One-Step Flow Q-Learning: Addressing the Diffusion Policy Bottleneck in Offline Reinforcement Learning

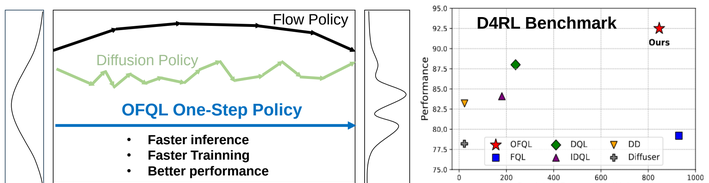
Thanh Nguyen, Chang D. Yoo
Korea Advanced Institute of Science and Technology (KAIST)

ICLR 2026 Rio de Janeiro

Paper | Project

TL;DR: OFQL transforms diffusion policy into one-step action generation policy via a learned average velocity field—simpler, faster, and state-of-the-art on D4RL.

Overview

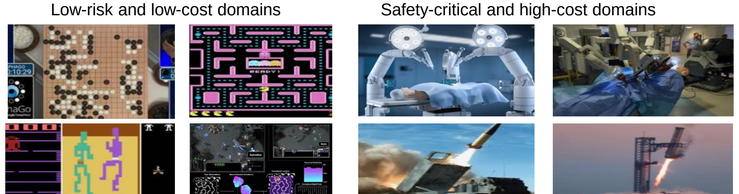


One-Step Flow Q-Learning (OFQL) enables direct one-step action generation for both training and inference, eliminating multi-step denoising, auxiliary modules, and distillation. By learning an average velocity field within the Flow Matching framework, it achieves faster, more stable learning. On D4RL, it reduces computation while significantly outperforming multi-step diffusion-based Q-learning.

Motivation

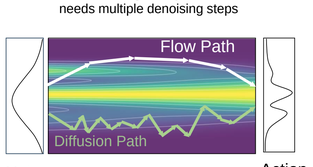
Online Reinforcement Learning
Low-risk and low-cost domains

Offline Reinforcement Learning
Safety-critical and high-cost domains

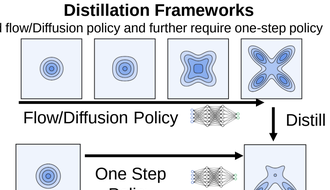


This paper addresses challenges in Offline Reinforcement Learning: the topic unlocks high-value applications where real-world interaction is costly or unsafe.

Diffusion/Flow policy frameworks
needs multiple denoising steps

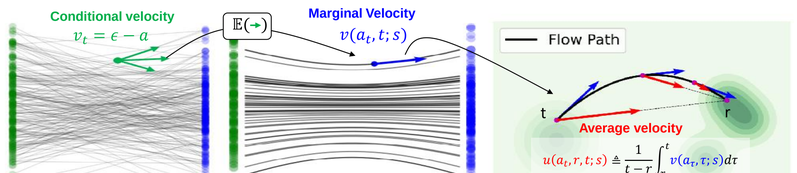


Distillation Frameworks
Still need flow/Diffusion policy and further require one-step policy



OFQL address two core challenges: (1) diffusion/flow-based policies rely on multi-step denoising, making training and inference slow and fragile; (2) existing one-step approaches depend on auxiliary modules or distillation, sacrificing simplicity or performance.

Methodology



Flow Matching policies essentially learn a **marginal velocity** from **conditional velocities**, leading to **curved trajectories** that require multiple timesteps to generate a single action.

OFQL instead models the average velocity, enabling a direct one-step jump to the target action.

Use the MeanFlow identity for a tractable average velocity.

$$u(a_t, r, t; s) = v(a_t, t; s) - (t - r)\frac{d}{dt}u(a_t, r, t; s).$$
$$\frac{d}{dt}u(a_t, r, t; s) = v(a_t, t; s) \cdot \partial_{a_t} u + \partial_t u.$$

OFQL Policy is modeled as follows:

$$a = T_\theta(\epsilon, s) = \epsilon - u_\theta(\epsilon, r{=}0, t{=}1; s), \qquad \epsilon \sim \mathcal{N}(0, I)$$
$$\pi_\theta = (T_\theta)_{\#}\mathcal{N}(0, I)$$

Corresponding behavior cloning loss

$$\mathcal{L}_{FBC}(\theta) = \mathbb{E}_{t,r,r\le t,(a,s)\sim\mathcal{D},\epsilon}\left\|u_\theta(a_t, r, t; s) - \mathrm{sg}(u_{\mathrm{tgt}})\right\|_2^2$$

where
$$v_t = \epsilon - a$$
$$u_{\mathrm{tgt}} = v_t - (t - r)\left(v_t \cdot \partial_z u_\theta + \partial_t u_\theta\right)$$

We learn an average velocity field within the Flow Matching framework to address the limitations of diffusion paths, which are typically stochastic, noisy, and highly curved—making accurate one-step denoising difficult. In contrast, Flow Matching maps noise to data through smoother, more direct trajectories, enabling stable and more denosing effective action generation.

OFQL adopts a behavior-regularized actor--critic framework, minimizing the following objective:

$$\mathcal{L}_Q(\phi) = \mathbb{E}_{s,a,r,s'\sim\mathcal{D},a'\sim\pi_\theta}\left[\left(Q_\phi(s,a) - r - \gamma Q_{\bar{\phi}}(s',a')\right)^2\right] \text{(Eq.6)}$$
$$\mathcal{L}_\pi(\theta) = \mathbb{E}_{s,a\sim\mathcal{D},a^\pi\sim\pi_\theta}\underbrace{\left[-\alpha Q_\phi(s,a^\pi)\right]}_{\text{Q loss}} - \underbrace{\mathcal{L}_{FBC}}_{\text{BC loss}} \quad \text{(Eq.7)}$$

All action sampling during both training and inference is performed in a single step.

$$a \sim \pi_\theta(\cdot \mid s)$$
$$a = \epsilon - u_\theta(\epsilon, r = 0, t = 1; s), \quad \text{where} \quad \epsilon \sim \mathcal{N}(0, I)$$

OFQL features a simple implementation

Algorithm 1 OFQL Algorithm

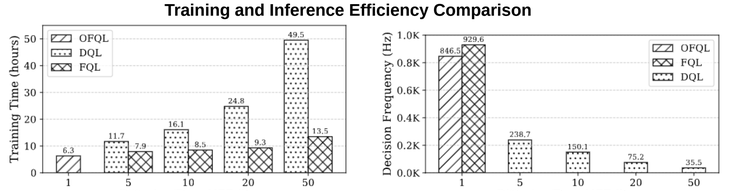
Initialize policy network π_θ , $\pi_{\theta'}$, critic networks Q_{ϕ_1} and Q_{ϕ_2}, $Q_{\phi_1'}$, $Q_{\phi_2'}$
for each iteration **do**
  Sample transition mini-batch $\mathcal{B} = \{(s_h, a_h, r_h, s_{h+1})\} \sim \mathcal{D}$
  # Q-value function learning
  Sample $a_{h+1} \sim \pi_{\theta'}(a_{h+1} \mid s_{h+1})$ by Eq. 16
  Update Q_{ϕ_1} and Q_{ϕ_2} using Eq. 6 {Max-Q backup (Kumar et al., 2020) optional}
  # Policy learning
  Sample $a_h \sim \pi_\theta(a_h \mid s_h)$ by Eq. 16
  Update policy π_θ by minimizing Eq. 7
  # Update target networks every K iteration
  $\theta' \leftarrow \rho\theta' + (1-\rho)\theta$
  $\phi_i' \leftarrow \rho\phi_i' + (1-\rho)\phi_i$ for $i = \{1, 2\}$
end for

OFQL adopts a behavior-regularized actor–critic framework, which proves more effective than alternatives such as IQL and standard policy gradient methods. OFQL **integrates the behavior cloning (BC) loss to guide the learning of an averaged velocity field**, ensuring stable and accurate one-step action generation. **All actions are sampled in a single step** during both training and inference, eliminating multi-step denoising and reducing computational overhead. Despite its simplicity in implementation, OFQL achieves both high efficiency and strong performance.

Experiment

Dataset	Non-Diffusion Policies			Diffusion Planners		Multi-step Diffusion Policies			One-step Flow Policies	
	BC	TD3-BC	IQL	Diffuser	DD	EDP	IDQL	DQL	FQL	OFQL (Ours)
Average (MuJoCo)	51.9	75.3	77.0	78.2	83.2	82.4	84.1	87.9	79.2	**92.5**
Average (AntMaze)	0.2	3.5	57.1	13.3	3.0	50.2	59.8	64.6	79.0	**84.6**
Average (Kitchen)	44.8	0.0	48.7	54.1	65.8	45.5	66.6	61.6	53.1	**67.0**

Training and Inference Efficiency Comparison



OFQL achieves state-of-the-art performance across diverse D4RL domains—including locomotion, navigation, and robot manipulation—consistently outperforming diffusion-based and one-step baselines while maintaining high efficiency via single-step action generation.

Ablations

Comparison of Strategies Toward One-Step Prediction

Method (Steps)	DQL (5)	DQL+DDIM (1)	FBRAC (1)	FQL (1)	OFQL (1)
Score	87.9	11.6 (-76.3)	67.1 (-20.8)	79.2 (-8.7)	**92.6** (+4.7)

Comparison of time sampling distributions

Time-Sampling	Uniform	Logit-Normal
Medium-Expert	94.5±0.5	**95.2±0.4**
Medium	61.1±0.1	**63.8±0.1**
Medium-Replay	**51.7±0.2**	51.2±0.1

Comparison of flow ratio selection

Flow Ratio	1	0.75	0.5	0.25	0
Medium Expert	38.3	90.86	**95.2**	92.03	90.47
Medium	46.3	62.03	**63.8**	63.76	63.2
Medium Replay	45.2	50.2	**51.2**	50.3	10.5

Comparison of the expressiveness of Flow Matching and OFQL