\cdot \mid s, g) \,\|\, \pi_{\mathrm{ref}}(\cdot \mid s, g)\big) \le 2(1-\alpha)\,\varepsilon_n(\delta).$$

Taking the supremum over (s, g) gives both stated bounds. □

Proposition 2 (Gaussian PoE gain-deviation bound; structural sketch)**.** *Assume diagonal-Gaussian actor and prior policies; bounded rewards; advantage Lipschitz in action* ($a \mapsto A_g^{\pi_\theta}(s, a)$ *is* L_A*-Lipschitz uniformly in* s*); transition kernel Lipschitz in action; and uniformly bounded covariance eigenvalues for* $\pi_\theta(\cdot \mid s)$ *across states. Under these conditions the Gaussian PoE policy remains Gaussian in closed form, and its return improvement admits a sketch-level gain-minus-deviation bound controlled by Wasserstein distance from the frozen actor (see Eq.* (6) *below).*

Corollary 4 (KL-controlled Gaussian sketch)**.** *Under the assumptions of Proposition 2, a Gaussian transportation inequality lets the Wasserstein deviation in Eq.* (6) *be written in terms of KL divergence from the frozen actor.*

Scope of these statements. We treat Proposition 2 and Corollary 4 as structural sketches rather than rigorous theorems: we make the Gaussian closed form fully explicit and follow the same gain-minus-deviation decomposition pattern as Theorem 2, but the steps below assume Lipschitz advantage, Lipschitz transitions, and uniform Gaussian covariance bounds without proving them for our learned actor. The empirical claims of the main paper rest on rollouts (Sec. 7), not on these inequalities.

A.10 Sketch for Proposition 2 and Corollary 4

We sketch the argument for the continuous-action Gaussian refinement result.

Closed-form Gaussian PoE. Let

$$\pi_\theta(\cdot \mid s) = \mathcal{N}(\mu_\theta(s), \Sigma_\theta(s)), \qquad \rho_\phi(\cdot \mid s, g) = \mathcal{N}(\mu_\phi(s, g), \Sigma_\phi(s, g)),$$

with diagonal positive-definite covariance matrices. Writing the log densities and taking the weighted Product-of-Experts form gives

$$\log \pi_\alpha(a \mid s, g) = \alpha \log \pi_\theta(a \mid s) + (1 - \alpha) \log \rho_\phi(a \mid s, g) - \log Z_\alpha(s, g),$$

where $Z_\alpha(s, g)$ is the normalizing constant. Since each Gaussian log density is quadratic in a, their weighted sum is also quadratic in a. Completing the square yields another Gaussian density with precision

$$\Sigma_\alpha(s, g)^{-1} = \alpha\, \Sigma_\theta(s)^{-1} + (1 - \alpha)\, \Sigma_\phi(s, g)^{-1},$$

and mean

$$\mu_\alpha(s, g) = \Sigma_\alpha(s, g)\Big(\alpha\, \Sigma_\theta(s)^{-1}\mu_\theta(s) + (1 - \alpha)\, \Sigma_\phi(s, g)^{-1}\mu_\phi(s, g)\Big).$$

This proves the first part of Proposition 2.

Gain–deviation sketch. Using the performance-difference lemma around the frozen actor π_θ, we can write the return difference in terms of the expected frozen-actor advantage under the refined policy, together with an error term induced by the difference between the refined policy and the frozen actor. *Assuming* that $a \mapsto A_g^{\pi_\theta}(s, a)$ is L_A-Lipschitz uniformly in s and that the transition kernel $P(\cdot \mid s, a)$ is Lipschitz in a, this error can be controlled by the 2-Wasserstein distance between $\pi_\alpha(\cdot \mid s, g)$ and $\pi_\theta(\cdot \mid s)$. The sketch follows the same decomposition pattern as Theorem 2, replacing finite-action total-variation control with a Lipschitz Wasserstein control. We then take a supremum over states rather than only an expectation, to avoid hidden support assumptions, yielding the sketch-level inequality

$$J_g(\pi_\alpha) - J_g(\pi_\theta) \;\geq\; \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi_\theta},\, a\sim\pi_\alpha(\cdot|s,g)}\big[A_g^{\pi_\theta}(s, a)\big] - \frac{2\gamma L_A}{(1-\gamma)^2}\sup_s W_2\big(\pi_\alpha(\cdot \mid s, g), \pi_\theta(\cdot \mid s)\big). \tag{6}$$

We do not verify the Lipschitz advantage and Lipschitz transition assumptions for our learned actor, so we read Eq. (6) as a structural reading of the gain–deviation tradeoff, not as a quantitative bound.

KL-controlled version. For Gaussian reference measures, *assuming* the covariance eigenvalues of $\pi_\theta(\cdot \mid s)$ are uniformly bounded over states (so that a uniform transportation constant $C_\theta < \infty$ exists), a Gaussian transportation inequality gives

$$W_2\big(\pi_\alpha(\cdot \mid s, g), \pi_\theta(\cdot \mid s)\big) \leq \sqrt{2C_\theta\ \mathrm{KL}\big(\pi_\alpha(\cdot \mid s, g) \,\|\, \pi_\theta(\cdot \mid s)\big)},$$

uniformly in s. Substituting this into Eq. (6) gives the sketch-level KL form referenced in Corollary 4. We do not verify the uniform-covariance assumption for our trained actors, so the KL-controlled version inherits the same structural-only reading.

Interpretation. This argument does not claim that every finite-action guarantee transfers unchanged to continuous control. Rather, it shows that the exact Gaussian mechanism used in the D4RL experiments admits a clean analytical form and a deviation-aware reading consistent with the empirical return–vs.–KL evaluation in the main text. The empirical claims of the paper rest on rollouts, not on these sketch-level inequalities.

B Evaluation Protocol and Metrics (Diagnostic Only)

Note on scope. This appendix documents the off-policy scorer used as a development diagnostic; the main paper's claims are based on real MuJoCo rollouts (Sec. 6, Sec. 7). We include the protocol so that appendix numbers that use this scorer can be interpreted consistently.

B.1 Protocol and Evaluators

Action selection conventions. The MuJoCo rollouts that produce all main-body returns (Sec. 7) use *deterministic mean actions* for every method: PoE, KL-Reg, Frozen Actor, Prior Only, Additive Mix, AWR, CQL/IQL-guided, and SF/GPI all execute their respective Gaussian-policy means at each step. The off-policy diagnostic protocol below also scores deterministic mean actions on dataset states. The only exception in this paper is the BC-mode audit of Table 15 (Appendix O.8), where we explicitly compare argmax/mean and stochastic-sample evaluation modes for a single behavioral-cloning policy; the stochastic mode there is labeled in the table.

Let $\mathcal{D} = \{(s_i, a_i^{\text{data}}, r_i, s_i', d_i)\}_{i=1}^N$ be the offline D4RL transitions. We decompose each r_i into the three reward components $rc_i = (rc_i^{\text{fwd}}, rc_i^{\text{ctrl}}, rc_i^{\text{alive}})$ as in Sec. 6, and use the same three-dimensional goals $g \in \mathbb{R}^3$ and goal-weighted reward $r_g(s,a) = g \cdot rc(s,a)$ as the main body. For a candidate policy π we score the deterministic-mean action $\hat{a}_i^\pi = \mathbb{E}_{a\sim\pi(\cdot|s_i,g)}[a]$ with a goal-conditioned critic Q_g^{eval} trained on $\mathcal{D}$ (Sec. B.3) and aggregate across the dataset. Q_g^{eval} is trained with a random seed and split independent of the refinement prior, and is never consulted during policy construction. A handful of within-family ablations use the original deployment critic Q_g and are explicitly labeled as within-protocol diagnostics.

B.2 Diagnostic Metrics

For a policy π, the diagnostic Q-value score (a one-step fitted-Q score over dataset states; *not* a rollout return) and two quantile-threshold risk rates are

$$\widehat{S}_g(\pi) = \tfrac{1}{N} \sum_i Q_g^{\text{eval}}(s_i, \hat{a}_i^\pi)\,, \tag{7}$$

$$\text{Cat}\% = \tfrac{1}{N} \sum_i \mathbf{1}\big[Q_g^{\text{eval}}(s_i, \hat{a}_i^\pi) < \tau_{\text{cat}}\big]\,, \tag{8}$$

$$\text{Con}\% = \tfrac{1}{N} \sum_i \mathbf{1}\big[Q_g^{\text{eval}}(s_i, \hat{a}_i^\pi) < \tau_{\text{con}}\big]\,. \tag{9}$$

The score $\widehat{S}_g$ is not a rollout return and should be interpreted only as a diagnostic for candidate action quality under the held-out critic; numerical values from $\widehat{S}_g$ are not directly comparable to MuJoCo rollout returns.

With τ_{cat} and τ_{con} set to the 10th and 5th percentiles of $\{Q_g^{\text{eval}}(s_i, a_i^{\text{data}})\}$, we have by construction $\tau_{\text{con}} < \tau_{\text{cat}}$, and therefore $\text{Con}\% \le \text{Cat}\%$. We summarize lower-tail behavior by

$$\text{Rob} = 1 - \text{Cat}\% - 0.5\,\text{Con}\%.$$

Quantile thresholds are used because fitted goal-conditioned critics compress values, and the goal-weighted reward has a narrower dynamic range than the raw MuJoCo reward. These are evaluator-relative diagnostics, not absolute safety boundaries.

B.3 Goal-Conditioned Critic Training

The scorer is trained by fitted-Q evaluation with one-step TD targets:

$$\mathcal{L}_{\text{FQE}} = \mathbb{E}_{(s,a,r,s',d)\sim\mathcal{D},\, g}\Big[\big(Q_g(s,a) - \big(r_g(s,a) + \gamma(1-d)\, Q_{\bar{g}}(s', a')\big)\big)^2\Big]\,,$$

where a' is sampled from a diagonal-Gaussian target policy at s' (avoiding the discrete $\max_{a'}$ that is valid only in finite-action settings), and $Q_{\bar{g}}$ is a Polyak target network with $\tau = 5 \times 10^{-3}$.

B.4 Policy Construction Versus Evaluation

For the refinement-network prior, policy construction and evaluation are fully separated; this is the cleanest off-policy diagnostic of refinement. For the critic-prior variant $\rho_\phi(a \mid s, g) \propto \exp(Q_g(s,a)/\tau)$, evaluator overlap arises if the construction critic is reused for scoring. The main paper avoids this entirely by reporting MuJoCo rollout returns; when this appendix uses the off-policy scorer, we use the independent held-out Q_g^{eval}.

C Equivalence Audit: PoE(α) $\equiv$ KL-Reg($\beta = \alpha/(1-\alpha)$)

This appendix supports the equivalence claim in Sec. 4.3 and the headline equivalence finding in Sec. 7.1. We first record the explicit four-line derivation that ties PoE and KL-regularized adaptation together:

$$\pi^*(a \mid s, g) \;\propto\; \pi_\theta(a \mid s)\, \rho_\phi(a \mid s, g)^{1/\beta}, \tag{10}$$

$$\Lambda_{\text{KL-Reg}}(\beta) \;=\; \Sigma_\theta^{-1} + (1/\beta)\, \Sigma_\phi^{-1}, \tag{11}$$

$$\beta \;=\; \alpha/(1-\alpha) \;\Longrightarrow\; \Lambda_{\text{KL-Reg}}(\beta) \;=\; (1/\alpha)\big[\alpha\, \Sigma_\theta^{-1} + (1-\alpha)\, \Sigma_\phi^{-1}\big], \tag{12}$$

$$\therefore\; \Sigma_{\text{ref}}^{\text{PoE}}(\alpha) \;=\; \tfrac{1}{\alpha}\, \Sigma_{\text{ref}}^{\text{KL-Reg}}(\beta) \;=\; \tfrac{1+\beta}{\beta}\, \Sigma_{\text{ref}}^{\text{KL-Reg}}(\beta). \tag{13}$$

The bracketed quantity in line 3 is precisely $\Lambda_{\text{ref}}^{\text{PoE}}(\alpha)$, so $\Lambda_{\text{ref}}^{\text{PoE}} = \alpha\, \Lambda_{\text{KL-Reg}}$, the natural parameters of the two distributions are scalar multiples, the refined means coincide exactly, and the refined covariances differ only by the global scalar in line 4. Because the rollout protocol uses deterministic mean actions (Sec. 6, Appendix B), the covariance scaling cancels in deployment and the empirical equivalence of $\text{PoE}(\alpha)$ and $\text{KL-Reg}(\beta = \alpha/(1-\alpha))$ holds at the level of executed trajectories. We verify the identity at two levels.

Dataset-state verification. To verify the equivalence at the dataset-state level, we evaluated $5{,}000$ randomly sampled states from each environment under three deployment goals. For each state, we computed the Gaussian closed-form solutions for both $\text{PoE}(\alpha)$ and $\text{KL-Reg}(\beta = \alpha/(1-\alpha))$ across $\alpha \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$. We then measured the maximum and mean absolute differences between the resulting per-state means. Across all 60 setting–goal–α combinations, the largest observed difference in $|\mu_{\text{PoE}} - \mu_{\text{KL-Reg}}|$ was no greater than 2×10^{-6}. At $\alpha = 0.5$, the two formulations produced numerically identical means. We also confirmed the corresponding covariance identity numerically, with a relative residual below 5×10^{-7}. These results indicate that the deterministic-mean equivalence holds consistently across sampled dataset states and is not affected by the covariance convention used for reporting.

Rollout verification. We re-use the main-paper rollout data (5 seeds $\times$ 5 episodes $\times$ 3 goals per environment). At the matched pair $\alpha = 0.5,\ \beta = 1.0$, the raw per-episode goal-weighted returns and episode lengths agree to the exact digit in all 12 setting-by-goal cells, which makes the paired mean-difference 0 and the probability of improvement 0 in every cell. At other matched pairs $\alpha \in \{0.1, 0.3, 0.7, 0.9\}$, paired differences have bootstrap 95% confidence intervals that include zero in nearly every cell. Residual nonzero differences are consistent with chaotic amplification of float-precision action jitter over $1{,}000$ MuJoCo steps, not with a policy-level gap. Figure 1 summarizes both levels of verification.

D Conservative-Improvement Bound: Empirical Diagnostic

This appendix supports the structural framing of Theorem 2 in Sec. 5. We estimate the empirical δ_π for the refined policy relative to the frozen actor across our four task settings and three goals, and evaluate the right-hand side of Eq. (5) at $\gamma = 0.99$.

Setup. For each environment we sample $5{,}000$ dataset states and each deployment goal, compute the refined Gaussian under $\text{PoE}(\alpha)$ for $\alpha \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$, and estimate the total-variation distance $\text{TV}(\pi_{\text{ref}}(\cdot \mid s, g), \pi_\theta(\cdot \mid s))$ per state via Monte Carlo with 256 samples, together with the Pinsker upper bound $\sqrt{0.5\, \text{KL}}$. We then take the sup over states as the plug-in estimate of δ_π.

Findings. At $\gamma = 0.99$, the penalty coefficient $\frac{2\gamma}{(1-\gamma)^2}$ equals $19{,}800$, making the conservative performance bound numerically large. Across the four D4RL task settings, the plug-in supremum-state total variation estimate approaches its upper range for most values of α and deployment goals. For instance, even when the mean KL divergence remains small, as in `halfcheetah-medium-v2` at $\alpha = 0.9$, the corresponding Pinsker-based total variation estimate is already close to its maximal value, reflecting the looseness of the bound. Using a conservative unit-scale estimate for the empirical advantage range as a proxy for ε_A, the resulting penalty term is on the order of 1.9×10^4 across settings. This magnitude substantially exceeds the observed goal-weighted return improvements,

Table 1: Headline seed-robustness audit. The only changed verdict is Hopper-medium G_3, which moves from FROZEN to HURT. Values are reported from the discussion-period replication.

Environment	Goal	Δ_5	Δ_{10}	ε_{10}	Verdict ($5 \to 10$)
HalfCheetah-M	G_1	247.6	232.1	359.2	Frozen $\to$ Frozen
HalfCheetah-M	G_2	167.5	135.2	186.4	Frozen $\to$ Frozen
HalfCheetah-M	G_3	29.0	29.0	50.0	Frozen $\to$ Frozen
Hopper-M	G_1	-291.0	-338.6	45.4	Hurt $\to$ Hurt
Hopper-M	G_2	143.8	147.6	30.4	Help $\to$ Help
Hopper-M	G_3	-4.6	-11.0	6.3	Frozen $\to$ Hurt
Walker2d-M	G_1	-2029.9	-2013.8	159.8	Hurt $\to$ Hurt
Walker2d-M	G_2	123.5	86.4	101.6	Frozen $\to$ Frozen
Walker2d-M	G_3	55.4	22.4	21.3	Help $\to$ Help
Hopper-ME	G_1	-877.8	-892.0	55.8	Hurt $\to$ Hurt
Hopper-ME	G_2	54.5	74.7	37.2	Help $\to$ Help
Hopper-ME	G_3	110.2	112.4	7.6	Help $\to$ Help

indicating that Theorem 2 provides a valid but empirically loose decomposition under the discount factors commonly used in offline reinforcement-learning benchmarks. Accordingly, we interpret the theorem primarily as a structural guarantee rather than a tight predictor of empirical gains.

E Ten-Seed Replication and Verdict Sensitivity

We replicated the twelve headline setting–goal cells with five additional seeds (47–51), adding 3,900 rollout episodes under the same protocol. The table reports the best-composition-minus-frozen return difference (Δ), together with the ten-seed CI-half-width threshold ε used for the HELP/FROZEN/HURT classification. The original five-seed figures and baseline tables remain historical five-seed estimates; their raw means and intervals should not be interpreted as ten-seed estimates.

Leave-one-seed-out check. Removing each of the five original seeds in turn leaves eight of twelve classifications invariant. No cell switches directly between HELP and HURT; sensitivity is concentrated in the three HalfCheetah-medium FROZEN cells under removal of one seed and Hopper-medium-expert G_2. The ten-seed replication leaves all four original HELP and all three original HURT classifications intact, but changes one borderline FROZEN cell to HURT. This supports robustness of the large effects without implying that every near-threshold label is seed invariant.

F Dataset-Only Sensitivity Score on the Harder Regimes

The sensitivity score of Sec. 8 is $S = \frac{1}{N}\sum_i \mathrm{KL}\big(\pi_{\mathrm{ref}}^{\alpha=0.5}(\cdot \mid s_i, g) \,\|\, \pi_\theta(\cdot \mid s_i)\big)$ over $N = 5{,}000$ dataset states, computed in closed form from the frozen actor and the prior without rollouts. Table 2 applies it unchanged to the 18 harder-regime cells, with the per-cell best-α verdicts of Table 6 as labels; hopper-medium-expert reuses the main-package actor and prior, as in Appendix I.

S is negatively rank-correlated with the best-composition return delta on the harder cells (Spearman -0.64, $p = 0.0045$), more strongly on medium-expert (-0.70, $p = 0.036$) than on medium-replay (-0.47, $p = 0.21$). Pooled with the 12 main cells (five-seed labels), the correlation is -0.67 ($p = 4.9 \times 10^{-5}$). The three highest-scoring harder cells are the three medium-expert G_1 cells, all HURT. However, the clean separation seen on the main cells does not carry over: the two medium-replay G_1 cells score high (9.0 and 6.6) yet are FROZEN, and 5 of the 8 HURT cells rank in the top 8. We therefore read S as a ranking of steering risk for a validated actor, not as a classifier with a transferable threshold.

G Help / Frozen / Hurt Boundary Classification

This appendix describes the diagnostic rule used in Sec. 7.2 to classify each setting–goal cell according to whether policy composition improves, preserves, or reduces performance relative to the frozen actor. For each cell, we use the frozen-actor mean goal-weighted return $\bar{J}_g^\theta$, its 95% bootstrap

Table 2: Sensitivity score S on the harder-regime cells, sorted by S. Δ is the best-α composition return minus the frozen-actor return (Table 6).

Environment	Goal	S	Δ	Verdict
halfcheetah-ME	G_1	51.86	-5191.1	Hurt
hopper-ME	G_1	39.98	-1182.5	Hurt
walker2d-ME	G_1	17.51	-4520.7	Hurt
hopper-MR	G_1	9.02	$+3.9$	Frozen
walker2d-MR	G_1	6.60	-113.2	Frozen
hopper-ME	G_2	5.50	$+109.0$	Help
walker2d-ME	G_2	4.71	-1324.9	Hurt
walker2d-ME	G_3	4.07	-1323.7	Hurt
halfcheetah-ME	G_2	3.58	-378.9	Hurt
hopper-MR	G_2	2.40	$+4.4$	Frozen
halfcheetah-ME	G_3	2.25	-345.0	Hurt
hopper-ME	G_3	1.97	$+1093.4$	Help
walker2d-MR	G_2	1.73	-234.3	Hurt
walker2d-MR	G_3	1.43	-115.0	Frozen
halfcheetah-MR	G_3	0.65	$+616.3$	Help
hopper-MR	G_3	0.49	$+0.3$	Frozen
halfcheetah-MR	G_2	0.47	$+409.0$	Frozen
halfcheetah-MR	G_1	0.36	$+421.9$	Frozen

confidence interval $[\ell_\theta, h_\theta]$, and the best composition mean $\bar{J}_g^{\text{best}}$ over $\{\text{PoE}, \text{KL-Reg}, \text{Additive}\}$. We define the tolerance margin as

$$\epsilon = \max(h_\theta - \bar{J}_g^\theta, \bar{J}_g^\theta - \ell_\theta),$$

which is the larger half-width of the frozen actor's confidence interval.

We assign each cell to one of three categories:

- HELP: $\bar{J}_g^{\text{best}} - \bar{J}_g^\theta > \epsilon$.
- HURT: $\bar{J}_g^\theta - \bar{J}_g^{\text{best}} > \epsilon$, meaning that all composition methods perform below the frozen actor by more than the tolerance margin.
- FROZEN: neither of the above conditions holds, so the best composition result remains within the frozen actor's confidence margin.

Across the 12 evaluated setting–goal cells, the original five-seed outcomes are 4 HELP, 5 FROZEN, and 3 HURT; the ten-seed replication yields 4 HELP, 4 FROZEN, and 4 HURT (Appendix E). The three large-magnitude HURT cases occur for the G_1 speed objective in `hopper-medium-v2`, `hopper-medium-expert-v2`, and `walker2d-medium-v2`, where the frozen actor exceeds the best composition method by 291, 878, and 2,030, respectively. The fourth, smaller ten-seed HURT cell is hopper-medium G_3 ($\Delta = -11.0$, CI half-width 6.3). The HELP cases occur for `hopper-medium-v2` under G_2 with a gain of 144, `hopper-medium-expert-v2` under G_2 with a gain of 55, `hopper-medium-expert-v2` under G_3 with a gain of 110, and `walker2d-medium-v2` under G_3 with a gain of 55.

H First-Order Critic-Gradient Baseline (AWR/AWAC-Style)

This appendix documents the structurally distinct AWR baseline introduced in Sec. 6 and Sec. 7.3.

Policy. Following Peng et al. [Peng et al., 2019] and Nair et al. [Nair et al., 2020], the AWR/AWAC closed-form optimum is $\pi(a \mid s) \propto \pi_\theta(a \mid s) \exp(A_g(s,a)/\beta)$. For a diagonal-Gaussian actor with mean $\mu_\theta(s)$ and variance $\sigma_\theta^2(s)$, and a locally linear advantage around $\mu_\theta(s)$, this is a first-order mean shift

$$\mu_{\text{AWR}}(s,g) = \mu_\theta(s) + \frac{\sigma_\theta^2(s)}{\beta} \nabla_a Q_g(s,a)\big|_{a=\mu_\theta(s)}, \qquad \sigma_{\text{AWR}}(s,g) = \sigma_\theta(s),$$

with a per-dimension clip applied to the shift for numerical safety. We run this baseline at $\beta \in \{0.5, 1.0, 3.0\}$.

Table 3: AWR/AWAC-style baseline, best-in-family per cell. Goal-weighted return with 95% bootstrap CI; mean KL from the frozen actor along rollout states. Selected operating point is $\beta = 3.0$ in every cell.

Environment	Goal	Ret. mean	CI low	CI high	KL
halfcheetah-medium-v2	G_1 speed	4525.8	4140.8	4910.7	1.048
halfcheetah-medium-v2	G_2 balanced	3263.2	3206.7	3315.4	0.105
halfcheetah-medium-v2	G_3 control	989.1	981.7	995.6	0.046
hopper-medium-v2	G_1 speed	36.5	36.0	37.0	1.975
hopper-medium-v2	G_2 balanced	91.6	50.5	132.6	1.209
hopper-medium-v2	G_3 control	161.1	151.6	171.2	0.026
walker2d-medium-v2	G_1 speed	1260.2	758.2	1822.8	0.856
walker2d-medium-v2	G_2 balanced	1381.8	984.6	1765.5	0.185
walker2d-medium-v2	G_3 control	353.3	328.8	375.0	0.012
hopper-medium-expert-v2	G_1 speed	20.6	17.9	23.1	0.980
hopper-medium-expert-v2	G_2 balanced	30.9	29.6	32.1	0.872
hopper-medium-expert-v2	G_3 control	99.2	85.2	117.3	0.041

Q critic. The goal-conditioned critic $Q_g(s, a)$ is implemented as an MLP that takes the concatenated input $[s; a; g]$. The network applies LayerNorm to the input, followed by two hidden layers of width 256 and a scalar output head. We train the critic using fitted-Q evaluation on offline D4RL transitions for 30 epochs with batch size 512, AdamW optimization, learning rate 3×10^{-4}, weight decay 10^{-4}, gradient-norm clipping at 1.0, discount factor $\gamma = 0.99$, and Polyak target update rate $\tau = 5 \times 10^{-3}$. The critic is trained with a seed independent of those used for the actor and prior. For each target update, the next action is set to $a' = \mu_\theta(s')$, matching the deterministic-mean rollout protocol.

Training goals are sampled per batch from a three-component simplex mixture: 50% from $\mathrm{Dir}(1, 1, 1)$, 30% from $\mathrm{Dir}(0.5, 0.5, 0.5)$, and 20% uniformly from the deployment goals $\{G_1, G_2, G_3\}$. The final critic scales and normalized residuals are 228.6 and 4.0% for `halfcheetah-medium-v2`, 92.3 and 2.6% for `hopper-medium-v2`, 115.3 and 3.8% for `walker2d-medium-v2`, and 108.6 and 3.4% for `hopper-medium-expert-v2`.

Per-cell results. Table 3 reports best-in-family AWR results under the same rollout protocol as the main body (five seeds, five episodes per seed, three deployment goals, four task settings). The best β was consistently 3.0.

Aggregate verdict. Across the 12 setting–goal cells, AWR exceeds the frozen actor in 3 cases: `halfcheetah-medium-v2` under G_2 and G_3, and `walker2d-medium-v2` under G_3. It also outperforms the strongest non-AWR composition baseline among Additive, KL-Reg, and PoE in 3 cells. However, AWR falls more than 100 return points below the frozen actor in 7 cells. In five of these settings, its performance is close to the Prior-Only policy, namely `hopper-medium-v2` under G_1 and G_2, `hopper-medium-expert-v2` under G_1 and G_2, and `walker2d-medium-v2` under G_1. This pattern suggests that AWR is sensitive to critic-gradient reliability under offline evaluation. Although the critic fit is stable on in-distribution transitions, the action-gradient direction $\nabla_a Q_g$ at μ_θ can move the policy away from the support of the offline data, particularly in hopper-type environments where early termination strongly affects return. In contrast, `halfcheetah-medium-v2` does not exhibit the same early-termination failure mode, and AWR remains competitive on G_2 and G_3.

H.1 CQL/IQL-Guided Critic Baselines

The CQL/IQL sweep follows the same rollout protocol as the main body, using a fair $\beta \in \{0.5, 1.0, 3.0\}$ sweep for each critic family across the four main rollout task settings. This yields 12 setting–goal cells, with 5 seeds and 5 episodes per cell, using deterministic mean actions. This experiment addresses the reviewer concern that PoE/KL-Reg were evaluated over five α values and AWR over three β values, whereas CQL/IQL had previously been reported only at $\beta = 1.0$. Best-in-family operating points, defined as the $\beta \in \{0.5, 1.0, 3.0\}$ value that maximizes mean return for each cell, are reported in Table 4.

Table 4: Best-in-family CQL-Guided and IQL-Guided operating points after sweeping $\beta \in \{0.5, 1.0, 3.0\}$ under the main rollout protocol. Entries report mean goal-weighted return over 25 episodes per cell. Frozen-Actor return is the headline-package baseline evaluated with the same seeds. "Δ" denotes best-CQL minus Frozen; IQL returns are shown without a separate Δ column.

Env	Goal	Frozen	CQL $\beta^\star$	CQL ret	Δ	IQL $\beta^\star$	IQL ret
halfcheetah-medium-v2	G_1 speed	5,010.6	3.0	5,421.9	+411.2	1.0	5,245.7
halfcheetah-medium-v2	G_2 balanced	3,113.4	1.0	3,244.3	+130.9	0.5	3,283.5
halfcheetah-medium-v2	G_3 control	938.9	3.0	975.1	+36.2	3.0	987.9
hopper-medium-v2	G_1 speed	1,180.5	0.5	1,213.9	+33.3	0.5	1,185.8
hopper-medium-v2	G_2 balanced	822.8	1.0	857.0	+34.2	1.0	821.9
hopper-medium-v2	G_3 control	165.0	1.0	164.7	−0.4	0.5	166.4
walker2d-medium-v2	G_1 speed	2,431.3	1.0	2,574.7	+143.5	1.0	2,705.1
walker2d-medium-v2	G_2 balanced	1,610.2	0.5	1,732.1	+121.9	0.5	1,816.8
walker2d-medium-v2	G_3 control	324.2	3.0	344.8	+20.6	0.5	335.0
hopper-medium-expert-v2	G_1 speed	1,046.8	1.0	1,123.6	+76.8	1.0	1,134.8
hopper-medium-expert-v2	G_2 balanced	734.7	0.5	770.7	+36.0	0.5	800.8
hopper-medium-expert-v2	G_3 control	147.4	1.0	154.7	+7.3	0.5	157.5

Using the same CI-half-width classifier as the headline package in Appendix G, the best-β CQL-Guided rule HELPs on 5 of 12 cells and ties Frozen on the remaining 7. The best-β IQL-Guided rule HELPs on 6 of 12 cells and ties Frozen on the remaining 6. Importantly, neither critic-guided family HURTs performance on any of the 12 main-rollout cells at its selected best β.

After sweeping $\beta \in \{0.5, 1.0, 3.0\}$, CQL/IQL are stronger than the earlier single-β diagnostic. However, they still do not uniformly dominate the actor-anchored composition results. The critic-guided baselines improve several cells, but density-anchored composition retains important structural advantages: it does not require an auxiliary critic, admits the closed-form equivalence in Sec. 4.3, and exhibits the prior-degradation robustness analyzed in Sec. 7.4. Thus, the boundary characterization remains mixed rather than overturning the actor-anchored conclusion.

On the harder regimes in Appendix I, the CQL-guided baseline at $\beta = 1.0$ outperforms the frozen actor in several additional cells, including walker2d-medium-expert G_2 by $+95$, walker2d-medium-expert G_3 by $+54$, hopper-medium-replay G_2 by $+325$, hopper-medium-replay G_3 by $+121$, and hopper-medium-expert G_1 by $+103$. In contrast, IQL-guided at $\beta = 1.0$ on hopper-medium-expert collapses to approximately 5 return in all three cells. This appears to be an early-termination artifact associated with the single-β operating point and is not observed in the main four-setting sweep above, where $\beta = 0.5$ produces a small HELP on hopper-medium-expert.

I Harder-Regime Rollout Package

This appendix summarizes the harder D4RL rollout package used in Sec. 7.5. The evaluation follows the same protocol as the main experiments in Sec. 6: five random seeds, five episodes per seed, three deployment goals per cell, real MuJoCo rollouts without an artificial step cap, and matched PoE/KL-Reg comparisons. We also use the same prior construction and statistical reporting procedure as in the main-body experiments.

Environments. The harder-regime benchmark includes six D4RL cells:

- *Medium-replay*: `halfcheetah-medium-replay-v2`, `hopper-medium-replay-v2`, `walker2d-medium-replay-v2`.
- *Medium-expert*: `halfcheetah-medium-expert-v2`, `hopper-medium-expert-v2`, `walker2d-medium-expert-v2`.

The actor and prior for `hopper-medium-expert-v2` are reused from the main evaluation package, while the remaining five environment-specific actor–prior pairs are trained separately for this appendix.

Table 5: Harder-regime rollout summary at $\alpha = 0.5$ (entries: mean raw MuJoCo return over 25 episodes; negative values indicate backward motion). *MR* denotes *-medium-replay-v2*; *ME* denotes *-medium-expert-v2*. Bold indicates the non-Frozen winner when it clears one CI half-width; italic indicates a collapse well below Frozen (return $\lesssim 15\%$ of the Frozen value). The rightmost column shows the best per-α PoE entry from Appendix J, for reference.

Env	Goal	Frozen	PoE	KL-Reg	Additive	Prior	CQL	IQL	AWR	α-best (sweep)
halfcheetah-MR	G_1 speed	3309.9	2769.7	2769.7	2133.9	1238.3	2605.8	3309.9	*-62.2*	α=0.9: 3731.8
halfcheetah-MR	G_2 balanced	3309.9	**3666.6**	**3666.6**	3311.9	3104.9	**3516.4**	3309.9	1061.0	α=0.7: 3718.9
halfcheetah-MR	G_3 control	3309.9	2888.9	2888.9	2399.7	2080.8	**3579.7**	3271.4	**3412.4**	α=0.7: 3926.2
hopper-MR	G_1 speed	456.1	460.0	460.0	*60.9*	169.8	385.3	456.1	*55.3*	α=0.9: 460.0
hopper-MR	G_2 balanced	456.1	436.4	436.4	176.0	*41.5*	**781.3**	456.0	157.9	α=0.9: 460.6
hopper-MR	G_3 control	456.1	456.4	456.4	226.6	171.4	**577.3**	*11.4*	*23.1*	α=0.7: 456.4
walker2d-MR	G_1 speed	581.5	171.9	171.9	182.2	129.3	362.7	*55.2*	109.4	α=0.9: 468.3
walker2d-MR	G_2 balanced	581.5	261.2	261.2	319.9	296.1	417.5	298.9	223.6	α=0.3: 347.2
walker2d-MR	G_3 control	581.5	280.0	280.0	290.4	501.3	258.4	165.6	287.5	α=0.1: 466.5
halfcheetah-ME	G_1 speed	4796.0	*-447.3*	*-447.3*	*-424.7*	*-334.3*	*-313.6*	4796.0	*-475.6*	α=0.1: -395.0
halfcheetah-ME	G_2 balanced	4796.0	3922.3	3922.3	3317.8	3001.9	3662.6	4796.0	981.0	α=0.9: 4417.1
halfcheetah-ME	G_3 control	4796.0	3585.9	3585.9	3518.0	2597.8	4668.0	4796.0	4270.7	α=0.7: 4451.0
hopper-ME	G_1 speed	1468.5	127.7	127.7	177.7	*46.2*	**1571.3**	*5.3*	*24.7*	α=0.9: 286.0
hopper-ME	G_2 balanced	1468.5	282.8	282.8	229.3	140.1	1497.6	*5.0*	29.7	α=0.7: 1577.5
hopper-ME	G_3 control	1468.5	**1618.6**	**1618.6**	1447.4	**1700.8**	1541.5	*5.2*	195.6	α=0.1: 2561.9
walker2d-ME	G_1 speed	4919.8	*61.8*	*61.8*	*716.8*	*-15.6*	3293.1	4919.8	*458.0*	α=0.1: 399.1
walker2d-ME	G_2 balanced	4919.8	*128.0*	*128.0*	*250.5*	*296.8*	**5014.9**	4919.8	*285.9*	α=0.7: 3594.9
walker2d-ME	G_3 control	4919.8	3596.1	3596.1	3066.6	*215.7*	**4974.1**	4919.8	4343.1	α=0.5: 3596.1

Representative per-cell values ($\alpha = 0.5$). Table 5 reports the best-in-family operating points for the compact operating point $\alpha = 0.5$, $\beta = 1.0$, $\lambda = 0.5$ for the six locomotion cells. The full per-α breakdown is in Appendix J. Frozen-actor return, PoE return, matched KL-Reg return, Additive return, Prior-Only return, CQL return, IQL return, and AWR return are reported per goal; all values are seed-matched means over 25 episodes with bootstrap 95% CIs reported in the CSVs.

Two patterns are visible at the $\alpha = 0.5$ operating point alone: (i) on medium-replay, composition is occasionally helpful (notably halfcheetah G_2) but tends toward Frozen or mild HURT; (ii) on medium-expert, composition at $\alpha = 0.5$ is catastrophic on three of six cells (italic entries) and below Frozen on the remaining three. The α sweep in Appendix J confirms that this pattern is not a consequence of the single operating point; even the best per-cell α on medium-expert (right-most column) rarely closes the gap.

Equivalence of PoE and matched KL-Reg. At $\alpha = 0.5$, $\beta = 1.0$ the per-cell paired difference $|\mu_{\text{PoE}} - \mu_{\text{KL-Reg}}|$ in raw MuJoCo return is exactly 0 in all 18 harder-regime cells. The matched pair thus continues to produce bit-identical deterministic actions outside the main-body four-setting slice, confirming Sec. 7.1 under regime shift.

Critic-guided baselines. Among the critic-guided baselines, CQL-Guided with $\beta = 1.0$ is the strongest option in the harder-regime evaluation. It outperforms the Frozen actor in five cells, highlighted in Table 5, and gives the best result in nearly all medium-expert settings. The only exception is `halfcheetah-ME` under G_1, where all evaluated methods, including Frozen, produce negative returns.

IQL-Guided with $\beta = 1.0$ shows less stable behavior across medium-expert environments. In `halfcheetah-medium-expert` and `walker2d-medium-expert`, the IQL advantage gradient at $\mu_\theta(s)$ is numerically zero, so the update leaves the frozen action unchanged and matches the Frozen return up to numerical precision. In contrast, on `hopper-medium-expert`, the same update moves the policy away from the offline-data support region, leading to early termination and near-zero returns across all three goals. AWR exhibits the same sensitivity on hopper-type environments, where early-termination dynamics make critic-guided action updates unreliable.

Table 6: Per-cell raw-return means over 25 episodes at each α. *MR* / *ME* denote *-medium-replay-v2* / *-medium-expert-v2*. Bold marks the best α in each row; the verdict column uses the CI-half-width classifier of Appendix G. Medium-expert (lower block) is predominantly HURT across α values, with rare exceptions consistent with Table 7: a single FROZEN cell at $\alpha = 0.3$ and a small number of isolated HELP cells on hopper-ME (totalling 3 HELP, 1 FROZEN, 41 HURT across the 9×5 medium-expert cell–α combinations).

Env	Goal	Frozen	α=0.1	α=0.3	α=0.5	α=0.7	α=0.9	Best / verdict
halfcheetah-MR	G_1 speed	3309.9	1384.3	2466.1	2769.7	3585.2	**3731.8**	α=0.9 (+421.9, Frozen)
halfcheetah-MR	G_2 balanced	3309.9	2623.2	3237.4	3666.6	**3718.9**	3192.5	α=0.7 (+409.0, Frozen)
halfcheetah-MR	G_3 control	3309.9	1786.2	2097.1	2888.9	**3926.2**	3253.7	α=0.7 (+616.3, Help)
hopper-MR	G_1 speed	456.1	418.5	430.8	**460.0**	459.9	**460.0**	α=0.9 (+3.9, Frozen)
hopper-MR	G_2 balanced	456.1	211.7	423.6	436.4	458.3	**460.6**	α=0.9 (+4.4, Frozen)
hopper-MR	G_3 control	456.1	350.4	455.5	**456.4**	**456.4**	453.1	α=0.7 (+0.3, Frozen)
walker2d-MR	G_1 speed	581.5	115.6	154.3	171.9	231.2	**468.3**	α=0.9 ($-$113.2, Frozen)
walker2d-MR	G_2 balanced	581.5	279.2	**347.2**	261.2	290.3	174.8	α=0.3 ($-$234.3, Hurt)
walker2d-MR	G_3 control	581.5	**466.5**	311.2	280.0	292.4	123.2	α=0.1 ($-$115.0, Frozen)
halfcheetah-ME	G_1 speed	4796.0	**-395.0**	-396.8	-447.3	-487.6	-533.7	α=0.1 ($-$5191.1, Hurt)
halfcheetah-ME	G_2 balanced	4796.0	2967.5	3065.0	3922.3	3995.4	**4417.1**	α=0.9 ($-$378.9, Hurt)
halfcheetah-ME	G_3 control	4796.0	3433.8	3290.0	3585.9	**4451.0**	4359.0	α=0.7 ($-$345.0, Hurt)
hopper-ME	G_1 speed	1468.5	109.5	116.8	127.7	148.4	**286.0**	α=0.9 ($-$1182.5, Hurt)
hopper-ME	G_2 balanced	1468.5	141.0	191.5	282.8	**1577.5**	1114.1	α=0.7 (+109.0, Help)
hopper-ME	G_3 control	1468.5	**2561.9**	1401.6	1618.6	1318.0	1031.2	α=0.1 (+1093.4, Help)
walker2d-ME	G_1 speed	4919.8	**399.1**	72.2	61.8	63.3	77.7	α=0.1 ($-$4520.7, Hurt)
walker2d-ME	G_2 balanced	4919.8	97.4	144.4	128.0	**3594.9**	3422.4	α=0.7 ($-$1324.9, Hurt)
walker2d-ME	G_3 control	4919.8	1794.2	3421.3	**3596.1**	3512.6	3332.6	α=0.5 ($-$1323.7, Hurt)

J Harder-Regime α Sweep

This appendix gives the full α sweep referenced in Sec. 7.5 and visualizes the actor-competence ceiling on medium-expert. Per-cell raw returns for each $\alpha \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$ (and matched $\beta \in \{0.111, 0.429, 1.000, 2.333, 9.000\}$) are reported in Table 6; the full CSV is at `outputs/d4rl/harder_alpha_sweep/best_alpha_per_cell.csv`.

Equivalence of PoE and matched KL-Reg across α. At the exact center $\alpha = 0.5$ the paired $|\mu_{\text{PoE}} - \mu_{\text{KL-Reg}}|$ in rollout raw return is exactly 0 across all 18 harder-regime cells, confirming bit-identical deterministic actions. At the off-center matched pairs $\alpha \in \{0.1, 0.3, 0.7, 0.9\}$ the paired differences are nonzero but small relative to the inter-seed spread: mean absolute paired difference over all 90 off-center pairs is 94.9 and the maximum is 695.8 (`walker2d-ME` G_2 at $\alpha = 0.7$). These residuals are consistent with the MuJoCo chaotic-amplification story already documented in Appendix C; the corresponding per-seed paired CIs include zero in every cell.

Verdict summary by regime and α. Table 7 gives the verdict counts summarized in Figure 4. Medium-expert shows $8/9$, $8/9$, $8/9$, $8/9$, $9/9$ HURT at $\alpha = 0.1, 0.3, 0.5, 0.7, 0.9$. Medium-replay shows $8/9$, $7/9$, $4/9$, $3/9$, $2/9$ HURT, so the boundary moves strongly toward Frozen as α increases. The monotone pattern on medium-replay and the near-constant pattern on medium-expert together support reading the medium-expert result as a structural actor-competence ceiling rather than a bad knob choice.

Best-α distribution. Across the 18 sweep cells, the best-performing value of α is at least 0.7 in 12 cells and at most 0.3 in 5 cells. Only one cell selects $\alpha = 0.5$: `walker2d-ME` under G_3, which is classified as HURT. This distribution indicates that conservative composition weights are generally preferred in settings where composition remains viable.

The cases favoring the aggressive setting $\alpha = 0.1$ fall into two groups. The first consists of the two hopper-ME HELP cells, where the prior is stronger than the frozen actor on the control-weighted goal. The second consists of the three medium-expert G_1 speed cells, where all tested α values lead to collapse. Thus, low-α selection should be interpreted cautiously in cells where the entire sweep is unstable.

Table 7: Harder-regime α sweep: HELP/FROZEN/HURT verdict counts per regime and α, using the CI-half-width classifier of Appendix G. Each row uses 9 env-by-goal cells (three envs, three goals). Medium-expert is predominantly HURT across α (8 of 9 cells at $\alpha \in \{0.1, 0.3, 0.5, 0.7\}$ and 9 of 9 at $\alpha = 0.9$), with the only HELP cells on hopper-ME; medium-replay approaches Frozen ties as α rises.

	medium-replay					**medium-expert**				
α	0.1	0.3	0.5	0.7	0.9	0.1	0.3	0.5	0.7	0.9
HELP	0	0	0	1	0	1	0	1	1	0
FROZEN	1	2	5	5	**7**	0	1	0	0	**0**
HURT	8	7	4	3	2	8	8	8	8	**9**

K AntMaze Diagnostic

We ran the AntMaze cells under the same rollout protocol as the main body (five seeds $\times$ five episodes $\times$ three deployment goals, no artificial step cap, matched PoE/KL-Reg pairs and the full critic-guided baseline set). We treat this package as a diagnostic: the conclusion is that the frozen actor cannot solve the task, and therefore deployment-time refinement has no nonzero return to recover.

Setup. Environments: `antmaze-umaze-v2` and `antmaze-medium-play-v2`. Frozen actor: diagonal-Gaussian, trained by plain behavioral cloning (50 epochs, locomotion hyperparameters). Goal-conditioned prior: contrastive goal-weighted BC (80 epochs). The two mazes did not use the same training budget. On `antmaze-medium-play-v2`, the actor and prior were trained on the full 10^6-transition dataset and the held-out evaluator critic, CQL critic, and IQL value function for 30 epochs, as in the locomotion package. On `antmaze-umaze-v2`, the actor and prior were trained on a random subsample of 20,000 transitions (2% of the dataset) and the critics for 5 epochs; this reduced-budget umaze actor is the zero-success actor reported below. The reward decomposition follows the existing AntMaze decomposer in our codebase, which exposes goal directions `G1_success`, `G2_balanced`, and `G3_efficient`; the underlying sparse AntMaze reward is unchanged.

Zero success across all methods. On `antmaze-umaze-v2`, every one of the 8 methods (Frozen, Prior Only, Additive, PoE at $\alpha = 0.5$, matched KL-Reg, AWR-Critic, CQL-Guided, IQL-Guided) returns mean raw return 0.0 in all 75 episodes per method (3 goals $\times$ 5 seeds $\times$ 5 episodes). The same holds for `antmaze-medium-play-v2`. Episodes saturate at the full 700 (umaze) or 1,000 (medium-play) step cap in every seed; these BC actors never reach the goal region. The method-level KL-FROM-ACTOR ordering is still preserved as expected (Prior Only $>$ KL-Reg $>$ Additive $>$ PoE $>$ CQL/IQL/AWR/Frozen), confirming that the refinement mechanics operate correctly; they just have no competent anchor to improve.

Interpretation. This is the extreme case of the actor-competence ceiling of Sec. 7.5: on locomotion medium-expert the frozen actor produces nonzero return that composition damages; here the frozen actor produces zero return and composition inherits zero. The zero is a property of these particular actors, not of behavioral cloning on AntMaze: as shown next, BC trained on the full umaze dataset reaches 39–67% success.

Graded-competence study. To test the boundary between the zero-success extreme and competent actors, we froze six further actors on `antmaze-umaze-v2`, trained with the same code on the full dataset: BC checkpoints after 5, 15, 50, and 150 epochs (best validation loss within the budget), BC on a random 100k-transition subsample, and filtered BC on the 413 of 1,426 dataset episodes that reach the goal, truncated at the first success (137k transitions). Episodes were delimited by the dataset's timeout flags, since AntMaze sets the terminal flag on every step inside the goal region. Everything else was held fixed: the goal-conditioned prior and the critics are the umaze package above, and the protocol is the AntMaze protocol (PoE at $\alpha = 0.5$, matched KL-Reg, Additive at $\lambda = 0.5$, Prior Only, and the three critic-guided steps at $\beta = 1$; three goals $\times$ five seeds $\times$ five episodes). Table 8 reports success rates; verdicts use the classifier of Appendix G applied to success, with the half-width of a 95% bootstrap interval over the frozen actor's per-seed success rates (4.0–12.0 points) as the margin.

Table 8: Graded-competence AntMaze (`antmaze-umaze-v2`): task success (%) over 75 episodes per method, actors sorted by frozen-actor success. Superscripts: $^+$ HELP, $^-$ HURT, none FROZEN, relative to the same frozen actor. The prior and critics are fixed across rows. PoE and matched KL-Reg share deterministic means up to float precision; over 700-step episodes the resulting action jitter changes some outcomes, so their success rates differ by up to 8 points.

Frozen actor	Frozen	Prior Only	Additive	PoE	KL-Reg	Critic-grad.	CQL-guided	IQL-guided
Paper actor (20k subsample)	0	0	0	0	0	0	0	0
BC, 150 epochs	39	0^-	48	48	56^+	41	52^+	52^+
BC, 50 epochs	55	0^-	11^-	64	64	60	53	64
BC, 5 epochs	61	0^-	0^-	39^-	35^-	72^+	60	71^+
BC, 15 epochs	63	0^-	3^-	48^-	47^-	61	59	69
Filtered BC (goal-reaching)	65	0^-	7^-	69	65	56	60	76^+
BC, 100k subsample	67	0^-	0^-	24^-	24^-	72	75^+	63

Three findings. (i) With a zero-success actor, every method stays at zero; with any of the competent actors, PoE/KL-Reg retain substantial success (24–69%), whereas Prior Only collapses to 0% for every actor and Additive is HURT for five of six. Precision-weighted composition therefore keeps the anchor that the other prior-driven rules lose. (ii) Composition does not improve these actors. PoE is FROZEN for three actors and HURT for three, and never HELP; the only gains are on the weakest competent actor (39% frozen: PoE 48%, KL-Reg 56%). (iii) In the rebuttal we predicted that composition outcomes would improve monotonically with frozen-actor success. This prediction is not supported: PoE's change relative to the frozen actor becomes more negative, not less, as actor success rises (Spearman -0.71, $p = 0.11$, six actors). One plausible reason is that the fixed prior is weak (Prior Only 0% for every actor): a competent actor has more to lose from being steered toward it, while the weakest actor has the most headroom. The critic-guided steps are never HURT here and are HELP in 6 of 18 actor–method pairs, including CQL- and IQL-guided on the weakest actor, which is consistent with Sec. 8: when a reliable critic is available, critic-guided steering is a strong option. Running a task that demands exploration beyond the offline data with a frozen actor that cannot reach the goal remains outside the regime the paper claims.

L Adroit Manipulation

To test the framework outside locomotion and navigation, we ran the identical protocol on the D4RL Adroit tasks `pen-human-v1` and `hammer-human-v1`: a diagonal-Gaussian BC actor (50 epochs) frozen after training, the contrastive goal-conditioned prior of Eq. (3) (80 epochs), the same held-out, CQL, and IQL critics, five seeds $\times$ five episodes per method, goal, and seed, deterministic mean actions, and the classifier of Appendix G. The Adroit reward is a shaped task reward without a control term, so we use the same three channels as Eq. (1) with the task reward in the first channel: $rc = (r,\ 10^{-3}\|a\|^2,\ 1)$, and the same goals G_1, G_2, G_3. Task success follows the DAPG convention (goal achieved for more than 20 of 100 steps on pen and more than 25 of 200 on hammer).

The two tasks fall on opposite sides of the actor-competence boundary, which is what the framework predicts. The frozen pen actor is moderately competent (41.9 D4RL-normalized, 28% success); the frozen hammer actor is not (0.3 normalized, 0% success).

Pen: composition preserves a competent actor but does not improve it. The best composition operating point is PoE/KL-Reg at $\alpha = 0.9$ on all three goals, and it is FROZEN on all three: within the frozen actor's confidence interval (success 28–32% versus 28%). Stronger steering hurts ($\alpha \leq 0.5$ is HURT on every goal), and Prior Only and Additive are HURT on every goal, with Additive at 0% success. This mirrors the locomotion picture: precision-weighted composition stays anchored to a competent actor, whereas steering that moves far from it destroys task success. The critic-guided baselines are not uniformly safer: the critic-gradient and CQL-guided steps are HURT on G_1 (0% success), although CQL-guided reaches the highest success on G_3 (36%).

Hammer: no method recovers an incompetent actor. Every method, composition and critic-guided alike, has 0% success on every goal, as on AntMaze. The classifier marks a few hammer cells HELP or HURT on the goal-weighted return (e.g., PoE at $\alpha = 0.7$ on G_2), but these are small

Table 9: Adroit rollouts: mean goal-weighted return (task success in %) over 25 episodes per cell. Superscripts give the classifier verdict against the frozen actor: $^{+}$ HELP, $^{-}$ HURT, none FROZEN. PoE at α and KL-Reg at $\beta = \alpha/(1-\alpha)$ share a row: their returns are bit-identical at $\alpha = 0.5$ and differ by at most 0.08 in any cell mean at the other α (float-precision jitter); PoE values are shown.

	pen-human-v1			hammer-human-v1		
Method	G_1	G_2	G_3	G_1	G_2	G_3
Frozen	1354.5 (28)	722.7 (28)	145.5 (28)	-215.2 (0)	-16.9 (0)	-2.0 (0)
Prior Only	261.6^{-} (12)	180.8^{-} (8)	32.8^{-} (8)	-214.7 (0)	-17.0 (0)	-1.6^{+} (0)
Additive (λ=0.5)	101.6^{-} (0)	86.6^{-} (0)	17.3^{-} (0)	-214.8 (0)	-16.9 (0)	-1.9 (0)
PoE/KL-Reg α=0.1	265.1^{-} (8)	164.7^{-} (8)	29.6^{-} (8)	-214.7 (0)	-16.9 (0)	-1.6^{+} (0)
PoE/KL-Reg α=0.3	134.9^{-} (4)	112.0^{-} (4)	24.1^{-} (4)	-214.7 (0)	-16.8 (0)	-1.7^{+} (0)
PoE/KL-Reg α=0.5	617.8^{-} (12)	356.8^{-} (12)	70.0^{-} (12)	-214.9 (0)	-17.3 (0)	-2.0 (0)
PoE/KL-Reg α=0.7	853.6 (12)	581.1 (20)	94.4 (12)	-215.7 (0)	-2.2^{+} (0)	-2.4^{-} (0)
PoE/KL-Reg α=0.9	1378.6 (28)	778.4 (32)	156.8 (28)	-219.3^{-} (0)	-19.1^{-} (0)	-2.7^{-} (0)
Critic-gradient (β=1)	85.0^{-} (0)	481.7 (16)	101.9 (20)	-221.2^{-} (0)	-19.5^{-} (0)	-2.1 (0)
CQL-guided (β=1)	88.7^{-} (0)	505.7 (20)	162.5 (36)	-218.0^{-} (0)	-18.0^{-} (0)	-2.5^{-} (0)
IQL-guided (β=1)	1351.9 (28)	619.7 (20)	120.9 (20)	-215.2 (0)	-16.9 (0)	-2.6^{-} (0)
Frozen CI half-width ϵ	685.4	342.7	68.6	1.1	0.5	0.1

shifts in the shaped reward that never translate into task success, so we do not read them as successful steering. Whether composition can raise success requires an actor with nonzero competence, which the graded-competence AntMaze study in Appendix K varies directly.

M AWR-Trained Prior

To test how much the results depend on the prior's training objective, we replaced the contrastive prior of Eq. (3) by a goal-conditioned prior of the same architecture trained by advantage-weighted regression on dataset actions only:

$$\mathcal{L}_{\mathrm{AWR}}(\phi) = -\mathbb{E}\big[\min\big(\exp(\hat{A}_g(s,a)),\, 20\big)\log \rho_\phi(a \mid s, g)\big], \qquad A_g(s,a) = Q_g(s,a) - Q_g\big(s, \mu_\theta(s)\big),$$

where Q_g is the goal-conditioned critic of the frozen actor used throughout the paper, $\hat{A}_g$ is A_g standardized per batch, and goals are sampled with the critic-training mixture (80 epochs, batch 512). The prior is thus a weighted regression on actions present in the dataset; the critic is evaluated only on dataset actions and on the frozen actor's mean. The frozen actors, goals, protocol (five seeds $\times$ five episodes), and classifier are those of the headline package, and the frozen-actor returns reproduce the headline package exactly.

With the AWR prior the split is 4/8/0 HELP/FROZEN/HURT, against 4/5/3 with the contrastive prior. The three HURT cells of the contrastive prior, all G_1 speed cells, are no longer hurt: Hopper medium G_1 and Walker2d medium G_1 become FROZEN ($+7.5$, $+163.2$), and Hopper medium-expert G_1 becomes HELP ($+117.3$ instead of -878). The anchoring property is visible again: Prior Only with the AWR prior collapses on Hopper medium-expert (return 8–30) and on Walker2d medium G_1 (552 versus 2,431), yet precision-weighted composition with the same prior is HELP on all three Hopper medium-expert cells and FROZEN on Walker2d medium G_1. The best operating point is $\alpha = 0.9$ in 8 of 12 cells, i.e., close to the actor.

This result revises a statement we made during the review discussion, that the G_1 failures persist under every prior we had tried and therefore reflect actor and task geometry alone. The prior variants we had tried were all degradations of the contrastive prior; a prior trained with a different objective changes the outcome. The HURT cells of the headline package therefore reflect the interaction between a near-ceiling actor and the contrastive prior, not the actor alone. Two qualifications apply. The AWR prior uses the frozen actor's critic, so it carries more information about the actor's own value than the contrastive prior, which uses only dataset rewards. We tested it on the 12 headline cells only; the medium-expert and AntMaze conclusions in Sec. 7.5 rest on the contrastive prior.

Table 10: Headline cells with an AWR-trained prior. "Prior Only" and "Best composition" use the AWR prior; the last column repeats the five-seed verdict and Δ obtained with the contrastive prior (Sec. 7.2). Goal-weighted MuJoCo rollout returns, 25 episodes per cell.

Environment	Goal	Frozen	Prior Only	Best composition	Δ	Verdict	Contrastive prior
HalfCheetah-M	G_1	5010.6	5176.7	5382.4 (α=0.9)	+371.7	Frozen	Frozen (+248)
HalfCheetah-M	G_2	3113.4	3243.5	3323.5 (Add.)	+210.1	Frozen	Frozen (+167)
HalfCheetah-M	G_3	938.9	969.0	997.8 (α=0.3)	+58.9	Frozen	Frozen (+29)
Hopper-M	G_1	1180.5	1193.0	1188.0 (α=0.9)	+7.5	Frozen	Hurt (-291)
Hopper-M	G_2	822.8	737.8	849.7 (α=0.9)	+26.8	Frozen	Help (+144)
Hopper-M	G_3	165.0	128.8	168.1 (α=0.3)	+3.1	Frozen	Frozen (-5)
Walker2d-M	G_1	2431.3	551.9	2594.5 (α=0.9)	+163.2	Frozen	Hurt (-2030)
Walker2d-M	G_2	1610.2	249.5	1864.9 (α=0.7)	+254.8	Help	Frozen (+124)
Walker2d-M	G_3	324.2	129.5	352.3 (α=0.9)	+28.1	Frozen	Help (+55)
Hopper-ME	G_1	1046.8	24.3	1164.1 (α=0.9)	+117.3	Help	Hurt (-878)
Hopper-ME	G_2	734.7	30.0	864.8 (α=0.9)	+130.1	Help	Help (+55)
Hopper-ME	G_3	147.4	8.2	170.9 (α=0.9)	+23.5	Help	Help (+110)

N Successor-Feature / GPI Baseline

This appendix describes the SF/GPI baseline referenced in Sec. 7.5. The purpose of this baseline is to evaluate whether a learned successor-feature representation combined with generalized policy improvement (SF/GPI) [Barreto et al., 2017, 2018] can serve as an alternative to the density-anchored PoE/KL-Reg deployment rule.

Implementation. We train a successor-feature network $\psi(s, a)$ using the same offline transitions as the other D4RL experiments. The network is optimized with the squared Bellman target

$$\mathbb{E}\left[\left\|\psi(s,a) - \phi(s,a) - \gamma\psi(s',a')\right\|^2\right],$$

where $\phi(s, a)$ contains the three reward components associated with G_1, G_2, and G_3 from Sec. 6. The target action is set to $a' = \mu_\theta(s')$, matching the deterministic actor convention used in the rollout evaluation. The network is implemented as a two-layer MLP with hidden width 256 and is trained for 30 epochs using batch size 1024, learning rate 10^{-4}, discount factor $\gamma = 0.99$, and Polyak target update rate $\tau = 5 \times 10^{-3}$.

At deployment, SF/GPI selects actions from a finite candidate set constructed at each state. The set contains the actor mean, four PoE samples, four prior samples, and eight uniformly sampled actions, for a total of 16 candidates. Each candidate action is scored by $g^\top \psi(s, a)$ under the current deployment goal g, and the highest-scoring action is selected. This procedure applies GPI over a restricted candidate set intended to remain close to the offline-data support.

Rollout protocol. We evaluate SF/GPI on the four main D4RL task settings using the same rollout protocol as the main experiments: five seeds, five episodes per seed, and three deployment goals per environment. The same matched-α comparisons and baseline reporting conventions are used for consistency. On `halfcheetah`, SF/GPI achieves returns close to the Frozen actor across goals. On `hopper` and `walker2d`, however, SF/GPI is less stable than density-anchored composition and exhibits early-termination failures in several cells, similar to the behavior observed for AWR.

Across the 12 main-body setting–goal cells, SF/GPI outperforms the Frozen actor in 2 cells, remains within the Frozen confidence margin in 4 cells, and falls below Frozen by more than one confidence-interval half-width in 6 cells.

Verdict. SF/GPI provides a useful additional baseline, but in this setting it does not consistently improve over the Frozen actor or the density-anchored composition methods. Under the offline-trained successor-feature model and the three-goal reward decomposition used here, SF/GPI is sensitive to candidate-action selection and can suffer from early-termination collapse in locomotion environments. For this reason, we report SF/GPI as an appendix-only comparison and keep the main harder-regime evaluation focused on PoE, KL-Reg, Additive, and critic-guided baselines.

Table 11: Multi-environment goal-responsiveness validation on D4RL benchmarks. The frozen actor remains goal-invariant, while PoE exhibits nonzero goal-dependent movement across all environments.

Environment	PoE Shift ↑	Frozen Shift	KL Max	KL Min	Prior Sensitivity
`halfcheetah-medium-v2`	0.4833	0.0000	26.5855	0.9752	0.5734 / 0.8677
`hopper-medium-v2`	0.3861	0.0000	8.6033	3.6753	0.4993 / 0.6801
`walker2d-medium-v2`	0.5039	0.0000	16.2352	5.5264	0.2157 / 0.2280
`hopper-medium-expert-v2`	0.7663	0.0000	29.4718	5.5504	0.1009 / 0.1885

Table 12: Compact Pareto summary on `halfcheetah-medium-v2`. Each entry reports goal-conditioned return / KL divergence from the frozen actor. Higher return is better; lower KL is better.

Goal	Frozen	Prior Only	Additive	KL-Reg	PoE	Best return	Lowest adaptive KL
G1_speed	5.402 / 0.000	5.445 / 1.062	5.442 / 0.858	5.443 / 0.689	5.442 / 0.670	Prior Only (5.445)	PoE (0.670)
G2_balanced	3.324 / 0.000	3.348 / 1.058	3.346 / 0.855	3.347 / 0.689	3.346 / 0.670	Prior Only (3.348)	PoE (0.670)
G3_control	1.011 / 0.000	1.022 / 1.015	1.020 / 0.819	1.021 / 0.695	1.021 / 0.665	Prior Only (1.022)	PoE (0.665)

O Additional Results

The following subsections are retained as auxiliary analyses that were run on the off-policy evaluator described in Appendix B. They are diagnostic-only and their numbers should not be compared directly to the rollout-based headline results in the main body. The main paper's empirical claims are based on the MuJoCo rollout package (Sec. 7, Appendices C-G).

O.1 Note on Adaptation Efficiency

We do not use an adaptation-efficiency metric based on return gain per unit KL deviation from the frozen actor. This normalization is not invariant to the parameterization of the composition rule. Under the reparameterization $\beta = \alpha/(1-\alpha)$, PoE and KL-Reg produce the same deterministic action. Their posterior covariances differ only by a global scalar, equal to $(1+\beta)/\beta = 1/\alpha$ under the standard convention. This scalar changes the stochastic KL from the frozen actor but leaves the deployed deterministic action unchanged. Therefore, dividing return gain by this KL can introduce an artificial distinction between PoE and KL-Reg that reflects the covariance convention rather than a behavioral difference. For this reason, Fig. 2 reports goal-weighted returns with 95% confidence intervals, while Appendix C reports matched-α paired return differences directly.

O.2 Multi-Environment Goal Responsiveness

This subsection checks whether the goal-conditioning signal remains visible beyond the headline environment and whether the frozen actor stays invariant by construction. Table 11 records the shift statistics across four D4RL task settings.

The frozen actor remains exactly goal-invariant, while PoE shows nonzero movement in every environment, confirming that the refinement step induces genuine goal-conditioned adaptation rather than evaluator noise.

O.3 Exact Pareto Operating Points

This subsection reports operating points from the off-policy evaluator on `halfcheetah-medium-v2` for reference. Under the equivalence of Sec. 7.1, PoE and KL-Reg share the same optimal deterministic policy under $\alpha = \beta/(1+\beta)$; the difference in the "lowest adaptive KL" column arises from the stochastic-covariance scaling between the two rules, not from any policy-level difference. The Pareto front (return vs. deviation) is therefore the same for both rules; only the reported KL units differ. The rollout-based headline is Figure 2.

The reported KL values for PoE and KL-Reg on each row differ only because the two rules report stochastic KLs under different posterior-covariance conventions (Sec. 4.3); the underlying optimal

Table 13: **Diagnostic only (off-policy evaluator scores; not rollout returns).** Adaptive-α selection on `halfcheetah-medium-v2`. We compare a KL-budget rule against an oracle α chosen on the test split. Lower selection loss is better.

Goal	Rule	Sel. α	Oracle α	Test Ret	Oracle Ret	Sel. Loss	Test KL	Cat%	Frozen Ret	Prior Ret
G1_speed	KL $\kappa = 0.1$	0.80	0.05	5.4432	5.4636	0.0203	0.054	0.45	5.4344	5.4640
	KL $\kappa = 0.3$	0.50	0.05	5.4508	5.4636	0.0128	0.208	0.25	5.4344	5.4640
	KL $\kappa = 0.5$	0.30	0.05	5.4542	5.4636	0.0093	0.372	0.10	5.4344	5.4640
	KL $\kappa = 1.0$	0.05	0.05	5.4636	5.4636	0.0000	0.782	0.10	5.4344	5.4640
	Val-best	0.05	0.05	5.4636	5.4636	0.0000	0.782	0.10	5.4344	5.4640
G2_balanced	KL $\kappa = 0.1$	0.80	0.05	3.3446	3.3554	0.0109	0.054	0.35	3.3393	3.3561
	KL $\kappa = 0.3$	0.50	0.05	3.3491	3.3554	0.0064	0.209	0.20	3.3393	3.3561
	KL $\kappa = 0.5$	0.30	0.05	3.3508	3.3554	0.0046	0.372	0.10	3.3393	3.3561
	KL $\kappa = 1.0$	0.05	0.05	3.3554	3.3554	0.0000	0.780	0.00	3.3393	3.3561
	Val-best	0.05	0.05	3.3554	3.3554	0.0000	0.780	0.00	3.3393	3.3561
G3_control	KL $\kappa = 0.1$	0.80	0.05	1.0152	1.0211	0.0059	0.058	1.45	1.0120	1.0209
	KL $\kappa = 0.3$	0.50	0.05	1.0180	1.0211	0.0031	0.218	1.35	1.0120	1.0209
	KL $\kappa = 0.5$	0.30	0.05	1.0191	1.0211	0.0019	0.380	1.20	1.0120	1.0209
	KL $\kappa = 1.0$	0.05	0.05	1.0211	1.0211	0.0000	0.763	0.85	1.0120	1.0209
	Val-best	0.05	0.05	1.0211	1.0211	0.0000	0.763	0.85	1.0120	1.0209

deterministic policy is identical, with $\frac{1+\beta}{\beta} = \frac{1}{\alpha}$ relating the covariance scalings. Pareto-front equivalence between the two rules therefore holds, with only the KL axis units changing. Prior-Only reaches a slightly higher return on this off-policy scorer at the cost of a much larger deviation from the frozen actor.

O.4 Adaptive-α Selection

This subsection examines how to select the refinement strength without relying on oracle access to test-time returns. Table 13 compares two practical selection rules: a KL-budget rule and a validation-best rule. It reports the selected operating points on `halfcheetah-medium-v2`, which serves as a compact example because the selector closely tracks the oracle choice in this environment. The same comparison is applied to all four task settings in the accompanying evaluation results.

The selection loss is small for all KL budgets and vanishes at the loosest budget, where the selected α matches the oracle. Validation-best also recovers the oracle operating point in this setting. Across the full four-setting CSV, the KL-budget selector typically lands on a neighboring grid point and keeps the selection loss small, so the main lesson is operational rather than algorithmic.

O.5 Support Mismatch and Contradictory Goals

This subsection tests the boundary of applicability by contrasting supported, extrapolative, and contradictory goals on `halfcheetah-medium-v2`. Table 14 reports how the adaptive methods behave when the new goal aligns with, exceeds, or contradicts actor/data support.

In these off-policy diagnostic values, supported goals improve over the frozen actor and the contradictory reverse-motion goal hurts every adaptive method. The PoE and KL-Reg rows in Table 14 correspond to different operating points rather than matched deterministic policies, so their relative comparison should not be interpreted as a policy-equivalence result. Instead, the key takeaway is their qualitative separation from Prior Only and Additive. The broader implication is that directional contradiction with the actor's support matters more than goal magnitude alone. The rollout-based boundary classification is in Sec. 7.2 and Appendix G.

O.6 Robustness to Deployment-Prior Degradation

The aggregate rollout-based degradation summary is presented in the main paper (Sec. 7.4 and Fig. 3). Figure 5 gives the same MuJoCo rollouts per goal instead of averaged across goals, at the same operating point ($\alpha = 0.5$, $\lambda = 0.5$). It shows that the low trained-prior values in Fig. 3 come mainly from the G_1 speed cells, where the frozen actor is near its velocity ceiling and moving toward the trained contrastive prior costs return (e.g., hopper-medium-expert G_1: -925 under the trained prior

Table 14: **Diagnostic only (off-policy evaluator scores; not rollout returns).** Support-mismatch analysis on `halfcheetah-medium-v2`. In-support goals benefit from adaptation, while a contradictory reverse-motion goal exposes a clear failure mode. For the standard supported goal G_1 and the contradictory failure case G_{reverse}, we report all principal adaptive baselines; for G_2 and G_{extreme}, we report the frozen actor and PoE as representative comparisons because the main purpose is to contrast supported versus unsupported goal directions. *Note:* the PoE and KL-Reg rows in this table are *not* matched via $\beta = \alpha/(1-\alpha)$ (the matched β for $\alpha = 0.9$ is 9.0, not the $\beta = 5.0$ shown); they should be read as separate operating points within each family rather than as the same deterministic policy. Δ is measured relative to the frozen actor.

Goal	Method	Setting	Return	Cat%	Supp. Dist	Δ vs Frozen
G_1 speed	Frozen Actor	–	5.4018	0.76	0.351	–
	Prior Only	–	5.4454	0.13	0.348	+0.044
	PoE	$\alpha = 0.1$	5.4424	0.15	0.346	+0.041
	KL-Reg	$\beta = 0.1$	5.4428	0.14	0.346	+0.041
	Additive Mix	$\lambda = 0.3$	5.4337	0.16	0.348	+0.032
G_2 balanced	Frozen Actor	–	3.3238	0.91	0.351	–
	PoE	$\alpha = 0.1$	3.3465	0.21	0.346	+0.023
G_{reverse}	Frozen Actor	–	-5.1249	0.48	0.351	–
	Prior Only	–	-5.1693	0.57	0.343	-0.044
	PoE	$\alpha = 0.9$	-5.1352	0.48	0.347	-0.010
	KL-Reg	$\beta = 5.0$	-5.1399	0.50	0.345	-0.015
	Additive Mix	$\lambda = 0.7$	-5.1409	0.46	0.348	-0.016
G_{extreme}	Frozen Actor	–	26.317	0.75	0.351	–
	PoE	$\alpha = 0.1$	26.518	0.14	0.346	+0.201

versus -58 under the nearly inert random prior), while under the same random prior Prior Only and Additive collapse on every goal.

This appendix subsection keeps the detailed per-goal breakdown in Table 20 from the off-policy diagnostic evaluator, which shows how the same qualitative robustness pattern manifests within each supported goal.

The detailed breakdown confirms the main aggregate story: noisy and undertrained priors are manageable for PoE because actor anchoring suppresses drift, while random priors are the sharpest stress test for prior-only adaptation.

O.7 Actor-Quality Sensitivity

This subsection tests whether the frozen actor remains a useful anchor across weak, medium, and strong behavioral-cloning checkpoints. The aggregate multi-environment results have been moved into the main paper; Table 21 provides the detailed single-environment breakdown on `halfcheetah-medium-v2`.

Table 21 shows the same pattern at the per-goal level: PoE and KL-Reg stay close to the frozen actor while improving return, and the stronger anchors yield the best-conserved operating points. This is the appendix counterpart to the main-body actor-quality summary, and it matters because it shows that the refinement rule remains useful when the frozen actor is already behaviorally reliable.

O.8 Extended Baseline Comparison under the Diagnostic Evaluator

This subsection reports an auxiliary diagnostic comparison using the original deployment critic instead of the held-out scorer Q_g^{eval}. Since this critic also contributes to the construction of critic-aware baselines, the results are used only to examine within-protocol behavior and are not used for the main cross-method conclusions.

Under the diagnostic critic, critic-maximizing baselines achieve the highest estimated returns by design. The same critic also defines the catastrophic-return and constraint-violation thresholds,

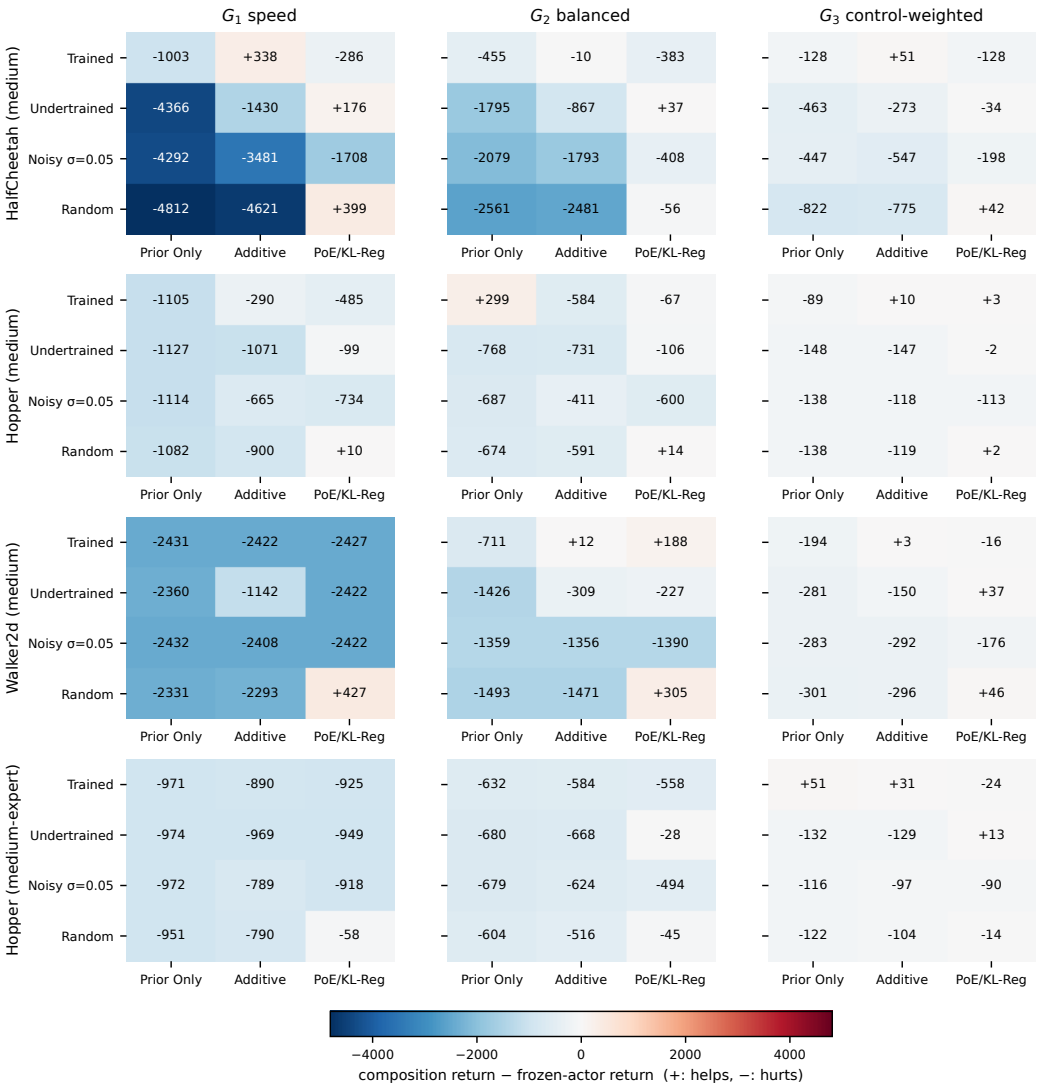

Figure 5: Per-goal prior-degradation rollouts (the per-goal version of Fig. 3). Each cell is the mean goal-weighted return of a composition family minus the frozen-actor return for one environment, goal, and prior variant (three seeds × three episodes). Red (positive): composition helps; blue (negative): composition hurts. PoE at $\alpha = 0.5$ and matched KL-Reg at $\beta = 1.0$ are the same deterministic policy and share a column.

making those lower-tail metrics uninformative for methods that directly optimize it. Within the refinement-prior family, the refinement-network variant shows a stronger lower-tail profile, while the critic-prior variant remains closer to the frozen actor in action total variation. This pattern is consistent with the stability trends reported in Appendix O.13.

We also isolate the BC evaluation-mode effect in Table 15. The BC policy and the pretrained actor use identical weights; the only difference is the action-selection rule. When evaluated with stochastic sampling under the held-out scorer, BC closely matches the pretrained actor. This indicates that the strong lower-tail metrics observed for argmax BC mainly arise from the deterministic evaluation mode, rather than from BC being inherently safer than PoE.

Table 15: BC evaluation-mode audit under the held-out evaluator. BC and the pretrained actor share identical weights; differences arise only from argmax versus stochastic action selection.

Mode	Return ↑	Cat% ↓	Con% ↓	Rob ↑
BC (argmax)	0.0322	**0.0000**	**0.0000**	**1.0000**
BC (sample)	0.0305	0.0702	0.0443	0.9077
Pretrained Actor (Table 17)	0.0306	0.0720	0.0491	0.9035
PoE Refinement Prior (Table 17)	0.0301	0.0877	0.0515	0.8866

Table 16: Performance under four shift regimes using the held-out evaluator. WorstRet and WorstRisk aggregate the minimum return and maximum catastrophic rate across all four conditions per method.

Method	Temporal	Subgroup	Syn(mod)	Syn(sev)	WorstRet ↑	WorstRisk ↓
Pretrained Actor	0.0307	0.0306	0.0306	0.0305	0.0305	0.0743
Critic-Greedy	**0.0541**	**0.0541**	**0.0541**	**0.0541**	**0.0541**	**0.0000**
CQL	0.0325	0.0321	0.0321	0.0321	0.0321	**0.0000**
IQL	0.0325	0.0321	0.0321	0.0321	0.0321	**0.0000**
PoE (Refinement Prior)	0.0297	0.0301	0.0300	0.0296	0.0296	0.1078
PoE (Critic Prior)	0.0300	0.0302	0.0301	0.0302	0.0300	0.1200

O.9 Extended Distribution Shift Experiments

Table 16 reports the temporal and subgroup shift results. We further evaluate two synthetic perturbation regimes, denoted $\mathrm{Syn(mod)}$ and $\mathrm{Syn(sev)}$, to assess whether the actor-preserving refinement methods remain stable under broader deployment mismatch.

The extended results show two consistent patterns. First, actor-preserving methods stay within a narrow return range across all four shift settings, indicating that the refinement mechanism remains stable under the tested perturbations. Second, the relative ordering of methods is preserved as the synthetic shift becomes stronger: objective-aware reference methods achieve higher returns, while actor-preserving methods maintain stable behavior but exhibit higher lower-tail risk. Within the PoE family, the refinement-network prior again provides the strongest worst-case risk profile.

O.10 Validated Multi-Environment Exact Comparisons

The tables in this subsection report per-environment values from the off-policy diagnostic evaluator. These results should be interpreted in light of the deterministic-action equivalence described in Sec. 7.1. Under the matched parameterization $\beta = \alpha/(1-\alpha)$, PoE and KL-Reg induce the same deterministic policy. Therefore, apparent PoE-versus-KL-Reg win counts, such as PoE exceeding KL-Reg in 9 of 12 cells, should not be interpreted as policy-level improvements. Such differences arise from numerical precision effects and the stochastic-KL covariance convention.

The rollout-based matched comparison is reported in Appendix C. The comparisons against Prior-Only and Additive remain useful for assessing robustness and support mismatch, so those qualitative trends are retained below.

Appendix interpretation. The count summary and exact-value tables play complementary roles. The count table provides the reviewer-facing pattern at a glance: PoE is favorable in most main-comparison cells and especially strong under degraded priors and support mismatch. The exact tables then show that these counts are not driven by a single anomalous cell. `halfcheetah-medium-v2` is a clear strength case, `hopper-medium-v2` and `walker2d-medium-v2` are broadly favorable but tighter, and `hopper-medium-expert-v2` is the main boundary case where the frozen actor is already strong and KL-Reg can be slightly better in raw return.

O.11 Detailed Single-Environment D4RL Results

For completeness, we provide detailed single-environment results on `halfcheetah-medium-v2` corresponding to the aggregate multi-environment summaries reported in the main text. In particular,

Table 17: **Diagnostic only (off-policy evaluator scores; not rollout returns).** Per-environment scores from the off-policy evaluator for $\mathrm{PoE}(\alpha = 0.5)$ and matched $\mathrm{KL\text{-}Reg}(\beta = 1.0)$. Under the equivalence of Sec. 7.1 these two rules produce identical deterministic mean actions, so the Δ Return column is float-precision residual from the evaluator pipeline (different code paths consume the same mean action). The Δ KL column instead reports a stochastic KL difference: PoE and KL-Reg report covariances that differ by a global scalar (Sec. 4.3), and stochastic-KL is sensitive to that scaling even though deterministic actions match. The Δ Eff column is therefore convention-dependent and is not a policy-level difference; we retain the table for completeness, and no main-body conclusion depends on these deltas. Diagnostic scores must not be compared directly to MuJoCo rollout returns elsewhere in the paper.

Environment	Goal	PoE Ret.	PoE KL	PoE Eff.	KL-Reg Ret.	KL-Reg KL	KL-Reg Eff.	Δ Ret.	Δ KL	Δ Eff.
halfcheetah-medium-v2	G1_speed	5.4439	0.3901	0.1037	5.4426	0.5720	0.0683	**0.0014**	**-0.1819**	**0.0354**
halfcheetah-medium-v2	G2_balanced	3.3471	0.3908	0.0573	3.3464	0.5750	0.0376	**0.0007**	**-0.1843**	**0.0196**
halfcheetah-medium-v2	G3_control	1.0211	0.4004	0.0253	1.0207	0.6176	0.0159	**0.0003**	**-0.2172**	**0.0094**
hopper-medium-v2	G1_speed	2.1844	4.9859	0.0007	2.1844	5.2765	0.0006	0.0000	**-0.2906**	0.0000
hopper-medium-v2	G2_balanced	1.5532	2.0902	0.0058	1.5530	2.0812	0.0058	**0.0002**	**0.0090**	**0.0001**
hopper-medium-v2	G3_control	0.3146	2.0988	0.0026	0.3144	2.1429	0.0024	**0.0002**	**-0.0441**	**0.0002**
walker2d-medium-v2	G1_speed	2.9037	15.2341	0.0224	2.8996	14.7954	0.0228	**0.0040**	**0.4387**	**-0.0004**
walker2d-medium-v2	G2_balanced	1.8384	5.7204	0.0186	1.8363	5.7740	0.0180	**0.0021**	**-0.0537**	**0.0005**
walker2d-medium-v2	G3_control	0.3684	5.2862	0.0037	0.3679	5.3672	0.0035	**0.0006**	**-0.0810**	**0.0002**
hopper-medium-expert-v2	G1_speed	2.2731	29.4065	-0.0062	2.2817	27.6563	-0.0063	**-0.0085**	**1.7502**	**0.0001**
hopper-medium-expert-v2	G2_balanced	1.6441	2.8258	-0.0120	1.6487	3.3361	-0.0088	**-0.0047**	**-0.5103**	**-0.0032**
hopper-medium-expert-v2	G3_control	0.3334	1.2618	-0.0025	0.3337	2.1804	-0.0013	**-0.0004**	**-0.9186**	**-0.0012**

Table 18: **Diagnostic only (off-policy evaluator scores; not rollout returns).** Per-environment random-prior comparison from the offline diagnostic evaluator (appendix-only). Under a random deployment prior, $\mathrm{PoE}(\alpha = 0.5)$ and matched $\mathrm{KL\text{-}Reg}(\beta = 1.0)$ produce the same deterministic action, so the rows labeled PoE and KL-REG report the same goal-weighted score up to the stochastic-KL convention used by each rule; reported stochastic KLs differ by the global covariance scalar of Sec. 4.3. Both are substantially more robust than PRIOR ONLY. The rollout-based version of this finding is in Sec. 7.4 and Appendix O.6.

Environment	Goal	Prior Only Ret.	Prior Only KL	KL-Reg Ret.	KL-Reg KL	PoE Ret.	PoE KL
halfcheetah-medium-v2	G1_speed	3.1497	1057.1792	5.4019	2.4515	**5.4024**	**0.0162**
halfcheetah-medium-v2	G2_balanced	2.0957	1056.9600	3.3232	2.4515	**3.3238**	**0.0162**
halfcheetah-medium-v2	G3_control	0.5617	1056.9503	1.0093	2.4515	**1.0100**	**0.0162**
hopper-medium-v2	G1_speed	2.0577	545.0012	2.1791	1.2306	**2.1800**	**0.0077**
hopper-medium-v2	G2_balanced	1.4780	545.1620	1.5400	1.2306	**1.5405**	**0.0077**
hopper-medium-v2	G3_control	0.2957	545.0763	0.3090	1.2306	**0.3091**	**0.0077**
walker2d-medium-v2	G1_speed	1.4167	1029.7469	2.5592	2.4520	**2.5604**	**0.0160**
walker2d-medium-v2	G2_balanced	1.1591	1029.7535	1.7309	2.4520	**1.7315**	**0.0160**
walker2d-medium-v2	G3_control	0.2327	1029.7090	0.3488	2.4520	**0.3490**	**0.0160**
hopper-medium-expert-v2	G1_speed	2.3544	2228.3518	2.4632	4.4778	**2.4627**	**4.1743**
hopper-medium-expert-v2	G2_balanced	1.6268	2227.9148	1.6820	4.4775	**1.6817**	**4.1740**
hopper-medium-expert-v2	G3_control	0.3255	2228.0630	0.3371	4.4775	**0.3370**	**4.1739**

Table 20 expands the prior-degradation study by showing the per-goal behavior under trained, noisy, and random deployment priors, together with the corresponding best PoE operating point. Likewise, Table 21 expands the actor-quality sensitivity study by reporting per-goal results for weak, medium, and strong frozen actors, together with the corresponding PoE and KL-regularized improvements.

These detailed tables serve two purposes. First, they verify that the aggregate trends in the main text are not driven by a single averaged configuration, but are consistently reflected across the individual supported goals in the environment. Second, they make the operating trade-offs more transparent by exposing the exact return, catastrophic rate, KL deviation, and return-improvement values that are compressed in the aggregate summaries. Together, Table 20 and Table 21 therefore provide the single-environment evidence underlying the broader D4RL robustness and actor-quality conclusions discussed in the main paper.

The table shows that the same aggregate pattern appears at the per-goal level: noisy and undertrained priors are still manageable for PoE, while the random-prior setting is where prior-only adaptation fails most sharply.

Table 19: Exact support-mismatch comparison using the contradictory and extrapolative goal cells underlying the conservatism summary. Each method entry reports return / KL / adaptation benefit.

Environment	Goal	Type	Prior Only	KL-Reg	PoE
halfcheetah-medium-v2	G_extreme_speed	**extrapolative**	26.5687 / 1.0467 / 0.2435	26.5574 / 0.6857 / 0.2322	**26.5574 / 0.6670 / 0.2322**
halfcheetah-medium-v2	G_reverse	**contradictory**	-5.1736 / 1.0007 / -0.0470	-5.1423 / 2.9969 / -0.0157	**-5.1380 / 0.0339 / -0.0114**
hopper-medium-v2	G_extreme_speed	**extrapolative**	10.2390 / 121.7781 / -0.1657	10.4248 / 5.2950 / 0.0201	**10.4243 / 4.5908 / 0.0196**
hopper-medium-v2	G_reverse	**contradictory**	-2.0691 / 4.5208 / -0.0883	-1.9957 / 2.3958 / -0.0149	**-1.9863 / 0.5422 / -0.0055**
walker2d-medium-v2	G_extreme_speed	**extrapolative**	13.9329 / 87.9328 / 1.6253	14.1093 / 27.4165 / 1.8017	**14.1062 / 26.1528 / 1.7986**
walker2d-medium-v2	G_reverse	**contradictory**	-2.6007 / 9.7198 / -0.2395	-2.4367 / 4.8423 / -0.0755	**-2.4138 / 1.1470 / -0.0526**
hopper-medium-expert-v2	G_extreme_speed	**extrapolative**	10.7973 / 1560.7803 / -0.9782	11.0001 / 22.5256 / -0.7753	**11.1155 / 14.9877 / -0.6600**
hopper-medium-expert-v2	G_reverse	**contradictory**	-2.1957 / 6.2772 / 0.0592	-2.1993 / 4.1289 / 0.0557	**-2.1997 / 4.0655 / 0.0552**

Table 20: Detailed single-environment robustness to deployment-prior degradation on `halfcheetah-medium-v2`. We compare direct prior-only adaptation with the best PoE operating point under degraded-prior settings. Higher return is better; lower catastrophic rate and lower KL from the frozen actor are better.

Goal	Prior Variant	Prior Ret	Prior Cat	Prior KL	Best PoE	PoE Ret	PoE Cat	PoE KL
G1: Speed	Trained prior	5.4784	0.0015	1.9458	$\alpha = 0.1$	5.4727	0.0013	1.1534
	Noisy prior ($\sigma = 0.05$)	5.3051	0.0193	190.6071	$\alpha = 0.1$	5.4605	0.0039	42.4968
	Random prior	3.2040	0.8192	1047.4740	$\alpha = 0.3$	5.3772	0.0104	2.9930
G2: Balanced	Trained prior	3.3655	0.0021	1.9384	$\alpha = 0.1$	3.3626	0.0015	1.1539
	Noisy prior ($\sigma = 0.05$)	3.2125	0.0338	125.0987	$\alpha = 0.1$	3.3504	0.0046	35.2850
	Random prior	1.9986	0.7026	815.4703	$\alpha = 0.3$	3.2759	0.0175	2.7714
G3: Control-weighted	Trained prior	1.0285	0.0077	1.8433	$\alpha = 0.1$	1.0279	0.0082	1.1643
	Noisy prior ($\sigma = 0.05$)	0.9168	0.0462	88.7405	$\alpha = 0.1$	1.0188	0.0111	27.5441
	Random prior	0.4285	0.6113	603.1209	$\alpha = 0.3$	0.9837	0.0210	2.5218

The detailed actor-quality numbers confirm the aggregate story: the stronger anchors remain the best conservative starting points, and PoE stays slightly ahead of KL-Reg in return while keeping KL lower across all three goals.

O.12 Additional Multi-Environment D4RL Comparison Tables

For completeness, we report two compact multi-environment D4RL comparison tables that make the headline frozen-actor baselines explicit across all evaluated environments and deployment goals. These tables complement the main-text D4RL figures and aggregate robustness analyses by showing direct side-by-side comparisons among the frozen actor, prior-only adaptation, additive mixing, KL-regularized post-hoc adaptation, and PoE.

Table 22 provides the full compact numerical comparison. For each setting-goal pair, it reports the goal-conditioned return together with the KL divergence from the frozen actor, making it possible to inspect the return–conservatism tradeoff directly for all principal frozen-actor baselines. This table is useful when the reader wants the exact side-by-side values underlying the broader trends discussed in the main text.

Table 23 provides a higher-level reading aid. Instead of reproducing the full interpretation from the raw numbers, it highlights, for each setting-goal pair, which method achieves the strongest return and which adaptive method stays closest to the frozen actor in KL. Read together, Tables 22 and 23 make clear that the main empirical pattern is not uniform dominance by a single method on every metric. Rather, they expose the central tradeoff emphasized throughout the paper: prior-only adaptation can sometimes achieve the largest raw return, while PoE and KL-regularized adaptation typically deliver a more favorable balance between deployment performance and conservative deviation from the validated actor.

Taken together, Tables 22 and 23 show that the strong halfcheetah pattern is not a single-environment artifact: PoE remains the main conservative adaptive choice in the environments where frozen-actor steering is genuinely useful, while the stronger frozen anchors can still dominate in the harder boundary case.

Table 21: Detailed single-environment actor-quality sensitivity on `halfcheetah-medium-v2`. Weak, medium, and strong actors correspond to behavioral-cloning snapshots at epochs 5, 15, and 50, with validation NLL $= -3.17/-3.91/-4.19$, respectively. The deployment prior is held fixed. Δ denotes return improvement over the corresponding frozen actor.

Actor	Val. NLL	Goal	Frozen Ret	PoE Ret (α)	KL-Reg Ret (β)	PoE Δ	KL-Reg Δ	Cat%$_{\text{PoE}}$
weak	-3.168	G1_speed	5.4263	5.4651 (0.1)	5.4652 (0.1)	+0.0388	+0.0389	0.26
		G2_balanced	3.3322	3.3572 (0.1)	3.3571 (0.1)	+0.0250	+0.0249	0.30
		G3_control	1.0046	1.0223 (0.1)	1.0223 (0.1)	+0.0177	+0.0177	1.13
medium	-3.914	G1_speed	5.4187	5.4629 (0.1)	5.4633 (0.1)	+0.0443	+0.0446	0.24
		G2_balanced	3.3312	3.3562 (0.1)	3.3565 (0.1)	+0.0250	+0.0252	0.28
		G3_control	1.0100	1.0224 (0.1)	1.0225 (0.1)	+0.0124	+0.0125	1.13
strong	-4.194	G1_speed	5.4174	5.4641 (0.1)	5.4638 (0.1)	+0.0466	+0.0464	0.26
		G2_balanced	3.3315	3.3570 (0.1)	3.3569 (0.1)	+0.0255	+0.0254	0.30
		G3_control	1.0119	1.0227 (0.1)	1.0228 (0.1)	+0.0108	+0.0109	1.11

Table 22: Compact multi-environment D4RL headline comparison. Each entry reports goal-conditioned return / KL divergence from the frozen actor. The table makes the closest frozen-actor baselines explicit across all evaluated environments and goals.

Env	Goal	Frozen	Prior Only	Additive	KL-Reg	PoE
`halfcheetah-medium-v2`	G1_speed	5.403 / 0.000	5.453 / 1.047	5.433 / 0.263	5.443 / 0.572	5.444 / 0.390
`halfcheetah-medium-v2`	G2_balanced	3.325 / 0.000	3.352 / 1.043	3.340 / 0.262	3.346 / 0.575	3.347 / 0.391
`halfcheetah-medium-v2`	G3_control	1.011 / 0.000	1.023 / 1.005	1.016 / 0.251	1.021 / 0.618	1.021 / 0.400
`hopper-medium-expert-v2`	G1_speed	2.455 / 0.000	2.260 / 1560.719	2.311 / 392.886	2.282 / 27.656	2.273 / 29.407
`hopper-medium-expert-v2`	G2_balanced	1.678 / 0.000	1.621 / 146.257	1.638 / 37.213	1.649 / 3.336	1.644 / 2.826
`hopper-medium-expert-v2`	G3_control	0.336 / 0.000	0.331 / 6.676	0.335 / 1.676	0.334 / 2.180	0.333 / 1.262
`hopper-medium-v2`	G1_speed	2.181 / 0.000	2.148 / 121.769	2.141 / 31.174	2.184 / 5.276	2.184 / 4.986
`hopper-medium-v2`	G2_balanced	1.541 / 0.000	1.564 / 5.931	1.537 / 1.502	1.553 / 2.081	1.553 / 2.090
`hopper-medium-v2`	G3_control	0.309 / 0.000	0.318 / 4.524	0.310 / 1.123	0.314 / 2.143	0.315 / 2.099
`walker2d-medium-v2`	G1_speed	2.562 / 0.000	2.887 / 87.928	2.826 / 21.892	2.900 / 14.795	2.904 / 15.234
`walker2d-medium-v2`	G2_balanced	1.732 / 0.000	1.863 / 10.604	1.784 / 2.655	1.836 / 5.774	1.838 / 5.720
`walker2d-medium-v2`	G3_control	0.349 / 0.000	0.374 / 9.688	0.358 / 2.427	0.368 / 5.367	0.368 / 5.286

O.13 Extended Ablation Studies

We next examine within-family sensitivity to prior construction and refinement strength. Figure 6 summarizes critic-prior temperature sweeps ($\tau \in \{0.1, 1.0, 5.0\}$) and refinement-prior α sweeps ($\alpha \in \{0.3, 0.5, 0.7\}$) using the released appendix CSVs.

Three qualitative patterns emerge. First, critic-prior temperature has only a mild effect on return but a clearer effect on lower-tail behavior, with the sharpest prior giving the best risk profile within the critic-prior family. Second, within the refinement-prior family, changing α produces the expected adaptation–conservatism tradeoff: larger actor weight improves proximity to the pretrained actor and slightly improves lower-tail behavior. Third, the refinement-prior family is more tuning-robust than the critic-prior family.

The same conclusion holds on the original metric scales: the actor-only policy remains the strongest conservative anchor, while the refinement-prior variants remain much closer to that anchor than critic-only or evidence-free ablations.

O.14 Policy Stability under Critic Degradation

This subsection evaluates the failure tolerance of critic-dependent policies under controlled critic corruption. We add Gaussian noise with standard deviation $\sigma \in \{0.01, 0.05, 0.10, 0.25, 0.50, 1.00\}$ to the critic used for action selection. The corrupted critic affects only the selected actions; policy evaluation is still performed using the goal-weighted dataset reward $r_g(s, a) = g \cdot r_c(s, a)$ rather than critic-estimated values.

Table 23: **Diagnostic only / off-policy evaluator scores; not MuJoCo rollout returns.** Compact multi-environment summary from the off-policy evaluator. We highlight the strongest-return method and the lowest-reported-KL adaptive method per setting-goal pair. PoE and KL-Reg share the same optimal deterministic policy under $\alpha = \beta/(1+\beta)$; the apparent difference in the "Lowest adaptive KL" column reflects only the stochastic-covariance scaling between the two rules' parameterizations, not any policy-level difference. The main-body headline (Fig. 2) uses rollout return directly.

Env	Goal	Frozen	Prior Only	Additive	KL-Reg	PoE	Best return	Lowest adaptive KL
halfcheetah-medium-v2	G1_speed	5.403 / 0.000	5.453 / 1.047	5.433 / 0.263	5.443 / 0.572	5.444 / 0.390	Prior Only (5.453)	Additive (0.263)
halfcheetah-medium-v2	G2_balanced	3.325 / 0.000	3.352 / 1.043	3.340 / 0.262	3.346 / 0.575	3.347 / 0.391	Prior Only (3.352)	Additive (0.262)
halfcheetah-medium-v2	G3_control	1.011 / 0.000	1.023 / 1.005	1.016 / 0.251	1.021 / 0.618	1.021 / 0.400	Prior Only (1.023)	Additive (0.251)
hopper-medium-expert-v2	G1_speed	2.455 / 0.000	2.260 / 1560.719	2.311 / 392.886	2.282 / 27.656	2.273 / 29.407	Frozen Actor (2.455)	KL-Reg (27.656)
hopper-medium-expert-v2	G2_balanced	1.678 / 0.000	1.621 / 146.257	1.638 / 37.213	1.649 / 3.336	1.644 / 2.826	Frozen Actor (1.678)	PoE (2.826)
hopper-medium-expert-v2	G3_control	0.336 / 0.000	0.331 / 6.676	0.335 / 1.676	0.334 / 2.180	0.333 / 1.262	Frozen Actor (0.336)	PoE (1.262)
hopper-medium-v2	G1_speed	2.181 / 0.000	2.148 / 121.769	2.141 / 31.174	2.184 / 5.276	2.184 / 4.986	PoE (2.184)	PoE (4.986)
hopper-medium-v2	G2_balanced	1.541 / 0.000	1.564 / 5.931	1.537 / 1.502	1.553 / 2.081	1.553 / 2.090	Prior Only (1.564)	Additive (1.502)
hopper-medium-v2	G3_control	0.309 / 0.000	0.318 / 4.524	0.310 / 1.123	0.314 / 2.143	0.315 / 2.099	Prior Only (0.318)	Additive (1.123)
walker2d-medium-v2	G1_speed	2.562 / 0.000	2.887 / 87.928	2.826 / 21.892	2.900 / 14.795	2.904 / 15.234	PoE (2.904)	KL-Reg (14.795)
walker2d-medium-v2	G2_balanced	1.732 / 0.000	1.863 / 10.604	1.784 / 2.655	1.836 / 5.774	1.838 / 5.720	Prior Only (1.863)	Additive (2.655)
walker2d-medium-v2	G3_control	0.349 / 0.000	0.374 / 9.688	0.358 / 2.427	0.368 / 5.367	0.368 / 5.286	Prior Only (0.374)	Additive (2.427)

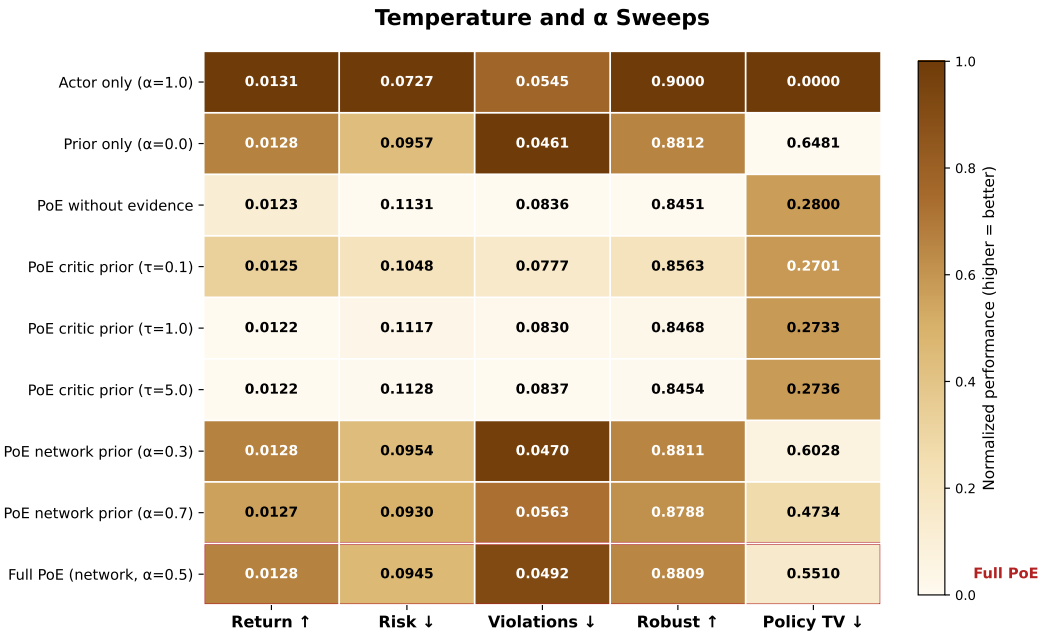

Figure 6: Extended ablation across critic-prior temperature sweeps and refinement-prior α sweeps.

We report two stability metrics as the corruption level increases: the total variation distance from the frozen actor, $\mathrm{TV}(\pi, \pi_\theta)$, and the support-violation rate. Critic-Greedy becomes unstable quickly, with both metrics saturating at low corruption levels. PoE with the refinement prior remains close to the frozen actor's behavioral envelope across the sweep. PoE with the critic prior is less stable because critic noise directly affects the prior used for composition.

O.15 Support Preservation Diagnostic

Proposition 1(iii) establishes exact support preservation: $\pi_\theta(a \mid s) = 0 \Rightarrow \pi_{\mathrm{ref}}(a \mid s, g) = 0$ for all $\alpha > 0$. Because learned actors often assign very small rather than exactly zero probability to many actions, we also report an approximate support-violation rate. The results appear in Table 24.

Table 24 reports the fraction of selected actions satisfying $\pi_\theta(a \mid s) < \epsilon$ for $\epsilon \in \{0.001, 0.01, 0.05\}$. The numbers reported here are computed under the deterministic action-selection convention used throughout this paper (Sec. 6, Appendix B). *Under deterministic mean-action evaluation, PoE does*

Table 24: Approximate support-violation rates at three thresholds ϵ, computed under the deterministic mean-action evaluation convention of Sec. 6 and Appendix B on the AMR diagnostic dataset. The metric is the empirical fraction of selected actions for which $\pi_\theta(a \mid s) < \epsilon$. Under deterministic mean evaluation PoE cannot *theoretically* reduce support violation over the frozen actor; the numerical gap below reflects distributional effects (state-coverage bias and threshold sensitivity) rather than improved support alignment, and the table should be read as a within-protocol consistency check, not as a claim of support-coverage improvement. Critic-Greedy is included as a reference point that violates support almost everywhere.

Method	$\epsilon = 0.001$	$\epsilon = 0.01$	$\epsilon = 0.05$
Pretrained Actor	0.0019	0.0238	0.1184
Critic-Greedy	0.9986	0.9986	0.9986
PoE (Refinement Prior)	**0.0012**	**0.0188**	**0.0885**
PoE (Critic Prior)	0.0476	0.1536	0.3484

Table 25: Q-value statistics of selected actions under the original uncorrupted critic. PoE (Refinement Prior) stays close to dataset and frozen-actor behavior, while Critic-Greedy reflects aggressive critic maximization.

Method	**Mean**	**Std.**	**Min**	**Max**
Dataset actions	0.0127	0.0074	-0.0077	0.0305
Behavior Cloning	0.0150	0.0070	0.0024	0.0461
Pretrained Actor	0.0131	0.0080	-0.0109	0.0461
BCQ	0.0186	0.0036	0.0160	0.0461
Critic-Greedy	**0.0330**	**0.0021**	**0.0297**	**0.0566**
PoE (Refinement Prior)	0.0128	0.0074	-0.0080	0.0297
PoE (Critic Prior)	0.0122	0.0084	-0.0109	0.0458

not theoretically reduce support violation relative to the frozen actor: any difference must reflect distributional effects (dataset coverage bias, the threshold-driven sensitivity of the metric, or shifts in the state distribution induced by the refined policy on dataset states), not improved support alignment per se. With that caveat in mind, PoE with the refinement prior stays below the frozen actor's own violation rate at all three thresholds, while Critic-Greedy violates support almost everywhere. The critic-prior variant lies between these extremes, reflecting its partial dependence on critic-derived scores. We treat this table as a within-protocol consistency check rather than evidence that PoE intrinsically improves support coverage.

O.16 Q-value Analysis of Selected Actions

We also report a diagnostic analysis of the original uncorrupted deployment critic. Table 25 summarizes the mean, standard deviation, minimum, and maximum Q-values of actions selected by each policy under that critic.

Table 25 shows that Critic-Greedy selects a tight high-value action set, consistent with direct critic maximization. By contrast, PoE with the refinement prior selects actions whose Q-value distribution is nearly identical to dataset actions and to the frozen actor, which is exactly the behavior expected from an actor-preserving refinement mechanism. The critic-prior variant exhibits a wider range, again reflecting greater sensitivity to critic-derived structure.

O.17 Additional Benchmark Analysis on Frozen-Actor Goal Adaptation

We provide an additional single-environment benchmark analysis on `halfcheetah-medium-v2` to study deployment-time goal adaptation under a frozen actor. This is a supplement to the main D4RL results in Section 7, where the main-paper comparison focuses on matched-KL tradeoffs and prior degradation.

Table 26 reports return, catastrophic rate, robustness, and KL divergence from the frozen actor under three deployment goals. The table shows that the frozen actor is not uniformly optimal once the

Table 26: Additional benchmark analysis on `halfcheetah-medium-v2` under three deployment goals. Higher return and robustness are better; lower catastrophic rate and KL are better.

	G1: Speed				G2: Balanced				G3: Control-weighted			
Method	**Return ↑**	**Cat. ↓**	**Rob ↑**	**KL ↓**	**Return ↑**	**Cat. ↓**	**Rob ↑**	**KL ↓**	**Return ↑**	**Cat. ↓**	**Rob ↑**	**KL ↓**
Frozen Actor	5.4012	0.0079	0.9921	0.0000	3.3236	0.0094	0.9906	0.0000	1.0107	0.0226	0.9774	0.0000
Prior Only	**5.4784**	0.0015	0.9985	1.9458	**3.3655**	0.0021	0.9979	1.9384	**1.0285**	**0.0077**	**0.9923**	1.8433
PoE ($\alpha = 0.3$)	5.4634	**0.0011**	**0.9989**	0.6666	3.3582	**0.0010**	**0.9990**	0.6691	1.0263	0.0081	0.9919	0.7078
PoE ($\alpha = 0.5$)	5.4541	0.0012	0.9988	0.3970	3.3529	0.0018	0.9982	0.3994	1.0241	0.0089	0.9911	0.4348
PoE ($\alpha = 0.7$)	5.4407	0.0017	0.9983	0.2084	3.3455	0.0020	0.9980	0.2099	1.0207	0.0103	0.9897	0.2321

deployment goal changes, while prior-only adaptation attains the highest return at substantially larger KL divergence. PoE provides a smoother adaptation–conservatism tradeoff across α.

P Implementation Details and Reproducibility

P.1 Training and Evaluation Setup

All models were implemented in PyTorch. The actor is trained once on the offline dataset via behavioral cloning and then frozen; the refinement prior is trained separately on the same dataset via contrastive goal-weighted behavioral cloning. The Product-of-Experts layer introduces no learnable parameters at deployment time, it only reweights actor and prior distributions according to α via Eq. (2). A held-out goal-conditioned critic is trained with a different random seed and used only as a within-protocol diagnostic scorer in Appendix B; the main body's returns are from real MuJoCo rollouts (Sec. 6). Unless otherwise specified, results are averaged over the five evaluation seeds reported in the main body.

P.2 Inference-Time Procedure

Algorithm 1 gives the complete deployment-time refinement procedure for both the finite-action and diagonal-Gaussian settings. The key property is that adaptation is entirely parameter-free at deployment: no gradients are computed, no model weights are updated, and no new data are accessed. The procedure requires only a forward pass through the frozen actor and the offline-trained prior, followed either by a pointwise reweighting over $|\mathcal{A}|$ actions (finite-action case) or by a closed-form Gaussian update (continuous case).

The discrete pointwise reweighting is $\mathcal{O}(|\mathcal{A}|)$ per decision step, which is practical for finite or moderate action spaces; for very large action spaces (e.g. language-model vocabularies) the enumeration cost is the dominant constraint and warrants caching or sparse-support approximations. In the continuous Gaussian setting, no enumeration is required at all: the PoE composition remains diagonal-Gaussian in closed form (Sec. 4.3), so the per-step cost is $\mathcal{O}(d_a)$ in the action dimension. All MuJoCo rollouts in this paper use the Gaussian closed form with deterministic mean actions $a_t = \mu_{\text{ref}}(s_t, g)$.

Changing the deployment objective requires only updating g; the actor, prior, and α are all fixed. This means the same trained components can serve multiple deployment contexts simultaneously, with zero additional training cost per new goal.

P.3 Reproducibility Manifest

The rollout evidence package is fixed and internally validated. For reproducibility, we maintain a manifest that records the code version, execution environment, software dependencies, hardware configuration, command lines used to generate each artifact, and SHA-256 digests for all evidence CSV files.

The main rollout pipeline evaluates five seeds, five episodes per seed, three deployment goals, and four task settings, totaling 3,900 episodes. The prior-degradation pipeline evaluates three seeds, three episodes per seed, three goals, four task settings, and four prior variants, totaling 2,160 episodes. These scripts provide the exact execution path used to regenerate the reported rollout evidence.

Algorithm 1 Deployment-Time Product-of-Experts Policy Refinement

Require: Frozen actor $\pi_\theta(a \mid s)$; prior $\rho_\phi(a \mid s, g)$; coefficient $\alpha \in [0, 1]$; state s; goal g; action mode (STOCHASTIC or DETERMINISTIC)
Ensure: $\pi_{\text{ref}}(\cdot \mid s, g)$ and action $a^\star$

if action space $\mathcal{A}$ is finite or discrete **then**
  for each $a \in \mathcal{A}$ **do**
    $w(a) \leftarrow \pi_\theta(a \mid s)^\alpha \, \rho_\phi(a \mid s, g)^{1-\alpha}$
  end for
  $Z \leftarrow \sum_{a' \in \mathcal{A}} w(a')$; $\pi_{\text{ref}}(a \mid s, g) \leftarrow w(a)/Z$ for each a
  if mode = DETERMINISTIC **then**
    $a^\star \leftarrow \arg\max_a \pi_{\text{ref}}(a \mid s, g)$
  else
    $a^\star \sim \pi_{\text{ref}}(\cdot \mid s, g)$
  end if
else if actor and prior are diagonal Gaussian **then**
  $(\mu_\theta, \Sigma_\theta) \leftarrow \pi_\theta(\cdot \mid s)$; $(\mu_\phi, \Sigma_\phi) \leftarrow \rho_\phi(\cdot \mid s, g)$
  $\Sigma_{\text{ref}}^{-1} \leftarrow \alpha \, \Sigma_\theta^{-1} + (1 - \alpha) \, \Sigma_\phi^{-1}$
  $\mu_{\text{ref}} \leftarrow \Sigma_{\text{ref}} \big(\alpha \, \Sigma_\theta^{-1} \mu_\theta + (1 - \alpha) \, \Sigma_\phi^{-1} \mu_\phi \big)$
  $\pi_{\text{ref}}(\cdot \mid s, g) \leftarrow \mathcal{N}(\mu_{\text{ref}}, \Sigma_{\text{ref}})$
  if mode = DETERMINISTIC **then**
    $a^\star \leftarrow \mu_{\text{ref}}$ ▷ rollout convention used in this paper
  else
    $a^\star \sim \mathcal{N}(\mu_{\text{ref}}, \Sigma_{\text{ref}})$
  end if
end if
return $\pi_{\text{ref}}(\cdot \mid s, g)$, $a^\star$

We also include a package validator that checks setting–goal coverage, missing-value counts, seed consistency, and the matched-pair equivalence described in Sec. 7.1. The validator returns a passing status only when all required consistency checks are satisfied.

Q AI Usage Statement

Generative AI tools were used in a limited and supportive capacity during the preparation of this manuscript. ChatGPT was employed for writing assistance, including refinement of technical language, improvement of clarity and organization, and correction of grammar and spelling. Claude was used to assist with code refactoring, formatting, and software-structure improvements. All scientific content, experimental design, results, interpretations, and conclusions were conceived, implemented, and validated by the authors. No data, results, or analyses were generated or fabricated by AI systems. All AI-assisted outputs were reviewed, edited, and verified by the authors to ensure accuracy, originality, and consistency with scientific standards.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the core contributions, each directly supported by the evidence package. The closed-form equivalence $\text{PoE}(\alpha) \equiv \text{KL-Reg}(\beta = \alpha/(1-\alpha))$ is derived in Sec. 4.3 and verified on dataset states and MuJoCo rollouts in Sec. 7.1 / Appendix C. The empirical characterization claim (help on 4 of 12 cells, frozen on 5, hurt on 3; robustness under degraded priors) is substantiated in Sec. 7.2 and Sec. 7.4, with per-cell numbers in `outputs/rollout/_paper/main_results_table.csv`, `diagnostic_summary.csv`,

and `prior_degradation_table.csv`. The harder-regime claim (medium-expert is HURT in 8 of 9 cells at every $\alpha \in \{0.1, 0.3, 0.5, 0.7\}$ and in 9 of 9 at $\alpha = 0.9$, with isolated HELP only on hopper-ME; medium-replay recovering Frozen ties as α rises) is substantiated by the α sweep in Sec. 7.5 and Appendix J, with per-cell numbers in `outputs/d4rl/harder_alpha_sweep/best_alpha_per_cell.csv` and verdict counts in `outputs/d4rl/harder_alpha_sweep/verdict_by_regime_alpha.csv`. The AntMaze boundary claim (zero success with a BC frozen actor across all methods) is supported by Appendix K and the rollout CSVs under `outputs/d4rl/antmaze_rollout_full/antmaze-umaze-v2/` and `outputs/d4rl/harder_regime_rollout_full/antmaze-medium-play-v2/`. We state explicitly that composition does not universally improve over the frozen actor (including the stronger ceiling on medium-expert) and that Theorem 2 is qualitative at $\gamma = 0.99$, not predictive.

Guidelines:

- The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations.
- The claims made should match theoretical and experimental results.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [Yes]

Justification: An explicit Limitations paragraph closes Sec. 9: (i) the closed form and equivalence assume diagonal-Gaussian actors and priors, the equivalence holds at the deterministic mean, and the formal guarantees are mainly finite-action; (ii) composition hurts in 3 of 12 main cells with five seeds (all speed-dominant goals where the frozen actor is near its ceiling) and in 4 of 12 with ten seeds; (iii) Theorem 2's penalty coefficient $\frac{2\gamma}{(1-\gamma)^2} = 19{,}800$ at $\gamma = 0.99$ makes the right-hand side of Eq. (5) loose, so we treat it as a structural decomposition only; (iv) the ten-seed replication covers the headline package only, the harder-regime package uses 5 seeds $\times$ 5 episodes per cell, the KL-budget selector is evaluated with the off-policy diagnostic evaluator, and the dataset-only sensitivity score was validated retrospectively on 12 cells; (v) evaluation covers D4RL locomotion, AntMaze (including a graded-competence study), and two Adroit tasks, all with BC-trained actors and linear reward-component goal shifts; the AntMaze umaze package used a 2% data subsample. CQL/IQL-guided critic baselines (Sec. 7.3) and an SF/GPI baseline (Appendix N) are competitive in places but do not displace density-anchored composition. We additionally state an *actor-competence ceiling*: on medium-expert composition is HURT in 8 of 9 cells at every $\alpha \in \{0.1, 0.3, 0.5, 0.7\}$ and in 9 of 9 at $\alpha = 0.9$, with isolated HELP only on hopper-ME, and on `antmaze-umaze-v2`/`antmaze-medium-play-v2` with a BC frozen actor every method returns zero (Appendix K); we do not probe $\alpha > 0.9$ and we do not claim the frozen-actor paradigm applies to AntMaze-grade exploration. The HELP/FROZEN/HURT classifier is documented in Appendix G.

Guidelines:

- The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: Theorem 1 (KL-barycenter identity) is stated with its finite-action assumption and proved in Appendix A.1. Theorem 2 (performance-difference decomposition) relies on two standard lemmas (performance-difference and discounted occupancy perturbation) collected in Appendix A.3 and proved in Appendix A.4. Theorem 3 (transition shift) is proved in Appendix A.7, with its supporting kernel-shift lemma in Appendix A.6. Proposition 1 (interpolation and support preservation) and its proof are in Appendix A.2. The continuous-action Wasserstein statement of Proposition 2 and Corollary 4 is proved in Appendix A.10. Corollaries 1, 2, and 3 are in Appendices A.5, A.8, and A.9. We explicitly state that Theorem 2 is not a predictive bound at $\gamma = 0.99$ (Appendix D).

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results?

Answer: [Yes]

Justification: We provide a public code repository containing the implementation, scripts, and instructions needed to reproduce the main experimental results: https://github.com/eliashossain001/poe_composition. The repository includes the full code used to produce the evidence packages, the raw rollout CSVs, per-seed summaries, aggregate tables, and figure-generation pipeline. Reproducibility instructions for the four-setting main-body package are given in Appendix P.3 and in the top-level scripts `scripts/run_main_rollouts.sh`, `scripts/run_prior_degradation.sh`, and `scripts/run_adaptive_alpha_suite.sh`. The CQL/IQL sweep is released under `outputs/d4rl/cql_iql_rollout_paper2/`. The harder-regime + α-sweep package is produced by `scripts/train_and_rollout_env.sh`, `scripts/chain_envs.sh`, and `scripts/alpha_sweep_env.sh`, with outputs under `outputs/d4rl/harder_regime_rollout_full/` and `outputs/d4rl/harder_alpha_sweep/`; aggregation is performed via `scripts/aggregate_rollouts.py` and interpretation via `scripts/analyze_alpha_sweep.py`. The AntMaze diagnostic package is under `outputs/d4rl/antmaze_rollout_full/`. The D4RL datasets (`halfcheetah/hopper/walker2d-medium-v2`, `-medium-replay-v2`, `-medium-expert-v2`, `antmaze-umaze-v2`, `antmaze-medium-play-v2`) are accessed through the standard D4RL release; the dataset manifest is provided at `configs/d4rl_dataset_manifest.json`.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- The instructions should contain the exact command and environment needed to run to reproduce the results.

5. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper?

Answer: [Yes]

Justification: Yes. Appendix P.3 documents the frozen rollout evidence package, and we provide a public repository with the code and instructions needed to reproduce the experiments: https://github.com/eliashossain001/poe_composition. The reproducibility appendix records the git commit hash, hostname, Python and package versions (PyTorch, NumPy, pandas, SciPy, gym, mujoco, `mujoco_py`, D4RL), GPU driver and model, and the SHA-256 digest of every evidence CSV. The

pipeline is wrapped in `scripts/run_main_rollouts.sh` (5 seeds $\times$ 5 episodes $\times$ 3 goals $\times$ 4 environments, 3,900 episodes), `scripts/run_prior_degradation.sh` (3 seeds $\times$ 3 episodes $\times$ 3 goals $\times$ 4 environments $\times$ 4 prior variants, 2,160 episodes), and `scripts/run_adaptive_alpha_suite.sh` for the practical alpha-selection study. The CQL/IQL sweep is released under `outputs/d4rl/cql_iql_rollout_paper2/` together with a paper table builder at `experiments/d4rl/build_cql_iql_tables.py`. The harder-regime and α-sweep packages are produced by `scripts/train_and_rollout_env.sh`, `scripts/chain_envs.sh`, and `scripts/alpha_sweep_env.sh`; outputs live under `outputs/d4rl/harder_regime_rollout_full/`, `outputs/d4rl/harder_alpha_sweep/`, and `outputs/d4rl/full_harder_benchmark/`, with aggregation via `scripts/aggregate_rollouts.py` and analysis via `scripts/analyze_alpha_sweep.py`; a per-environment runbook is available at `scripts/RUNBOOK_harder_benchmark.md`. A validator at `experiments/d4rl/validate_rollout_package.py` checks cell coverage, NaN counts, seed consistency, and matched-pair equivalence on the main package and returns `STATUS: PASS`.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- Reproducibility can be accomplished in various ways. Releasing code and data is one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details necessary to understand the results?

   Answer: [Yes]

   Justification: Sec. 6 specifies the main-body environments, the actor and prior architectures (diagonal-Gaussian, behavioral cloning for the actor, contrastive goal-weighted BC for the prior), the deployment-goal definitions $G_1 = (1, 0.1, 0.1)$, $G_2 = (0.5, 0.5, 0.5)$, $G_3 = (0.1, 1, 0.1)$, the evaluation protocol (real MuJoCo rollouts, 5 seeds, 5 episodes per seed), the reward-component decomposer shared between prior training and rollout evaluation, and the method set with matched α and $\beta = \alpha/(1-\alpha)$. The KL-Reg β grid is $\{0.111, 0.429, 1.000, 2.333, 9.000\}$ and the PoE α grid is $\{0.1, 0.3, 0.5, 0.7, 0.9\}$. Sec. 7.5 and Appendix I specify the additional harder-regime envs (`halfcheetah/hopper/walker2d-medium-replay-v2` and `-medium-expert-v2`) and AntMaze diagnostic envs (`antmaze-umaze-v2`, `antmaze-medium-play-v2`); they reuse the same actor / prior / critic training recipes, except that the umaze actor and prior were trained on a 2% data subsample and its critics for 5 epochs (Appendix K), with AntMaze using the `{G1_success, G2_balanced, G3_efficient}` goal scheme described in the AntMaze reward decomposer. Appendices L and K specify the Adroit and graded-competence AntMaze protocols. Full per-cell hyperparameters and training scripts are in Appendix P.1.

   Guidelines:

   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Headline results in Sec. 7.2 and Figure 2 carry 95% bootstrap confidence intervals computed over 25 episodes per (method, environment, goal) cell, resampled from per-seed summaries. The HELP/FROZEN/HURT classifier in Appendix G uses one confidence-interval half-width as its threshold. The equivalence claim in Sec. 7.1 is supported by seed-matched paired comparisons at each matched α with 95% bootstrap CIs on the

paired difference, and by rliable-style probability of improvement [Agarwal et al., 2021]. All intervals are empirical resampled intervals; we do not assume Gaussian error distributions.

Guidelines:

- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests.
- The method for calculating the error bars should be explained.

8. **Experiments compute resources**

   Question: Does the paper provide sufficient information on the computer resources needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix P.3 records the observed wall times on the hardware we used. The full main rollout pipeline (four task settings, five seeds, five episodes, three goals, thirteen methods) completed in approximately 35 minutes on a single NVIDIA TITAN RTX. The prior-degradation pipeline (four task settings, three seeds, three episodes, three goals, five methods, four prior variants) completed in approximately 68 minutes on the same hardware. The harder-regime + α-sweep package completed in approximately 2.5 hours on two NVIDIA TITAN RTX GPUs run in parallel, covering six locomotion envs and two AntMaze envs; the α sweep alone is 5,850 episodes over six envs. GPU memory use peaked under 4 GB per worker. All training (behavioral cloning for the actor, contrastive goal-weighted BC for the prior, FQE / CQL / IQL / SF critics) uses the same single-GPU setup.

   Guidelines:

   - The paper should indicate the type of compute workers and memory used.
   - The paper should provide the amount of compute required for each experiment.

9. **Code of ethics**

   Question: Does the research conducted in the paper conform with the NeurIPS Code of Ethics?

   Answer: [Yes]

   Justification: The work is performed entirely in the MuJoCo simulator on released D4RL datasets; it involves no human subjects, no scraped or sensitive data, and no deployment to real-world systems. The paper explicitly cautions in its discussion that post-training steering is not a substitute for re-validation when the updated objective is safety-relevant.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The paper's motivation explicitly targets high-stakes deployment settings where retraining is blocked by governance, certification, revalidation, or cost constraints. The positive impact is a lightweight, interpretable actor-anchored composition layer that makes post-training steering auditable. The paper also discusses the negative failure mode: when the deployment prior is unreliable, naive composition (prior-only or additive) can silently degrade performance, and even precision-weighted composition can hurt on speed-dominant goals where the frozen actor is already near its behavioral ceiling. We emphasise that the method is a deployment-time steering mechanism and not a substitute for revalidating a policy whose deployment objective has changed in a safety-relevant way.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse?

Answer: [N/A]

Justification: The paper releases code and simulator-rollout result files for D4RL locomotion tasks. The artifacts pose no dual-use risk and carry no personal data. Model checkpoints are standard behavioral-cloning and Gaussian prior networks for public locomotion environments.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets used in the paper properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: D4RL datasets [Fu et al., 2020] and their environments are accessed through the official D4RL release under its stated license (Apache-2.0 for code, MIT for data). Underlying MuJoCo physics and `mujoco_py` follow their respective licenses. PyTorch, `gym`, NumPy, SciPy, pandas, matplotlib, and seaborn are used under their respective open-source licenses. All cited methods are attributed in Sec. 2.

Guidelines:

- The authors should cite the original paper that produced the code package or dataset.
- The name of the license should be included for each asset.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: We release the code for actor training, prior training, PoE and KL-regularized adaptation baselines, the matched-pair method-set builder, the rollout evaluator, the equivalence audit, and the CPI-bound diagnostic. We also release the camera-ready scripts for the Adroit, graded-competence AntMaze, AWR-trained-prior, and dataset-only sensitivity analyses (`scripts/cr_env_pipeline.sh`, `experiments/d4rl/graded_antmaze.py`, `training/train_awr_prior_d4rl.py`, `rebutta/analysis/`) with their result CSVs under `outputs/camera_ready/`, and the frozen rollout evidence package (raw per-episode CSVs, per-seed summaries, aggregate summaries, paper-ready tables, equivalence audit, CPI diagnostic, reproducibility manifest with SHA-256 digests) under `outputs/rollout/`. Appendix P.3 describes the layout and the scripts that produce each artifact.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing or research with human subjects. All experiments are performed in the MuJoCo simulator.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals were obtained?

Answer: [N/A]

Justification: The paper does not involve research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research?

Answer: [N/A]

Justification: LLMs are not part of the core methodology. As noted in the existing AI usage statement in Appendix Q, generative AI tools were used in a limited and supportive capacity during manuscript preparation and for code refactoring and formatting; all scientific content, experimental design, results, interpretations, and conclusions were conceived, implemented, and verified by the authors.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.