PACE: PATH-AWARE CONSTRAINTS ON EXPLORATION IN FROZEN FLOWS FOR OFFLINE-TO-ONLINE REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Flow policies capture complex, multimodal offline behaviors, yet direct value optimization through iterative generation can destabilize training. Existing methods instead train a flow teacher to imitate offline data and guide the actor that interacts with the environment. During the online phase, however, updating this teacher on newly collected experience can undermine its learned behavioral structure and destabilize policy improvement. Preserving a stable behavioral reference while continually improving its guidance remains an open problem. To address this, we introduce Path-Aware Constraints on Exploration (PACE) in frozen flows for offline-to-online RL, which freezes the pretrained flow and uses an evolving critic to guide exploration in its latent noise space. The explored and original outputs jointly supervise the actor, enabling a frozen teacher to offer both stable behavioral guidance and continually updated knowledge. Using the flow's fixed geometry, PACE further constrains deformation along complete noise-to-action paths to reduce out-of-distribution risk. Experiments on 35 OGBench tasks demonstrate improved sample efficiency and final performance under limited online interaction.

1 INTRODUCTION

Environment interaction is expensive in many reinforcement learning applications. Offline-to-online reinforcement learning (O2O RL) addresses this setting by initializing from an offline dataset and improving through a limited online budget (Lee et al., 2022; Ball et al., 2023; Nakamoto et al., 2023). As tasks become more complex, these datasets contain diverse, multimodal behaviors that conventional Gaussian policies can struggle to represent. Expressive policies based on diffusion and flow models offer richer behavioral priors for this setting (Wang et al., 2023; Hansen-Estruch et al., 2023; Park et al., 2025b).

Stable value optimization remains challenging for such policies because backpropagating through iterative generation can destabilize training (Kang et al., 2023; Park et al., 2025b). FQL addresses this difficulty by training a flow teacher through behavior cloning and distilling its outputs into a one-step actor, which learns from the critic without updating the teacher through the value objective (Park et al., 2025b). This separation provides an effective foundation for expressive-policy learning and online adaptation (Shin et al., 2026).

During online fine-tuning, methods such as FQL and FINO continue fitting the flow teacher to replay data that include the learned policy's new experience (Park et al., 2025b; Shin et al., 2026). Although these updates may incorporate useful new behaviors, their benefit is uncertain under limited feedback, and repeatedly fitting a replay distribution that shifts with the actor can instead distort the teacher's learned behavioral structure and weaken its supervision. Our flow diagnostics illustrate this change through more diffuse peripheral structure around the pretrained modes, with probability mass spreading into a broader set of low-density regions (figure 1). Our controlled comparisons further show that FQL with a pretrained, frozen teacher achieves higher average final performance than standard FQL under a limited online budget (section 3.1). These observations bring a stability–plasticity challenge into focus. How can we preserve the teacher's behavioral reference while continually improving the guidance it provides, without optimizing its parameters through the critic?

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

We introduce *Path-Aware Constraints on Exploration (PACE)*, which freezes a pretrained flow and uses Q-guided exploration in its latent noise space to construct adaptive actor targets. Freezing preserves the learned behavioral structure, while the evolving critic identifies promising changes to the latent inputs without updating the flow parameters. The resulting actions are distilled into the one-step actor alongside the teacher's original outputs. As the critic learns from new experience, these explored targets change, allowing the same frozen teacher to provide both stable behavioral guidance and continually updated knowledge. Reliable exploration, however, requires considering the flow's directional sensitivity. Perturbations of similar magnitude can affect the generation process differently, and constraints on latent inputs and final actions alone do not capture changes along the full trajectory. PACE therefore measures deformation along complete noise-to-action paths through the frozen flow and uses this geometric information to keep critic-guided exploration within better-supported action regions. In this way, the offline flow serves as both a stable behavioral reference and a geometric basis for online policy improvement.

To our knowledge, PACE is the first approach in expressive-policy O2O RL to introduce full-path-constrained latent exploration into a teacher–student framework. Our contributions are threefold.

- We combine a fixed behavioral reference with critic-guided latent exploration, enabling a frozen flow teacher to provide adaptive distillation targets without updating its parameters.
- We derive a path-aware constraint that accounts for the teacher's directional geometry throughout generation, making value-guided exploration more reliable.
- Experiments on 35 OGBench tasks demonstrate improved learning efficiency and final performance within 500k online interactions, supported by component ablations.

2 Related Work

The main challenge in offline-to-online reinforcement learning (O2O RL) is to combine offline knowledge with online feedback for stable policy improvement. Prior work with Gaussian policies has addressed this challenge through value calibration (Nakamoto et al., 2023; Yu & Zhang, 2023) and replay-buffer design (Lee et al., 2022; Ball et al., 2023). For expressive policies such as multi-step diffusion and flow, however, direct value optimization through the generator entails back-propagation through time (BPTT), leading to costly and unstable optimization (Park et al., 2025b; Zhang et al., 2025). This motivates alternative interfaces for incorporating value information.

Combining generative priors with value information. Existing approaches have explored different interfaces for combining offline knowledge with online value information, including updating generator parameters (Zhang et al., 2025), modifying latent inputs (Wagenmaker et al., 2025), and resampling generated actions according to their values (Hansen-Estruch et al., 2023). A promising direction separates a generative reference from a lightweight policy-improvement module, combining rich behavioral guidance with tractable value optimization (Dong et al., 2026; Shin et al., 2026; Li & Levine, 2026). For example, SRPO regularizes an independent offline actor using a pretrained diffusion score (Chen et al., 2024), while FQL distills a flow behavior model into a value-optimized one-step actor (Park et al., 2025b). Our work follows this direction. These methods continue fitting the generative reference to replay data through imitation learning during online adaptation. However, under a limited online budget, our work demonstrates that freezing the offline flow can offer a more favorable stability-plasticity tradeoff, avoiding potential instability from flow drift while retaining adaptation in the lightweight policy-improvement module.

Constraints on value-guided improvement. Beyond choosing an interface, stable improvement requires deciding which candidate action modifications to permit. In value-guided learning, inaccurate values of out-of-distribution (OOD) actions can misdirect policy updates and destabilize training (Yu & Zhang, 2023). Existing approaches restrict latent inputs or penalize latent perturbations (Simsir & Oguz, 2026; Im et al., 2026), bound action residuals or penalize output discrepancies (Dong et al., 2026; Park et al., 2025b; Im et al., 2026), or preserve conditioning sensitivity during guidance (Ren et al., 2026). However, these constraints do not directly measure the deformation induced along the noise-to-action path, which provide a complementary geometric signal for limiting OOD deviations. PACE evaluates candidate action modifications through a path-dependent cost to constrain value-guided search and thereby construct trustworthy improvement targets.

3 METHOD

In this section, we develop PACE within a teacher–student actor–critic framework, where a conditional flow models the behavior distribution, a one-step actor produces actions, and a critic provides the value objective for policy improvement. We focus on how to preserve a stable behavioral reference while adapting actor supervision (section 3.1), and how to constrain critic-guided exploration using the flow's generation paths (section 3.2). The technical backgrounds are described below.

Offline-to-online reinforcement learning. We formulate reinforcement learning as a Markov decision process with states $s \in \mathcal{S}$, continuous actions $a \in \mathbb{R}^d$, reward r, and discount γ, with the objective of maximizing expected discounted return. Offline-to-online RL first learns a policy from a fixed dataset $\mathcal{D}_{\text{off}}$ of transitions (s, a, r, s'). The policy is then fine-tuned through online interaction under a fixed budget B.

Flow-based action generation. For any fixed state s, a conditional flow transforms a Gaussian latent input $z \sim \mathcal{N}(0, I)$ into an action through a learned velocity field v_θ. Starting from $\Phi_0(s, z) = z$, it follows $\mathrm{d}\Phi_t(s, z)/\mathrm{d}t = v_\theta(s, \Phi_t(s, z), t)$, whose integration to $t = 1$ yields the action $\Phi_1(s, z)$. Intuitively, the velocity field transports a standard Gaussian distribution into a state-conditioned action distribution learned from behavioral data (Lipman et al., 2023).

Value-guided policy improvement. A common strategy separates a generative behavior model from a lightweight policy-improvement module (Chen et al., 2024; Park et al., 2025b; Dong et al., 2026). The critic drives value maximization, while behavioral regularization or constraints on action modifications control departures from the generative reference. PACE builds on FQL, an instance of this policy improvement where a flow teacher provides reference actions for a one-step actor (Park et al., 2025b). Given the same Gaussian input $\epsilon \sim \mathcal{N}(0, I)$, their outputs are $a_B = \Phi_1(s, \epsilon)$ and $a_\pi = \mu_\omega(s, \epsilon)$. The actor is trained with

$$\mathcal{L}_{\text{FQL}} = \mathbb{E}_{s,\epsilon}\left[-Q^{\text{actor}}(s, a_\pi) + \alpha \ell(a_\pi, a_B)\right], \qquad \ell(a, b) = \frac{\|a - b\|_2^2}{d}. \tag{1}$$

Here states are sampled from replay, Q^{actor} denotes the critic value used for actor updates, and α weights distillation toward a_B. The teacher target a_B is held constant during actor optimization.

3.1 FROZEN TEACHERS WITH ADAPTIVE LATENT EXPLORATION

Frozen teacher improves stability. In FQL's setting, online flow updates fit the replay buffer through behavior cloning, changing the reference used to supervise the actor. However, under limited feedback, repeatedly fitting experience from an evolving policy can weaken this reference before the agent can reliably assess the new behaviors. Our diagnostics reveal that after online fine-tuning, there exists more diffuse peripheral structure around the offline captured action modes (figure 1 left). Moreover, this spread of probability mass into sparser regions yields poorer student performance. Standard FQL fails catastrophically on Cube Double Task 4 and achieves lower average final performance across the seven tested tasks than its frozen-teacher counterpart (figure 1 right). These observations motivate preserving the pretrained teacher while allocating online adaptation to the critic and actor. We therefore pretrain a teacher before offline RL begins, using the same flow-matching procedure as FQL's flow teacher, and keep its parameters fixed throughout the subsequent offline and online RL stages.

Improved action targets with a frozen teacher. Freezing preserves the behavioral reference, yet a fixed generator need not provide fixed guidance. A flow policy takes an explicit latent input z in addition to s. This gives us another way to improve its guidance by searching for better inputs instead of changing its parameters. The first step is to determine where exploration should begin. Specifically, we integrate backward through the frozen flow's velocity field from a_π at $t = 1$ to $t = 0$, obtaining $z_\pi = \Phi_1^{-1}(s, a_\pi)$ as the starting point for exploration, so that we can improve the policy directly from its current behavior a_π. And the path traced by this backward integration is retained as the reference trajectory for the subsequent path constraint.

The critic then guides a perturbation δ of z_π, regulated by the path constraint in section 3.2. One straightforward way to use the critic for exploration is to decode and rank many latent candidates, but this requires a full ODE solve for every proposal, and finite sampling can miss useful local directions.

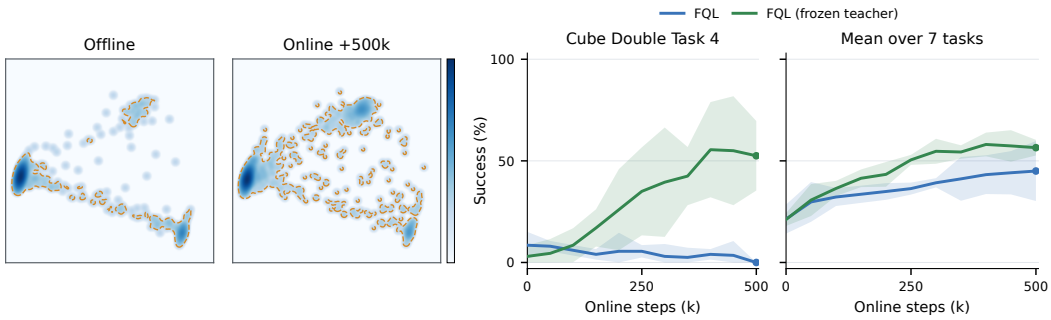

Figure 1: Teacher drift and the benefit of a stable behavioral reference. Distributions of FQL's flow teacher before and after online learning (left), with its projected 99% probability-mass region expanding to 2.16 times its offline area, alongside standard and frozen-teacher FQL learning curves on Cube Double and averaged over seven tasks (right). Appendix C.5 provides the complete seven-task comparison.

PACE instead uses the latent Q-gradient $g = J_1^\top \nabla_a \bar{Q}(s, a_\pi)/q$ to take a single constrained step, where J_1 is the terminal flow Jacobian, $\bar{Q}$ denotes the target-critic aggregation, and q normalizes Q values. Using the path metric $G = G(s, z_\pi)$ defined in section 3.2, we compute

$$\tilde{\delta} = \eta G^{-1} g, \qquad \delta = \min\left(1, \frac{\rho}{\sqrt{\tilde{\delta}^\top G \tilde{\delta}}}\right) \tilde{\delta}. \tag{2}$$

where η controls the proposed step and ρ is the trust radius; a zero proposal gives $\delta = 0$. In the continuous-flow formulation, we decode $z_\pi + \delta$, clip its action difference from a_π, and add the resulting residual to the current action,

$$a_{\text{imp}} = a_\pi + \text{clip}(\Phi_1(s, z_\pi + \delta) - a_\pi)\,. \tag{3}$$

The resulting a_{imp} serves as the explored target for actor supervision. PACE also retains FQL's ordinary distillation target $a_B = \Phi_1(s, \epsilon)$, generated from the same noise input ϵ as the actor, ensuring that the actor continues to receive a stable behavioral reference from the frozen teacher. All targets are treated as constants when optimizing the actor loss, so value-guided exploration changes the supervision without updating the teacher parameters. Together, these objectives give the actor loss

$$\mathcal{L}_\pi = \mathcal{L}_{\text{FQL}} + \alpha\beta\mathbb{E}_{s,\epsilon}\left[\frac{\|a_\pi - a_{\text{imp}}\|_2^2}{d}\right]. \tag{4}$$

We use FQL's recommended task-specific distillation weight α and tune β by task for the explored targets, with d denoting the action dimension. PACE thus preserves FQL's direct value objective and ordinary distillation term in $\mathcal{L}_{\text{FQL}}$ while adding supervision from a_{imp}, offering both stable behavioral guidance and continually updated knowledge.

3.2 Path-aware constraints on exploration

Unrestricted Q maximization can favor poorly supported actions with overestimated values, motivating constraints on latent exploration (Fujimoto et al., 2019; Kumar et al., 2019). For a flow teacher, these constraints should account for how latent perturbations propagate through the generation process. A key observation is that flow models respond differently to perturbations in different directions. The learned velocity field amplifies or attenuates perturbations to different degrees along the generation path. This anisotropic sensitivity makes equally sized latent changes produce different path deformations. Bounding the initial latent displacement ignores this directional response, while bounding the final action displacement measures only its net effect. Even when both are small, substantial deviations may occur at intermediate stages, yielding unexpected perturbation directions that results in OOD actions. As illustrated in figure 2, these boundary constraints can still admit actions in poorly supported regions. If the critic overvalues such an action, using it as an improvement target can misdirect actor updates and destabilize learning. PACE instead measures deformation relative to the reference trajectory throughout generation, making directions that strongly alter the path more expensive.

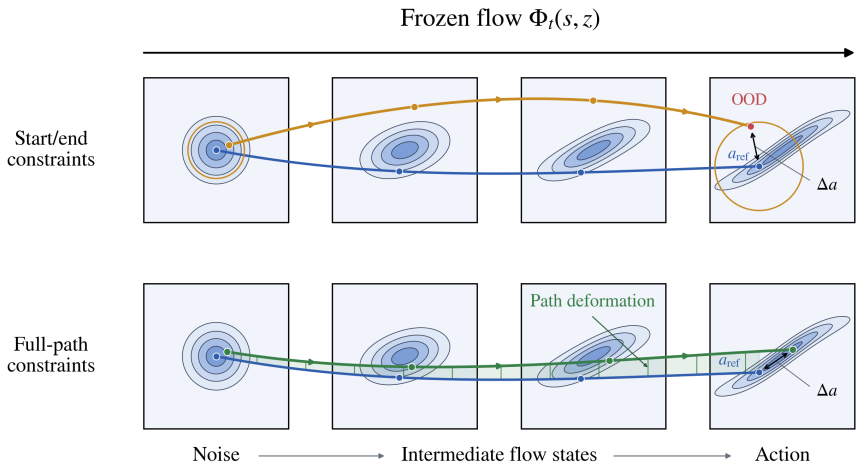

Figure 2: Start/end constraints (top) and full-path constraints (bottom). Blue traces the reference flow; orange illustrates a perturbation ending outside the high-density region. PACE penalizes deformation throughout generation (green), accounting for directional sensitivity along the path.

To formalize the deformation we aim to control, let D_{path} denote the total squared difference between the original and perturbed generation paths. Direct optimization with this nonlinear cost would require repeated evaluations of perturbed flow trajectories. We therefore linearize the flow with respect to its latent input to obtain a quadratic approximation, whose metric G includes scale calibration and damping,

$$D_{\text{path}}(s, z, \delta) = \int_0^1 \left\| \Phi_t(s, z+\delta) - \Phi_t(s, z) \right\|_2^2 \mathrm{d}t,$$
$$G(s, z) = \frac{1}{c} \int_0^1 J_t(s, z)^\top J_t(s, z)\, \mathrm{d}t + \lambda I. \tag{5}$$

Here $J_t = \partial \Phi_t / \partial z$, $c > 0$ calibrates scale, and $\lambda > 0$ provides damping. For small perturbations, $\delta^\top G \delta \simeq D_{\text{path}}/c + \lambda \|\delta\|_2^2$. PACE uses this quadratic approximation in place of directly evaluating D_{path} during optimization, allowing the constrained update in equation 2 to be computed with a single linear solve and radius rescaling. This makes directions with high path sensitivity more costly while seeking critic-predicted improvement. And thanks to the frozen flow parameters, we can compute this geometry using a consistent reference throughout both offline and online RL. Appendix A provides the derivation.

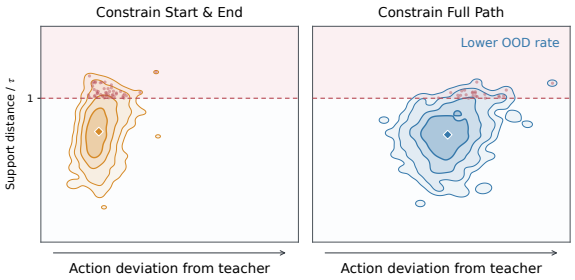

Figure 3: Actor distributions under start/end and full-path constraints at a representative state. At online +500k, across 256 fixed Cube Double states and 1,024 common samples per state, the full-path actor reduces empirical OOD from 3.46% to 3.12% relative to start/end constraints, while the mean action deviation increases from 0.0286 to 0.0382. See Appendix C.4 for the diagnostic protocol.

We design a simple diagnostic experiment to examine how this geometric control affects exploration. Compared with start/end constraints, full-path constraints reduce the empirical OOD rate, and this reduction is not achieved simply by conservatively pulling the actor back toward the teacher, as the mean action deviation actually increases (figure 3). The subsequent performance comparison also shows higher average success across the seven task families with full-path constraints (Table 2). These results suggest that path deformation helps steer critic-guided exploration toward higher predicted value with lower OOD risk while preserving room for actor improvement.

3.3 PRACTICAL COMPUTATION AND EXECUTION

PACE adopts FQL's architecture, including its flow teacher and one-step actor. We use ten Euler steps for both forward and backward ODE integration, matching the step count of FQL's flow teacher. Backward integration locates the latent z_π corresponding to the actor's current action, providing a starting point for exploration that directly seeks to improve that action. The explored latent is decoded to construct the explored target, which is computed for 32 examples per batch of 256 to limit training cost. We set $\rho = 0.05$ in equation 2 to bound the latent perturbation under the path metric, and $\lambda = 0.01$ in equation 5 to stabilize the linear solve through the damping term λI. Further numerical details appear in Appendix B.

The teacher receives 61k offline flow-matching updates with a batch size of 4,096 and is then frozen. Actor–critic training uses 1M offline updates and 500k online interactions. Because critic estimates can be unreliable early in RL training, we initially omit the exploration-target loss and then gradually increase its weight. During the online phase, we follow FINO's entropy-guided sampling to select among candidate actions according to their Q values (Shin et al., 2026). Based on the action dimensionality of these tasks, we use $K = 2$ candidates for AntMaze and AntSoccer, $K = 10$ for HumanoidMaze, and $K = 1$ for Cube Double and Puzzle 4×4. These candidate counts are all less than or equal to those in FINO's official settings.

Table 1: Offline $\rightarrow$ online success (%) after 500k interactions across seven OGBench families, each averaging five tasks. Our FINO and PACE results average eight seeds; † marks published ten-seed results from Shin et al. (2026). The $\pm$ terms give 95% CI half-widths. Final scores within 2 percentage points of the best mean within each family are bolded. Due to reproduction differences, FINO parentheses show published online final scores for HumanoidMaze Large and AntSoccer. Appendix D reports all 35 tasks.

Family	ReBRAC†	Cal-QL†	RLPD†	IFQL†	FQL†	FINO	PACE
HumanoidMaze Medium	21 ±5 → 3 ±3	0 ±0 → 0 ±0	0 ±0 → 1 ±1	59 ±7 → 70 ±5	53 ±6 → 61 ±2	76 ±7 → 90 ±9	55 ±11 → **97** ±2
HumanoidMaze Large	2 ±1 → 1 ±0	0 ±0 → 0 ±0	0 ±0 → 0 ±0	11 ±3 → 10 ±2	5 ±1 → 10 ±3	4 ±3 → 22 ±4 (**33** ±7)	2 ±2 → 28 ±8
AntMaze Large	85 ±3 → **99** ±0	12 ±9 → 12 ±8	0 ±0 → 80 ±7	32 ±4 → 72 ±6	81 ±2 → 92 ±1	78 ±8 → **98** ±1	78 ±5 → **98** ±1
AntMaze Giant	35 ±7 → **96** ±4	2 ±3 → 0 ±0	0 ±0 → 47 ±9	1 ±1 → 0 ±0	16 ±6 → 71 ±5	6 ±5 → 72 ±9	13 ±5 → 67 ±6
AntSoccer	0 ±0 → 0 ±0	0 ±0 → 0 ±0	0 ±0 → 2 ±1	33 ±4 → 35 ±5	61 ±2 → 74 ±4	49 ±6 → 48 ±5 (77 ±5)	50 ±6 → **91** ±2
Cube Double	9 ±2 → 28 ±2	2 ±3 → 0 ±0	0 ±0 → 2 ±3	14 ±1 → 40 ±2	31 ±3 → 73 ±2	47 ±4 → 75 ±5	32 ±3 → **95** ±1
Puzzle 4×4	14 ±1 → 29 ±5	2 ±3 → 20 ±8	0 ±0 → 58 ±11	26 ±2 → 42 ±4	15 ±2 → 45 ±8	30 ±2 → 51 ±8	30 ±3 → **99** ±0
Mean	24 → 37	3 → 5	0 → 27	25 → 38	37 → 61	41 → 65	37 → **82**

4 EXPERIMENTS

4.1 SETUP AND BASELINES

Environment. We evaluate PACE on OGBench, a recently introduced benchmark featuring more diverse and complex tasks than the widely used D4RL suite (Park et al., 2025a; Shin et al., 2026). We adopt the same seven OGBench task families as FINO (Shin et al., 2026), covering 35 tasks with five tasks per family. All experiments use the state-based single-task setting, where each task specifies a fixed goal.

Baselines. We compare against ReBRAC, Cal-QL, RLPD, IFQL, and FQL using the ten-seed results reported by Shin et al. (2026). ReBRAC is a behavior-regularized actor–critic method that combines policy improvement with penalties for departing from dataset actions (Tarasov et al., 2023). Cal-QL calibrates conservative value estimates during offline pretraining to support efficient online fine-tuning (Nakamoto et al., 2023). RLPD combines offline and newly collected data in an off-policy learner to improve online sample efficiency (Ball et al., 2023). IFQL combines implicit Q-learning with a flow behavior model and selects high-value actions from generated candidates (Park et al., 2025b). FQL, introduced in section 3, serves as a direct backbone baseline for PACE (Park et al., 2025b). We also reproduce FINO using its official implementation and task settings (Shin et al., 2026). Our final scores differ substantially from the published results on HumanoidMaze Large and

AntSoccer; we therefore include the published values in parentheses for these families and their tasks.

Evaluation and hyperparameters. PACE first pretrains its flow teacher on the offline dataset for 61k flow-matching updates with a batch size of 4,096. Each run then uses 1M offline actor–critic updates followed by 500k online interactions. FINO and PACE each use eight training seeds per task, with 50 evaluation episodes every 50k online interactions. Final success is measured at online +500k. Normalized AUC measures average performance throughout online learning, using evaluations every 50k interactions. Families average five tasks equally, and the overall score averages all 35 tasks. Both main comparisons and four-seed ablations report means with 95% CIs. Tables display the CI half-width with an explicit ± sign. For offline-to-online comparisons, we bold online scores within 2 percentage points of the best mean within each family. Details appear in Appendix C. All hyperparameters shared with FINO use the same settings. The borrowed candidate-selection mechanism likewise sets candidate counts according to task action dimensionality, with counts no larger than FINO's (section 3.3), so the comparison does not give PACE an unfair advantage through candidate sampling.

4.2 Final performance and learning efficiency

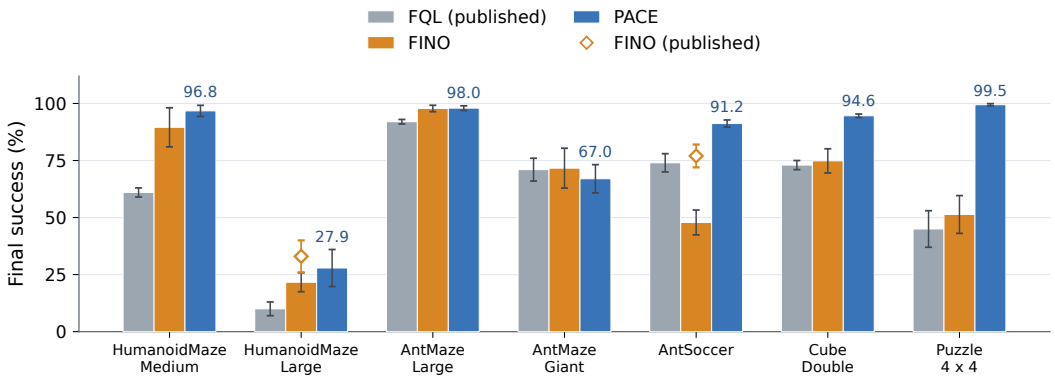

Figure 4: Final success after 500k interactions with 95% CIs. FQL uses published results; FINO and PACE use our experiments. Diamonds show published FINO scores on HumanoidMaze Large and AntSoccer.

Table 1 and figure 4 summarize final performance across the seven task families. PACE achieves strong performance across diverse navigation, control, and manipulation environments, increasing overall success from FINO's 65.0% to 82.1% and finishing higher in six of seven families. The comparisons reveal substantial variation in baseline performance across environments. ReBRAC performs strongly on AntMaze, but this strength does not extend to HumanoidMaze or the manipulation tasks, where flow-based methods are more competitive. Even among flow-based methods, the gap between IFQL, FQL, and PACE shows that expressive behavior modeling alone does not ensure effective online adaptation. PACE's largest advantages over FINO occur on AntSoccer, Cube Double, and Puzzle, whereas several methods already approach the performance ceiling on AntMaze Large. On AntSoccer, PACE also exceeds FINO's published result, showing that the advantage on this family extends beyond the gap in our reproduction. These results demonstrate that retaining a fixed flow is compatible with substantial online improvement when its guidance is adapted through the critic. The component analysis in section 4.3 examines how freezing, latent exploration, and path constraints contribute to this behavior.

The learning curves in figure 5 further show PACE's solid performance and adaptivity across varied environments over the course of online training. Normalized AUC increases from FINO's 59 to 72 (table 7), with the overall mean remaining higher at every evaluated checkpoint from 50k onward. By 150k interactions, PACE reaches 72.0% average success, already exceeding FINO's 65.0% at the end of the 500k budget. The family-level trajectories clarify where this efficiency comes from. On AntSoccer, FINO remains near its initial success level while PACE makes sustained progress throughout training. On Puzzle, the methods begin at comparable success rates but separate in-

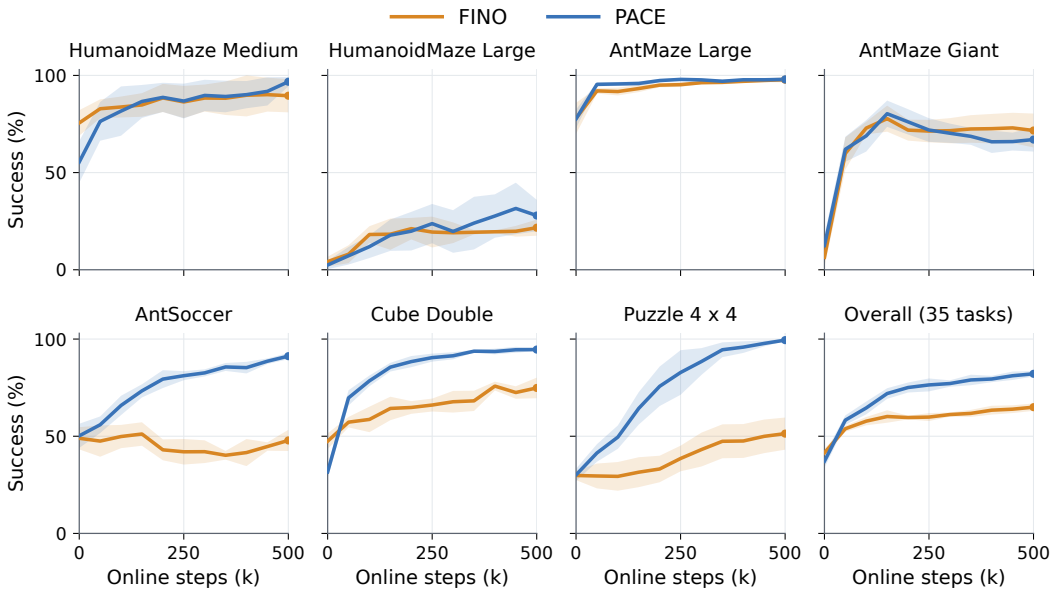

Figure 5: Online learning across seven families and the overall average across all 35 tasks. Curves show eight-seed means; shading gives 95% CIs across seed-level task averages.

creasingly as online experience accumulates; on Cube Double, PACE establishes a clear advantage within the first 50k interactions. HumanoidMaze exhibits a different pattern, with the final advantage emerging later in training. Thus the aggregate AUC gain reflects both rapid improvement and sustained adaptation across multiple task families. Throughout these changes, the teacher remains frozen, illustrating how an evolving critic can extract increasingly useful guidance from a fixed generative prior.

4.3 Component ablations and analysis

In component ablations, we evaluate five variants on Task 4 of each family with seeds 0–3, 1M offline actor–critic updates, and 500k online interactions. Full PACE uses the corresponding task and seed results from the main benchmark. The variants retain PACE's candidate mechanism and examine teacher freezing, the exploration target, and the geometry of exploration.

- *Continually updated flow.* Continue FQL-style flow-matching updates throughout offline and online training from the standard FQL initialization; retain the exploration target and the full-path metric.
- *w/o exploration target.* Remove the exploration-target loss while retaining ordinary distillation, the actor's direct Q objective, and the frozen flow teacher.
- *Start-point constraint.* Use $G_{\mathrm{S}} = I$, constraining the latent perturbation at the start of generation.
- *Endpoint constraint.* Use $G_{\mathrm{E}} = J_1^\top J_1/c + \lambda I$, retaining only the terminal action sensitivity and numerical damping.
- *Start & end constraints.* Replace the complete-path metric by $G_{\mathrm{SE}} = I + J_1^\top J_1/c + \lambda I$, retaining the initial latent and endpoint sensitivities with the same calibration and damping.

Table 2 shows that the three components address complementary challenges in online adaptation. The continually updated flow learns effectively on AntMaze, but encounters non-finite training on Cube Double and Puzzle, indicating that a changing teacher does not provide an equally reliable foundation across tasks. Updating the teacher changes both the actor's distillation reference and the geometry used to guide exploration. When the teacher follows experience from the evolving policy, these changes may weaken the behavioral structure that originally made the offline prior

Table 2: Component ablations on Task 4 of seven families. Entries give offline → online success (%) with 95% CI half-widths across four seeds shown after ±. Mean gives task averages. $0^{\dagger}$ marks failed runs assigned zero at both offline and online checkpoints.

Task 4	Continually updated flow	w/o exploration target	Start-point constraint	Endpoint constraint	Start & end constraints	PACE
HumanoidMaze Medium	$0_{\pm 0} \rightarrow 73_{\pm 77}$	$35_{\pm 48} \rightarrow 64_{\pm 68}$	$16_{\pm 43} \rightarrow \mathbf{96}_{\pm 6}$	$25_{\pm 55} \rightarrow 70_{\pm 74}$	$31_{\pm 60} \rightarrow 49_{\pm 90}$	$5_{\pm 8} \rightarrow \mathbf{95}_{\pm 8}$
HumanoidMaze Large	$0_{\pm 0} \rightarrow 21_{\pm 65}$	$0_{\pm 0} \rightarrow 22_{\pm 68}$	$0_{\pm 0} \rightarrow 28_{\pm 75}$	$0_{\pm 0} \rightarrow 10_{\pm 32}$	$0_{\pm 0} \rightarrow 17_{\pm 53}$	$0_{\pm 0} \rightarrow \mathbf{37}_{\pm 68}$
AntMaze Large	$18_{\pm 33} \rightarrow \mathbf{100}_{\pm 0}$	$76_{\pm 17} \rightarrow 94_{\pm 5}$	$46_{\pm 60} \rightarrow 98_{\pm 4}$	$73_{\pm 8} \rightarrow 97_{\pm 3}$	$72_{\pm 10} \rightarrow 93_{\pm 2}$	$75_{\pm 10} \rightarrow \mathbf{98}_{\pm 3}$
AntMaze Giant	$0_{\pm 0} \rightarrow \mathbf{99}_{\pm 3}$	$0_{\pm 0} \rightarrow 91_{\pm 18}$	$18_{\pm 56} \rightarrow 83_{\pm 36}$	$17_{\pm 54} \rightarrow 86_{\pm 20}$	$18_{\pm 57} \rightarrow 93_{\pm 8}$	$0_{\pm 0} \rightarrow 96_{\pm 5}$
AntSoccer	$38_{\pm 24} \rightarrow 63_{\pm 28}$	$54_{\pm 13} \rightarrow \mathbf{91}_{\pm 9}$	$47_{\pm 19} \rightarrow 88_{\pm 7}$	$49_{\pm 8} \rightarrow \mathbf{91}_{\pm 5}$	$51_{\pm 18} \rightarrow \mathbf{91}_{\pm 3}$	$41_{\pm 14} \rightarrow \mathbf{90}_{\pm 7}$
Cube Double	$0^{\dagger} \rightarrow 0^{\dagger}$	$4_{\pm 5} \rightarrow 58_{\pm 21}$	$1_{\pm 2} \rightarrow \mathbf{75}_{\pm 18}$	$4_{\pm 6} \rightarrow 64_{\pm 9}$	$3_{\pm 4} \rightarrow 61_{\pm 13}$	$3_{\pm 6} \rightarrow \mathbf{77}_{\pm 5}$
Puzzle 4×4	$0^{\dagger} \rightarrow 0^{\dagger}$	$13_{\pm 13} \rightarrow 88_{\pm 7}$	$17_{\pm 12} \rightarrow 85_{\pm 11}$	$14_{\pm 4} \rightarrow 91_{\pm 4}$	$12_{\pm 3} \rightarrow 88_{\pm 8}$	$21_{\pm 11} \rightarrow \mathbf{100}_{\pm 2}$
Mean	$8 \rightarrow 51$	$26 \rightarrow 72$	$20 \rightarrow 79$	$26 \rightarrow 72$	$27 \rightarrow 70$	$21 \rightarrow \mathbf{84}$

useful. The diagnostic in figure 1 illustrates this concern through expansion of the distribution's sparse periphery, while the accompanying FQL controls show the benefit of retaining a pretrained, frozen teacher. Together, these observations support separating adaptation from the reference that anchors it.

The exploration-target ablation then examines how to improve the actor while preserving this reference. Without the explored target, the actor still receives direct Q gradients and ordinary teacher distillation, yet its mean final success drops by 12.1 percentage points. This gap is particularly pronounced on HumanoidMaze Medium and Cube Double, suggesting that the additional guidance contributes beyond the actor's existing value objective. Ordinary distillation transmits the teacher's fixed outputs; latent exploration combines the current critic with the frozen flow teacher to search for improved actions. Distilling those actions allows newly acquired value information to change the teacher's guidance while preserving its parameters. The result supports our central design choice of obtaining adaptive supervision from a stable generative prior.

Finally, the geometry ablations show that restricting exploration at its boundaries does not recover the benefit of considering the full path. Start-point constraints ignore the flow's directional sensitivity, while endpoint constraints omit how a perturbation propagates through intermediate flow states. Combining the two still trails full PACE on average, and no restricted alternative matches its performance on HumanoidMaze Large and Puzzle. This pattern is consistent with the motivation in figure 2, where small boundary displacements can conceal substantial path deformation. The empirical diagnostic in figure 3 further shows that full-path constraints permit greater action deviation while lowering OOD incidence. Thus their benefit is consistent with more selective control over exploration directions, allowing meaningful policy changes while reducing excursions into poorly supported regions. Freezing, latent exploration, and path-aware control together provide the stable reference, adaptive guidance, and geometric selectivity needed for this improvement.

5 Conclusion

In this work, we introduced PACE, an offline-to-online RL framework that treats a pretrained flow as both a stable behavioral reference and a geometric foundation for policy improvement. Its central idea is that adapting the guidance of an expressive policy need not require changing its parameters. By combining a frozen flow teacher with an evolving critic, PACE provides continually updated supervision while preserving the offline behavioral prior. Path-aware constraints use the structure of the complete generation process to guide exploration toward better-supported improvements. Experiments across 35 OGBench tasks demonstrate gains in average learning efficiency and final performance under limited online interaction. More broadly, this work points to a role for expressive policies beyond behavior modeling, where learned generative structure actively shapes how policies improve from new experience.

Reproducibility Statement

The supplementary PACE package contains the training code, environment utilities, pinned software requirements, and example commands for reproducing the method. It also includes per-checkpoint FINO and PACE success rates for all 35 tasks and eight seeds, allowing the main tables and learning-efficiency metrics to be recomputed. Appendix A derives the exploration operator, Appendix B specifies the objectives and numerical settings, and Appendix C details evaluation, ablations, and diagnostic measurements. Complete task-level results appear in Appendix D.

AI Use Statement

Generative AI tools assisted with manuscript drafting, language polishing, and organization to improve the clarity and coherence of the presentation. The authors reviewed and revised the resulting text and are responsible for the manuscript's technical content, interpretations, and reported results.

References

Philip J. Ball, Laura Smith, Ilya Kostrikov, and Sergey Levine. Efficient online reinforcement learning with offline data. In *International Conference on Machine Learning*, pp. 1577–1594, 2023. URL https://proceedings.mlr.press/v202/ball23a.html.

Huayu Chen, Cheng Lu, Zhengyi Wang, Hang Su, and Jun Zhu. Score regularized policy optimization through diffusion behavior. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/2a8ce71baac4c89bf9ff479d8240c7d9-Abstract-Conference.html.

Perry Dong, Qiyang Li, Dorsa Sadigh, and Chelsea Finn. EXPO: Stable reinforcement learning with expressive policies. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=aFjSjkB6CV.

Scott Fujimoto, David Meger, and Doina Precup. Off-policy deep reinforcement learning without exploration. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pp. 2052–2062, 2019. URL https://proceedings.mlr.press/v97/fujimoto19a.html.

Philippe Hansen-Estruch, Ilya Kostrikov, Michael Janner, Jakub Grudzien Kuba, and Sergey Levine. IDQL: Implicit Q-learning as an actor-critic method with diffusion policies. *arXiv preprint arXiv:2304.10573*, 2023. URL https://arxiv.org/abs/2304.10573.

Hokyun Im, Andrey Kolobov, Jianlong Fu, and Youngwoon Lee. Latent policy steering through one-step flow policies. *arXiv preprint arXiv:2603.05296*, 2026. URL https://arxiv.org/abs/2603.05296.

Bingyi Kang, Xiao Ma, Chao Du, Tianyu Pang, and Shuicheng Yan. Efficient diffusion policies for offline reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 67195–67212, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/d45e0bfb5a39477d56b55c0824200008-Abstract.html.

Aviral Kumar, Justin Fu, Matthew Soh, George Tucker, and Sergey Levine. Stabilizing off-policy Q-learning via bootstrapping error reduction. In *Advances in Neural Information Processing Systems*, volume 32, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/c2073ffa77b5357a498057413bb09d3a-Abstract.html.

Seunghyun Lee, Younggyo Seo, Kimin Lee, Pieter Abbeel, and Jinwoo Shin. Offline-to-online reinforcement learning via balanced replay and pessimistic Q-ensemble. In *Proceedings of the 5th Conference on Robot Learning*, volume 164, pp. 1702–1712, 2022. URL https://proceedings.mlr.press/v164/lee22d.html.

Qiyang Li and Sergey Levine. Q-learning with adjoint matching. In *The Fourteenth International Conference on Learning Representations*, 2026. URL

https://proceedings.iclr.cc/paper_files/paper/2026/hash/0f85efb1e7545dc35a1b5e4d45aaf3c2-Abstract-Conference.html.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2210.02747.

Mitsuhiko Nakamoto, Yuexiang Zhai, Anikait Singh, Max Sobol Mark, Yi Ma, Chelsea Finn, Aviral Kumar, and Sergey Levine. Cal-QL: Calibrated offline RL pre-training for efficient online fine-tuning. In *Advances in Neural Information Processing Systems*, 2023. URL https://arxiv.org/abs/2303.05479.

Seohong Park, Kevin Frans, Benjamin Eysenbach, and Sergey Levine. OGBench: Benchmarking offline goal-conditioned RL. In *International Conference on Learning Representations*, 2025a. URL https://arxiv.org/abs/2410.20092.

Seohong Park, Qiyang Li, and Sergey Levine. Flow Q-learning. In *International Conference on Machine Learning*, pp. 48104–48127, 2025b. URL https://proceedings.mlr.press/v267/park25f.html.

Hao Ren, Zetong Bi, Yiming Zeng, Le Zheng, Zhi Li, Zhaoliang Wan, Lu Qi, and Hui Cheng. Fisher-preserving guidance: Training-free manifold constraints for safe diffusion control. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2605.29937.

Yongjae Shin, Jongseong Chae, Jongeui Park, and Youngchul Sung. Flow matching with injected noise for offline-to-online reinforcement learning. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2602.18117.

Hikmet Simsir and Ozgur S. Oguz. Lagrangian perturbation diffusion steering: Latent reinforcement learning for generative policies. *arXiv preprint arXiv:2606.01151*, 2026. URL https://arxiv.org/abs/2606.01151.

Denis Tarasov, Vladislav Kurenkov, Alexander Nikulin, and Sergey Kolesnikov. Revisiting the minimalist approach to offline reinforcement learning. In *Advances in Neural Information Processing Systems*, 2023. URL https://papers.neurips.cc/paper_files/paper/2023/hash/26cce1e512793f2072fd27c391e04652-Abstract-Conference.html.

Andrew Wagenmaker, Mitsuhiko Nakamoto, Yunchu Zhang, Seohong Park, Waleed Yagoub, Anusha Nagabandi, Abhishek Gupta, and Sergey Levine. Steering your diffusion policy with latent space reinforcement learning. In *Conference on Robot Learning*, 2025. URL https://arxiv.org/abs/2506.15799.

Zhendong Wang, Jonathan J. Hunt, and Mingyuan Zhou. Diffusion policies as an expressive policy class for offline reinforcement learning. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2208.06193.

Zishun Yu and Xinhua Zhang. Actor-critic alignment for offline-to-online reinforcement learning. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pp. 40452–40474, 2023. URL https://proceedings.mlr.press/v202/yu23k.html.

Yixian Zhang, Shu'ang Yu, Tonghe Zhang, Mo Guang, Haojia Hui, Kaiwen Long, Yu Wang, Chao Yu, and Wenbo Ding. SAC Flow: Sample-efficient reinforcement learning of flow-based policies via velocity-reparameterized sequential modeling, 2025. URL https://arxiv.org/abs/2509.25756.

A Derivation of Path-Constrained Latent Exploration

This section derives the geometric cost and the update used in section 3.2. Throughout, we fix a state s and a reference latent z, and write $J_t = \partial \Phi_t(s, z)/\partial z$. The critic supplies the local improvement direction, while the frozen flow determines the cost of following that direction.

A.1 From path deformation to a quadratic metric

The deformation cost in equation 5 compares the complete trajectories generated from z and $z + \delta$. Its local form follows from expanding the flow at the reference latent.

Proposition 1. *Suppose $\Phi_t(s, \cdot)$ is twice continuously differentiable near z, with first and second derivatives bounded uniformly over $t \in [0, 1]$. Define $H = \int_0^1 J_t^\top J_t \, \mathrm{d}t$. Then*

$$D_{\mathrm{path}}(s, z, \delta) = \delta^\top H \delta + O(\|\delta\|_2^3). \tag{6}$$

Proof. Taylor expansion gives $\Phi_t(s, z+\delta) - \Phi_t(s, z) = J_t\delta + r_t$, where $\|r_t\|_2 \le M\|\delta\|_2^2$ uniformly in t. Squaring produces $\delta^\top J_t^\top J_t \delta + 2\langle J_t\delta, r_t\rangle + \|r_t\|_2^2$. The second and third terms have orders $O(\|\delta\|_2^3)$ and $O(\|\delta\|_2^4)$, respectively. Integrating over $[0, 1]$ gives the result. □

The calibrated metric is $G = H/c + \lambda I$, with $c > 0$ and $\lambda > 0$. Consequently,

$$\delta^\top G \delta = D_{\mathrm{path}}/c + \lambda\|\delta\|_2^2 + O(\|\delta\|_2^3). \tag{7}$$

Calibration sets the overall scale of the deformation penalty. Damping makes the metric positive definite and limits amplification along directions with small path sensitivity. In particular, $G \succeq \lambda I$ implies that the trust constraint also bounds $\|\delta\|_2 \le \rho/\sqrt{\lambda}$.

Writing $G = U \,\mathrm{diag}(\nu_1, \ldots, \nu_d) U^\top$ makes the directional effect explicit. A component $u_i^\top g$ of the value gradient is scaled by $1/\nu_i$ before radius rescaling. Directions that strongly deform the reference path therefore receive smaller latent steps, while directions with lower deformation cost retain more freedom to follow the critic.

A.2 The constrained local value objective

At the actor's reference latent z_π, the normalized critic gradient is $g = J_1^\top \nabla_a \bar{Q}(s, a_\pi)/q$. Holding the current critic and positive scale q fixed, PACE chooses a step by maximizing the linearized critic objective with a quadratic movement cost,

$$\max_{\delta^\top G \delta \le \rho^2} \left\{ g^\top \delta - \frac{1}{2\eta} \delta^\top G \delta \right\}. \tag{8}$$

The parameter η sets the unconstrained proposal size, while ρ caps its length in the flow-induced metric.

Proposition 2. *For $G \succ 0$, $\eta > 0$, and $\rho > 0$, equation 8 has the unique solution in equation 2. If $g \neq 0$, this solution satisfies*

$$g^\top \delta = \kappa \eta g^\top G^{-1} g > 0, \qquad \kappa = \min\left(1, \frac{\rho}{\eta\sqrt{g^\top G^{-1} g}}\right). \tag{9}$$

For $g = 0$, the solution is $\delta = 0$.

Proof. Set $u = G^{1/2}\delta$ and $b = G^{-1/2} g$. The objective becomes $-\|u - \eta b\|_2^2/(2\eta) + \eta\|b\|_2^2/2$, and the feasible region becomes $\|u\|_2 \le \rho$. Its maximizer is the Euclidean projection of ηb onto this ball. Transforming back yields $\delta = \kappa\eta G^{-1} g$. Positive definiteness gives equation 9. □

Thus the update follows an ascent direction for the local critic while respecting a fixed deformation budget. The explored action is then used as a supervised target through equation 3, connecting this local operator to actor learning. Freezing the teacher keeps the mapping used to measure deformation fixed; the metric can still vary with the state and the actor's current reference latent as learning proceeds.

A.3 What full-path geometry adds to endpoint control

Since $\Phi_0(s, z) = z$, the start-point Jacobian is $J_0 = I$. A start-point penalty therefore measures ordinary latent displacement. An endpoint penalty instead measures $\|J_1\delta\|_2^2$, the first-order action displacement. Full-path geometry integrates these sensitivities across generation time, including intermediate directions that neither endpoint alone describes.

The distinction is also visible in action coordinates. For the undamped endpoint metric $C_{\text{end}} = J_1^\top J_1$ and an invertible J_1,

$$J_1 C_{\text{end}}^{-1} J_1^\top = I. \tag{10}$$

Consequently, an endpoint-preconditioned latent gradient becomes a Euclidean action gradient at first order, up to scalar step factors. With the full-path metric, the corresponding action-space direction is

$$J_1 G^{-1} g = \frac{1}{q} \underbrace{J_1 G^{-1} J_1^\top}_{M_{\text{path}}} \nabla_a \bar{Q}(s, a_\pi). \tag{11}$$

The matrix M_{path} incorporates sensitivity along the complete transport. This explains how PACE can permit a larger final action displacement while assigning a smaller cost to the underlying path deformation. The metric describes the frozen flow's transport geometry; empirical action support is measured separately by the offline-data diagnostic in Appendix C.4.

B Training Objectives and Implementation

B.1 Teacher pretraining and actor–critic learning

PACE first learns the flow teacher from offline state–action pairs. With $z \sim \mathcal{N}(0, I)$, $t \sim \mathcal{U}[0, 1]$, and $x_t = (1 - t)z + ta$, its flow-matching objective is

$$\mathcal{L}_{\text{FM}}(\theta) = \mathbb{E}_{(s,a)\sim\mathcal{D}_{\text{off}},z,t}\left[\frac{1}{d}\left\|v_\theta(s, x_t, t) - (a - z)\right\|_2^2\right]. \tag{12}$$

After 61,000 updates with batches of 4,096, the teacher parameters are fixed. The one-step actor and twin critics are initialized for the subsequent offline RL stage, which uses one million gradient updates. Online learning then adds 500,000 environment transitions to replay, with one actor–critic update per interaction. The replay buffer is initialized with the offline dataset, and online batches sample from the combined buffer.

For the actor, the original teacher output $a_B = \Phi_1(s, \epsilon)$ and the explored output a_{imp} provide complementary supervision. Using the notation of section 3,

$$\begin{aligned} \mathcal{L}_{\text{FQL}} &= \mathbb{E}_{s,\epsilon}\left[-Q^{\text{actor}}(s, a_\pi) + \frac{\alpha}{d}\|a_\pi - a_B\|_2^2\right], \\ \mathcal{L}_\pi &= \mathcal{L}_{\text{FQL}} + \alpha\beta\mathbb{E}_{s,\epsilon}\left[\frac{\|a_\pi - a_{\text{imp}}\|_2^2}{d}\right]. \end{aligned} \tag{13}$$

Here, d denotes the action dimension. Actions passed to the critic or environment are clipped to the action bounds. Ordinary distillation and the actor's value objective use all 256 batch examples. The explored-target loss is averaged over a sub-batch of 32 examples. Both targets are held constant during actor optimization, so gradients update the one-step actor through its prediction a_π.

Exploration supervision is introduced gradually to allow the critic to learn before it supplies strong auxiliary targets. At offline update u, the multiplier is $\beta w(u)$, where

$$w(u) = \min\left(1, \max\left(0, \frac{u - 100{,}000}{100{,}000}\right)\right). \tag{14}$$

The full multiplier β is used after 200,000 offline updates and throughout online learning. This schedule changes only the explored-target term; the FQL actor objective remains active throughout.

Critic training retains the standard squared Bellman loss. For a replay transition (s, a, r, s', m), with bootstrap mask m, draw one next action $a' = \mu_\omega(s', \epsilon')$ and form

$$y = r + \gamma m\, \bar{Q}(s', a'), \qquad \mathcal{L}_Q = \frac{1}{2}\sum_{i=1}^{2}\mathbb{E}\left[(Q_i(s, a) - y)^2\right]. \tag{15}$$

Table 3: Shared PACE configuration.

Parameter	Value
Flow-only pretraining updates	61,000
Flow pretraining batch size	4,096
Offline actor-critic updates	1,000,000
Online environment steps	500,000
Actor-critic batch size	256
PACE target sub-batch size	32
Forward Euler steps	10
Backward Euler steps	10
Latent step size η	10
Metric trust radius ρ	0.05
Metric damping λ	0.01
Q-scale floor	0.05
PACE offline warmup	100,000 updates
PACE offline linear ramp	100,000 updates
Metric calibration states	16
Learning rate	3×10^{-4}
Target-critic update rate	0.005
Evaluation episodes per checkpoint	50
Online evaluation interval	50,000 steps

Table 4: Task-family hyperparameters and candidate budgets.

Family	α	γ	β	K	Backup / PACE Q
HumanoidMaze Medium	100	0.995	30	10	mean / mean
HumanoidMaze Large	30	0.995	10	10	mean / mean
AntMaze Large	10	0.990	100	2	min / min
AntMaze Giant	10	0.995	30	2	min / min
AntSoccer Arena	30	0.995	10	2	mean / mean
Cube Double	300	0.990	10	1	mean / mean
Puzzle 4×4	1000	0.990	10	1	mean / mean

The target critics use Polyak updates with coefficient 0.005. Their aggregation is specified in Table 4; the actor's direct value objective uses the mean of the current twin critics for every family.

B.2 Architectures and hyperparameters

The teacher velocity field, one-step actor, and each critic use four hidden layers of width 512 with GELU activations. Layer normalization is applied to the critics. All networks are optimized with Adam at learning rate 3×10^{-4}. Table 3 collects the shared settings. Table 4 gives the task-dependent values, shared across the five tasks within each family. The weight α follows FQL's distillation settings, while β controls the additional explored-target loss. The exploration inverse temperature used for candidate selection is a separate parameter.

B.3 Numerical construction of the path metric

Both forward generation and backward integration use $N = 10$ Euler steps. The backward path starts at $x_1 = a_\pi$ and follows

$$x_{t_{j-1}} = x_{t_j} - \frac{1}{N} v_{\theta_0}(s, x_{t_j}, t_j), \qquad t_j = \frac{j}{N}, \quad j = N, \dots, 1. \tag{16}$$

Its initial point supplies z_π, and its intermediate states supply the reference trajectory. To reuse derivatives along this trajectory, let $A_t = \partial x_t / \partial x_1$. The chain rule gives $A_t = J_t J_1^{-1}$ and

$A_0 = J_1^{-1}$, hence $J_t = A_t A_0^{-1}$. Differentiating the backward path and solving these small action-dimensional systems supplies the Jacobians at all stored times.

We use trapezoidal weights to integrate their squared sensitivities,

$$\widehat{H} = \frac{1}{N}\left[\frac{1}{2}J_0^\top J_0 + \sum_{j=1}^{N-1} J_{t_j}^\top J_{t_j} + \frac{1}{2}J_1^\top J_1\right], \qquad \widehat{G} = \widehat{H}/c + \lambda I. \tag{17}$$

The scale c is calibrated once, before actor–critic training, from 16 offline states and Gaussian inputs to the pretrained teacher. Specifically, c is the median of $\mathrm{tr}(\widehat{H})/d$ across these examples, with a lower bound of 10^{-6}. This scalar fixes the overall metric scale while preserving its directional structure. The state-dependent matrix is recomputed at each exploration point and symmetrized before solving the linear system.

We set $\eta = 10$, $\rho = 0.05$, and $\lambda = 0.01$ in the update of equation 2. Value normalization uses

$$q = \max\left(\frac{1}{32}\sum_{i=1}^{32}|\bar{Q}(s_i, a_{\pi,i})|,\ 0.05\right). \tag{18}$$

The floor stabilizes the gradient scale when predicted values are close to zero. Target construction requires a metric solve, radius rescaling, and decoding of the explored latent; the resulting target enters the sub-batch loss in equation 13. The flow is evaluated and differentiated with respect to its inputs while its parameters remain fixed.

B.4 Action selection and execution

Following FINO (Shin et al., 2026), the actor generates K candidates from independent Gaussian inputs. Online collection samples a candidate using normalized Q scores,

$$p_i = \frac{\exp(\xi q_i/\overline{|q|})}{\sum_{j=1}^{K}\exp(\xi q_j/\overline{|q|})}, \qquad \overline{|q|} = \frac{1}{K}\sum_{j=1}^{K}|q_j|. \tag{19}$$

The nonnegative inverse temperature ξ follows FINO's entropy-guided adaptation, with entropy estimated from the generated actions using a Gaussian mixture. Evaluation selects the highest-Q candidate. The critic bootstrap in equation 15 continues to use a single sampled next action.

Candidate counts are $K = 2$ for AntMaze and AntSoccer, $K = 10$ for HumanoidMaze, and $K = 1$ for Cube Double and Puzzle 4×4. With $K = 1$, the sampled one-step action is executed directly. Environment interaction uses the one-step actor; inverse integration and path-metric computation are confined to training.

B.5 Supplementary implementation and result files

The supplementary package provides `train.py`, the PACE agent and path-metric routines, environment and evaluation utilities, and a README with task-family settings and example commands. Its pinned runtime uses Python 3.9, JAX 0.4.30, Flax 0.8.5, Optax 0.2.4, OGBench 1.1.0, and MuJoCo 3.3.0. The headless rendering setup uses Xvfb with MuJoCo's GLFW backend. Full dependency versions appear in `requirement.txt`.

The file `results/pace_fino_8seeds.csv` contains 560 records, one for each method–task–seed combination. Each record stores the model name, task identifier, training seed, and eleven pairs of online step and success percentage. These measurements support recomputation of the main FINO/PACE tables, learning curves, and AUC. Training writes the primary evaluation curve to `eval_hard_argmax.csv`.

C Experimental Protocols and Additional Diagnostics

C.1 Benchmark and baseline comparison

We evaluate 35 OGBench tasks, with five tasks in each of seven families. AntMaze Large and Giant, HumanoidMaze Medium and Large, and AntSoccer Arena use navigation datasets; Cube Double

and Puzzle 4×4 use play datasets. The single-task setting fixes the task goal during learning and evaluation. Each PACE and FINO experiment uses eight training seeds, giving 280 runs per method.

ReBRAC, Cal-QL, RLPD, IFQL, and FQL results are taken from Tables 1 and 4 of Shin et al. (2026), using their published ten-seed statistics. FINO's main entry is our eight-seed reproduction. For HumanoidMaze Large and AntSoccer, where the reproduction differs substantially from the published results, the original FINO online score is also shown in right-hand parentheses. This convention is used consistently in the family and task-level tables. The FQL diagnostic controls in figure 1 are separate experiments described in Appendix C.5.

Our online evaluations use 50 episodes at each of eleven checkpoints, from step 0 through 500,000 in increments of 50,000. A common evaluation seed, 20260807, is used across the comparisons. The step-zero score is the offline-1M checkpoint. The main learning curves and tables use the same FINO/PACE measurements. Candidate budgets, score aggregation, and the highest-Q evaluation policy follow Appendix B.4.

C.2 Performance metrics and confidence intervals

Let R_j be a run's evaluation success percentage at $h_j = 50{,}000j$, for $j = 0, \ldots, 10$. We report offline success R_0, final online success R_{10}, and normalized area under the online learning curve,

$$\mathrm{AUC} = \frac{1}{500{,}000} \sum_{j=0}^{9} (h_{j+1} - h_j) \frac{R_j + R_{j+1}}{2}. \tag{20}$$

This normalization places AUC on the same 0–100 scale as success. It measures performance throughout the interaction budget, rewarding methods that reach useful behavior earlier as well as those with strong final performance.

For n training seeds, every confidence interval computed from our runs has half-width

$$\mathrm{CI}_{95} = t_{n-1,0.975} \frac{\widehat{\sigma}}{\sqrt{n}}, \qquad \widehat{\sigma}^2 = \frac{1}{n-1} \sum_{i=1}^{n} (R^{(i)} - \bar{R})^2. \tag{21}$$

The main comparison uses $n = 8$; ablations and frozen-teacher controls use $n = 4$. For a family or overall aggregate, task scores are first averaged equally within each seed, and the interval is computed across these seed-level averages. The same procedure applies to per-run AUC and to each point on an aggregate learning curve. Published baseline intervals retain the source paper's convention.

Tables display means $\pm$ 95% CI half-widths, rounded to the nearest integer, with half values rounded upward. Calculations and boldface selection use unrounded measurements. Online means within two percentage points of the best row mean are bolded; AUC uses the same rule within its comparison. Learning-curve confidence bands are clipped to the success range $[0, 100]$ for display.

C.3 Component ablation definitions

The ablations in Table 2 use Task 4 of all seven families and training seeds 0–3. They preserve the actor–critic architecture, training budget, task-family settings, and candidate-selection protocol. Full PACE uses the corresponding four-seed subset of the main experiment.

The continually updated-flow variant retains the exploration target but updates the teacher by behavior cloning throughout RL. It begins with standard FQL initialization, and calibrates the metric scale from that initial flow. The variant without exploration targets keeps the pretrained teacher frozen and sets the auxiliary target weight to zero. Together, these comparisons examine the value of a stable teacher and the additional guidance extracted from it.

The geometry ablations retain the frozen teacher and explored-target loss but replace the matrix in equation 2. Their exact costs are

$$\begin{aligned} G_{\mathrm{S}} &= I, \\ G_{\mathrm{E}} &= J_1^\top J_1/c + \lambda I, \\ G_{\mathrm{SE}} &= I + J_1^\top J_1/c + \lambda I, \\ G_{\mathrm{path}} &= \widehat{H}/c + \lambda I. \end{aligned} \tag{22}$$

The scale c and damping λ are the same quantities used by PACE. Start-point control measures displacement in latent space; endpoint control measures its decoded action effect; their combination includes both boundary penalties. Full-path control additionally accounts for the intermediate generation states through equation 17. These comparisons use the same proposed-step scale and metric trust radius.

Each of the five ablation variants has 28 runs. Eight continually updated-flow runs, covering all four seeds on Cube Double and Puzzle 4×4, encounter non-finite values and are scored as zero for the reported offline and online results. These entries are marked with a dagger and included in the average. All other ablation runs complete training. Both ablation scores and the full-PACE comparison use the mean and 95% CI convention in Appendix C.2.

C.4 Flow and actor diagnostic protocols

Teacher distribution and peripheral spread. The density panels in figure 1 compare the FQL flow teacher at offline 1M and online +500k on Cube Double Task 4, seed 3. We hold the observation fixed at offline dataset index 528883, selected by screening the offline checkpoint for multimodal generated actions. Both checkpoints receive the same 8,192 Gaussian inputs. Their five-dimensional actions are clipped as in execution and projected into a shared two-dimensional PCA coordinate system. The PCA is fitted to balanced samples from all eleven checkpoints, and its first two components retain 89.1% of the action variance.

Each projected distribution is estimated on the same grid with Gaussian bandwidth 0.025 and displayed on a shared logarithmic color scale. A probability-mass region is obtained by lowering the density threshold until the enclosed grid mass reaches the specified fraction. The 99% region expands to 2.16 times its offline area, indicating broader peripheral structure after online updates. The accompanying performance controls test the benefit of retaining a fixed behavioral reference and are detailed in Appendix C.5.

Actor departure from the teacher. Figure 3 compares start/end constraints with full-path constraints on Cube Double Task 4, seed 0, at online +500k. We use 256 fixed offline states and 1,024 common Gaussian inputs at each state. For either method, departure is measured in the full five-dimensional action space relative to its own frozen teacher under the same input,

$$d_{\mathrm{teacher}}(s, \epsilon) = \frac{\|\mu_\omega(s, \epsilon) - \Phi_1(s, \epsilon)\|_2}{\sqrt{5}}. \tag{23}$$

The population mean is 0.0286 with start/end constraints and 0.0382 with full-path constraints. The action bounds used for execution are also applied when computing these diagnostics.

Empirical offline support. The OOD rule is constructed from offline data independently of either method's path metric. We select 50,000 reference transitions, standardize their complete observations, and find the 64 nearest reference observations for each query state. Let $B_{64}(s)$ contain their corresponding actions. The action-space support distance and binary OOD indicator are

$$d_{\mathrm{off}}(s, a) = \min_{b \in B_{64}(s)} \frac{\|a - b\|_2}{\sqrt{5}}, \qquad \mathrm{OOD}_{\mathrm{emp}}(s, a) = \mathbf{1}\{d_{\mathrm{off}}(s, a) > \tau\}. \tag{24}$$

The threshold $\tau \simeq 0.186848$ is the 95th percentile of leave-one-out distances for 2,048 offline calibration transitions. We apply this same neighborhood construction and threshold to both methods, and compute OOD rates directly from the generated actions. Across all 256 states, start/end constraints give 3.46% OOD and full-path constraints give 3.12%. Thus the lower empirical OOD rate accompanies a larger mean departure from the teacher.

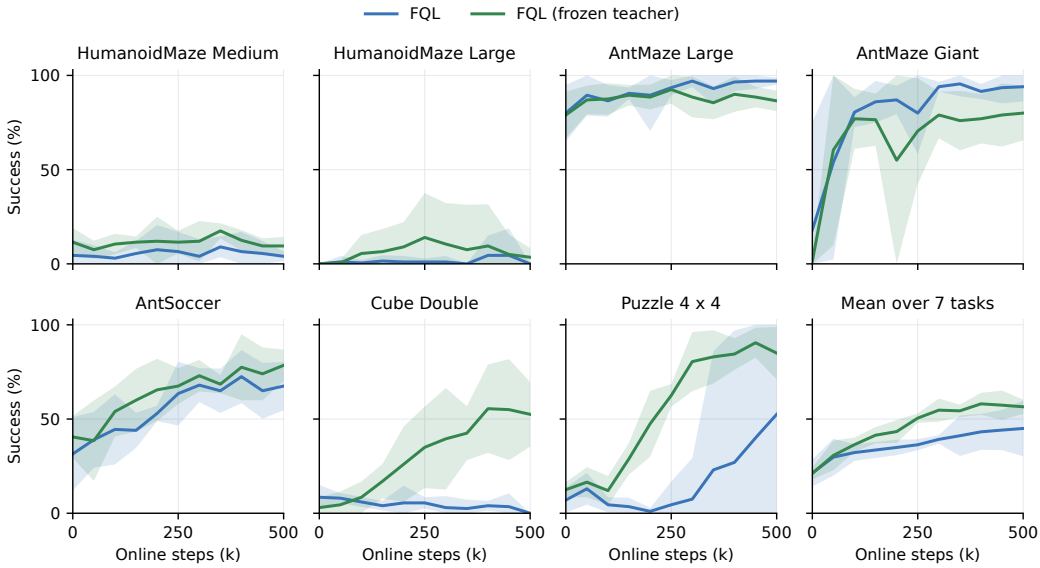

Figure 6: Standard FQL and frozen-teacher FQL on Task 4 of seven families and their task average. Curves show four-seed means with 95% CIs.

Representative-state visualization. The actor figure displays one state, index 609577, selected by its smallest relative Euclidean distance to the population OOD and departure statistics of both methods. Its horizontal coordinate is d_{teacher} and its vertical coordinate is d_{off}/τ. The boundary at height one therefore directly separates actions according to equation 24. Shared axes and kernel bandwidths are used for both methods; contours enclose 50%, 80%, 95%, and 99% of the smoothed sample mass, and diamonds mark means.

The percentages printed in the figure report the 256-state aggregate. At the displayed state itself, OOD rates are 4.79% and 2.34%, and mean departures are 0.0185 and 0.0430 for start/end and full-path constraints, respectively. At this state, full-path constraints likewise allow greater departure from the teacher with lower empirical OOD, supporting the population-level conclusion. The greater-departure conclusion holds when either teacher is used as a common reference, and the lower-OOD conclusion holds when the support threshold changes by $\pm 10\%$ at this state.

C.5 Complete frozen-teacher control comparison

The controls underlying figure 1 compare ordinary FQL with FQL using a separately pretrained, frozen flow teacher. They use Task 4 of each family, four training seeds, one million offline actor–critic updates, and 500,000 online interactions. Both retain ordinary FQL distillation and single-action execution. The frozen teacher receives the pretraining described in Appendix B.1; these controls omit PACE exploration targets.

Figure 6 shows the complete learning curves, and Table 5 reports the offline and final online scores. The frozen-teacher variant improves final success on five of the seven tasks and raises average final success from 45.0% to 56.5%. Standard FQL performs better on the two AntMaze tasks. On Cube Double Task 4, the frozen variant's final scores are 64%, 54%, 38%, and 54%, yielding $52.5\% \pm 17.1\%$, compared with zero in all four standard-FQL seeds. The aggregate curve first averages tasks within each seed and then applies the confidence interval in Appendix C.2.

Table 5: FQL controls before and after 500k online interactions. Entries give offline → online success (%) as means ± 95% CI half-widths across four seeds. The Mean row averages the seven tasks within each seed.

Task	FQL	FQL (frozen teacher)
HumanoidMaze Medium Task 4	$5 \pm 8 \rightarrow 4 \pm 3$	$12 \pm 8 \rightarrow 10 \pm 5$
HumanoidMaze Large Task 4	$0 \pm 0 \rightarrow 0 \pm 0$	$0 \pm 0 \rightarrow 4 \pm 5$
AntMaze Large Task 4	$80 \pm 15 \rightarrow 97 \pm 2$	$79 \pm 12 \rightarrow 87 \pm 5$
AntMaze Giant Task 4	$18 \pm 57 \rightarrow 94 \pm 8$	$2 \pm 5 \rightarrow 80 \pm 14$
AntSoccer Task 4	$32 \pm 20 \rightarrow 68 \pm 13$	$41 \pm 11 \rightarrow 79 \pm 8$
Cube Double Task 4	$9 \pm 7 \rightarrow 0 \pm 0$	$3 \pm 6 \rightarrow 53 \pm 17$
Puzzle 4×4 Task 4	$7 \pm 7 \rightarrow 53 \pm 86$	$13 \pm 4 \rightarrow 85 \pm 14$
Mean	$21 \pm 7 \rightarrow 45 \pm 15$	$21 \pm 3 \rightarrow 57 \pm 4$

D Complete Task-Level Results

Table 6 expands the family comparison in Table 1 into all 35 tasks, reporting offline → online success before and after 500k interactions. Each score gives the mean with ± its 95% CI half-width. ReBRAC, Cal-QL, RLPD, IFQL, and FQL (†) use the published ten-seed results in Shin et al. (2026). FINO and PACE use our eight-seed experiment. Right-hand parentheses add the published FINO online final score for HumanoidMaze Large and AntSoccer only. Online scores within 2 percentage points of the highest mean for that task are bold. Table 7 additionally reports normalized AUC for our FINO and PACE runs, using equation 20. Together, the tables show both the offline-to-online change and performance accumulated throughout the interaction budget.

Table 6: Offline → online success (%) on all 35 tasks.

Task	ReBRAC†	Cal-QL†	RLPD†	IFQL†	FQL†	FINO	PACE
HumanoidMaze Medium Task 1	14 ±7 → 1 ±1	0 ±0 → 0 ±0	0 ±0 → 0 ±0	65 ±18 → 45 ±11	11 ±5 → 14 ±4	73 ±25 → 78 ±27	24 ±31 → **97** ±1
HumanoidMaze Medium Task 2	18 ±9 → 1 ±1	0 ±0 → 0 ±0	0 ±0 → 0 ±0	92 ±4 → 83 ±7	89 ±13 → 90 ±3	95 ±2 → **98** ±1	82 ±28 → **99** ±1
HumanoidMaze Medium Task 3	30 ±14 → 1 ±1	0 ±0 → 0 ±0	0 ±0 → 1 ±1	38 ±30 → 74 ±18	56 ±23 → 89 ±2	53 ±32 → 88 ±17	60 ±28 → **93** ±13
HumanoidMaze Medium Task 4	17 ±11 → 1 ±1	0 ±0 → 0 ±0	0 ±0 → 0 ±0	0 ±0 → 56 ±9	9 ±12 → 18 ±6	61 ±8 → 85 ±7	13 ±20 → **95** ±4
HumanoidMaze Medium Task 5	28 ±14 → 10 ±16	0 ±0 → 0 ±0	0 ±0 → 4 ±3	99 ±1 → 92 ±3	100 ±0 → 94 ±3	96 ±3 → **100** ±1	99 ±1 → **100** ±1
HumanoidMaze Large Task 1	1 ±1 → 0 ±0	0 ±0 → 0 ±0	0 ±0 → 0 ±0	1 ±1 → 0 ±0	4 ±3 → 1 ±1	5 ±7 → 14 ±20 (5 ±9)	0 ±0 → **23** ±26
HumanoidMaze Large Task 2	0 ±0 → 0 ±0	0 ±0 → 0 ±0	0 ±0 → 0 ±0	0 ±0 → 0 ±0	0 ±0 → 0 ±0	0 ±0 → 0 ±0 (**6** ±6)	0 ±0 → **4** ±5
HumanoidMaze Large Task 3	9 ±6 → 2 ±2	0 ±0 → 0 ±0	0 ±0 → 0 ±0	45 ±10 → 41 ±7	17 ±6 → 36 ±11	16 ±11 → 84 ±29 (**99** ±1)	8 ±7 → 74 ±38
HumanoidMaze Large Task 4	1 ±1 → 1 ±1	0 ±0 → 0 ±0	0 ±0 → 0 ±0	0 ±0 → 5 ±3	2 ±2 → 8 ±8	0 ±0 → 11 ±25 (**48** ±29)	0 ±0 → 29 ±29
HumanoidMaze Large Task 5	1 ±1 → 0 ±1	0 ±0 → 0 ±0	0 ±0 → 0 ±0	8 ±10 → 5 ±3	1 ±1 → 5 ±7	0 ±1 → 0 ±0 (8 ±16)	5 ±11 → **10** ±24
AntMaze Large Task 1	94 ±4 → **100** ±0	20 ±26 → 30 ±30	0 ±0 → 93 ±8	32 ±10 → 66 ±15	82 ±5 → **98** ±1	64 ±27 → **98** ±1	76 ±16 → **99** ±2
AntMaze Large Task 2	88 ±2 → **98** ±1	0 ±0 → 0 ±0	0 ±0 → 54 ±25	17 ±7 → 70 ±4	63 ±5 → 71 ±4	72 ±9 → 94 ±5	68 ±10 → 94 ±3
AntMaze Large Task 3	65 ±14 → **100** ±0	20 ±26 → 10 ±20	0 ±0 → **99** ±1	57 ±9 → 88 ±4	94 ±2 → **100** ±1	96 ±3 → **100** ±0	82 ±20 → **100** ±0
AntMaze Large Task 4	86 ±5 → **99** ±1	0 ±0 → 0 ±0	0 ±0 → 86 ±11	13 ±5 → 75 ±8	80 ±5 → 96 ±1	73 ±7 → **98** ±2	71 ±10 → **99** ±1
AntMaze Large Task 5	90 ±4 → **100** ±1	20 ±26 → 20 ±26	0 ±0 → 69 ±24	39 ±10 → 59 ±23	85 ±4 → 95 ±2	85 ±6 → **99** ±2	91 ±6 → **98** ±1
AntMaze Giant Task 1	53 ±16 → **97** ±1	0 ±0 → 0 ±0	0 ±0 → 7 ±12	0 ±0 → 0 ±0	8 ±7 → 65 ±24	16 ±9 → 67 ±35	20 ±19 → 36 ±27
AntMaze Giant Task 2	25 ±17 → **98** ±1	0 ±0 → 0 ±0	0 ±0 → 48 ±24	0 ±0 → 0 ±0	17 ±11 → **96** ±1	1 ±2 → **98** ±2	1 ±1 → 95 ±3
AntMaze Giant Task 3	34 ±20 → **86** ±19	0 ±0 → 0 ±0	0 ±0 → 44 ±18	0 ±0 → 0 ±0	0 ±1 → 2 ±2	0 ±0 → 8 ±14	0 ±0 → 12 ±21
AntMaze Giant Task 4	0 ±0 → **98** ±1	0 ±0 → 0 ±0	0 ±0 → 59 ±26	0 ±0 → 0 ±0	10 ±13 → **96** ±2	9 ±21 → 87 ±26	13 ±21 → 93 ±4
AntMaze Giant Task 5	61 ±12 → **99** ±1	10 ±20 → 0 ±0	0 ±0 → 80 ±12	4 ±4 → 0 ±0	43 ±21 → **99** ±1	7 ±15 → **99** ±1	28 ±26 → **99** ±1
AntSoccer Task 1	1 ±1 → 0 ±0	0 ±0 → 0 ±0	0 ±0 → 6 ±4	69 ±15 → 64 ±10	82 ±4 → 91 ±2	64 ±8 → 46 ±9 (**93** ±2)	67 ±10 → **94** ±4
AntSoccer Task 2	0 ±1 → 0 ±0	0 ±0 → 0 ±0	0 ±0 → 5 ±4	70 ±7 → 66 ±21	88 ±4 → **97** ±3	43 ±7 → 57 ±17 (**98** ±1)	45 ±13 → 90 ±4
AntSoccer Task 3	0 ±0 → 0 ±0	0 ±0 → 0 ±0	0 ±0 → 1 ±1	6 ±6 → 26 ±9	60 ±4 → 88 ±4	46 ±10 → 80 ±6 (91 ±2)	49 ±6 → **95** ±3
AntSoccer Task 4	1 ±1 → 0 ±0	0 ±0 → 0 ±0	0 ±0 → 0 ±0	20 ±9 → 17 ±6	32 ±4 → 70 ±5	48 ±9 → 55 ±6 (70 ±6)	44 ±8 → **89** ±4
AntSoccer Task 5	0 ±0 → 0 ±0	0 ±0 → 0 ±0	0 ±0 → 0 ±0	1 ±2 → 4 ±3	43 ±9 → 22 ±18	45 ±15 → 1 ±2 (33 ±25)	46 ±17 → **89** ±4
Cube Double Task 1	27 ±8 → **100** ±0	10 ±20 → 0 ±0	0 ±0 → 11 ±15	31 ±5 → 89 ±3	59 ±9 → 97 ±2	75 ±7 → **98** ±2	63 ±8 → **100** ±0
Cube Double Task 2	8 ±3 → 20 ±6	0 ±0 → 0 ±0	0 ±0 → 0 ±0	13 ±3 → 29 ±4	42 ±9 → 86 ±7	55 ±9 → 85 ±15	31 ±8 → **99** ±1
Cube Double Task 3	3 ±2 → 16 ±7	0 ±0 → 0 ±0	0 ±0 → 0 ±0	6 ±2 → 31 ±7	29 ±5 → 89 ±9	51 ±13 → 93 ±5	27 ±9 → **98** ±1
Cube Double Task 4	1 ±1 → 0 ±1	0 ±0 → 0 ±0	0 ±0 → 0 ±0	1 ±1 → 0 ±0	5 ±3 → 4 ±2	12 ±4 → 13 ±7	4 ±4 → **79** ±5
Cube Double Task 5	3 ±2 → 3 ±2	0 ±0 → 0 ±0	0 ±0 → 0 ±0	17 ±3 → 51 ±6	20 ±6 → 88 ±3	45 ±14 → 86 ±9	34 ±12 → **97** ±2
Puzzle 4 × 4 Task 1	25 ±4 → **100** ±0	10 ±20 → 70 ±30	0 ±0 → **100** ±0	38 ±6 → **100** ±1	32 ±6 → **100** ±0	57 ±10 → **100** ±0	59 ±9 → **100** ±1
Puzzle 4 × 4 Task 2	10 ±4 → 0 ±0	0 ±0 → 0 ±0	0 ±0 → 40 ±32	16 ±5 → 1 ±1	13 ±4 → 0 ±0	14 ±6 → 0 ±0	20 ±2 → **99** ±2
Puzzle 4 × 4 Task 3	16 ±4 → 42 ±26	0 ±0 → 30 ±30	0 ±0 → 89 ±19	49 ±8 → 91 ±12	18 ±5 → 70 ±28	50 ±13 → **100** ±0	43 ±7 → **100** ±1
Puzzle 4 × 4 Task 4	8 ±2 → 3 ±3	0 ±0 → 0 ±0	0 ±0 → 40 ±32	20 ±4 → 19 ±17	7 ±4 → 53 ±30	19 ±8 → 57 ±41	19 ±5 → **99** ±1
Puzzle 4 × 4 Task 5	9 ±2 → 0 ±0	0 ±0 → 0 ±0	0 ±0 → 20 ±26	8 ±3 → 0 ±0	5 ±2 → 0 ±0	9 ±6 → 0 ±0	10 ±6 → **100** ±0

Table 7: Additional eight-seed FINO and PACE results. Cells give Offline $\rightarrow$ Online; normalized AUC, in percentage units.

Task	FINO	PACE
HumanoidMaze Medium Task 1	$73 \rightarrow 78$; 79	$24 \rightarrow 97$; 70
HumanoidMaze Medium Task 2	$95 \rightarrow 98$; 97	$82 \rightarrow 99$; 97
HumanoidMaze Medium Task 3	$53 \rightarrow 88$; 81	$60 \rightarrow 93$; 86
HumanoidMaze Medium Task 4	$61 \rightarrow 85$; 78	$13 \rightarrow 95$; 76
HumanoidMaze Medium Task 5	$96 \rightarrow 100$; 99	$99 \rightarrow 100$; 99
HumanoidMaze Large Task 1	$5 \rightarrow 14$; 4	$0 \rightarrow 23$; 15
HumanoidMaze Large Task 2	$0 \rightarrow 0$; 0	$0 \rightarrow 4$; 2
HumanoidMaze Large Task 3	$16 \rightarrow 84$; 69	$8 \rightarrow 74$; 58
HumanoidMaze Large Task 4	$0 \rightarrow 11$; 14	$0 \rightarrow 29$; 12
HumanoidMaze Large Task 5	$0 \rightarrow 0$; 0	$5 \rightarrow 10$; 13
AntMaze Large Task 1	$64 \rightarrow 98$; 93	$76 \rightarrow 99$; 98
AntMaze Large Task 2	$72 \rightarrow 94$; 90	$68 \rightarrow 94$; 91
AntMaze Large Task 3	$96 \rightarrow 100$; 99	$82 \rightarrow 100$; 99
AntMaze Large Task 4	$73 \rightarrow 98$; 94	$71 \rightarrow 99$; 95
AntMaze Large Task 5	$85 \rightarrow 99$; 95	$91 \rightarrow 98$; 98
AntMaze Giant Task 1	$16 \rightarrow 67$; 69	$20 \rightarrow 36$; 49
AntMaze Giant Task 2	$1 \rightarrow 98$; 87	$1 \rightarrow 95$; 82
AntMaze Giant Task 3	$0 \rightarrow 8$; 11	$0 \rightarrow 12$; 24
AntMaze Giant Task 4	$9 \rightarrow 87$; 81	$13 \rightarrow 93$; 86
AntMaze Giant Task 5	$7 \rightarrow 99$; 94	$28 \rightarrow 99$; 94
AntSoccer Task 1	$64 \rightarrow 46$; 47	$67 \rightarrow 94$; 85
AntSoccer Task 2	$43 \rightarrow 57$; 50	$45 \rightarrow 90$; 74
AntSoccer Task 3	$46 \rightarrow 80$; 62	$49 \rightarrow 95$; 79
AntSoccer Task 4	$48 \rightarrow 55$; 48	$44 \rightarrow 89$; 73
AntSoccer Task 5	$45 \rightarrow 1$; 19	$46 \rightarrow 89$; 73
Cube Double Task 1	$75 \rightarrow 98$; 94	$63 \rightarrow 100$; 97
Cube Double Task 2	$55 \rightarrow 85$; 71	$31 \rightarrow 99$; 91
Cube Double Task 3	$51 \rightarrow 93$; 75	$27 \rightarrow 98$; 91
Cube Double Task 4	$12 \rightarrow 13$; 14	$4 \rightarrow 79$; 54
Cube Double Task 5	$45 \rightarrow 86$; 74	$34 \rightarrow 97$; 91
Puzzle 4×4 Task 1	$57 \rightarrow 100$; 97	$59 \rightarrow 100$; 97
Puzzle 4×4 Task 2	$14 \rightarrow 0$; 1	$20 \rightarrow 99$; 51
Puzzle 4×4 Task 3	$50 \rightarrow 100$; 73	$43 \rightarrow 100$; 92
Puzzle 4×4 Task 4	$19 \rightarrow 57$; 24	$19 \rightarrow 99$; 67
Puzzle 4×4 Task 5	$9 \rightarrow 0$; 1	$10 \rightarrow 100$; 70
Mean	$41 \rightarrow 65$; 59	$37 \rightarrow 82$; 72

E LIMITATIONS AND FUTURE WORK

Understanding when expressive behavioral priors support stable and sustained online policy improvement is a broader research challenge. PACE addresses the role of a stable behavioral reference and flow geometry in this process, providing empirical evidence that a frozen teacher with path-aware exploration can support effective online adaptation. The analysis in Appendix A characterizes local target construction within this design. A more complete understanding would also need to account for the coupled evolution of value estimates, policy behavior, and the replay distribution. Our current analysis leaves open how these factors interact with behavioral regularization to determine task-specific bottlenecks and the conditions for sustained improvement. The remaining performance gaps and late-training declines, documented in Appendix D and the checkpoint records described in Appendix B.5, motivate extending this understanding beyond the mechanisms examined here.

Future work could develop a more predictive account of these coupled learning dynamics by combining theoretical analysis with controlled empirical diagnosis. Starting from simplified teacher–student actor–critic settings, theoretical analysis could examine how value-estimation errors propagate through policy updates and shifts in the visited state–action distribution. This could clarify how a fixed behavioral reference and path-constrained exploration affect error propagation, and under what conditions other factors become limiting. Controlled experiments that isolate the effects of critic accuracy, offline-data coverage, and behavioral constraint strength could test these explanations and distinguish the causes of stalled improvement from those of performance degradation.

Such understanding could then inform algorithm design. Measurable diagnostics could help determine when to improve value estimation, broaden exploration, or adjust the influence of the behavioral reference. Connecting these diagnostics to the underlying learning dynamics could provide principles for adapting exploration strength and distillation weights across tasks and training stages.