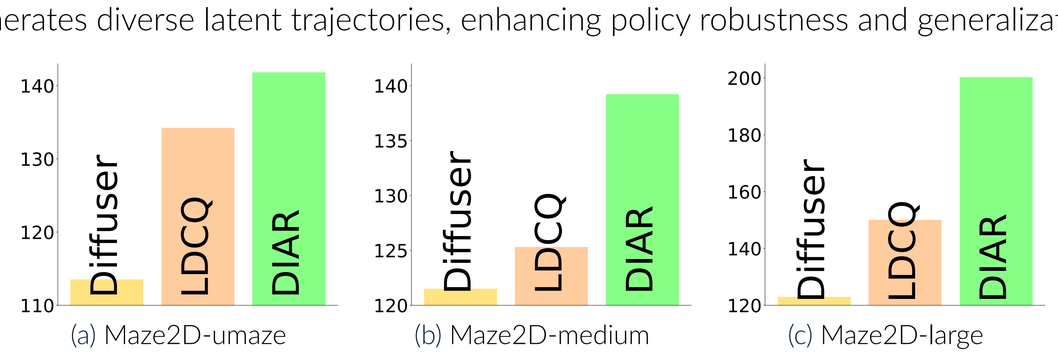
DIAR: Diffusion-model-guided Implicit Q-learning with Adaptive Revaluation

Jaehyun Park[1] Yunho Kim[1] Sejin Kim[1] Byung-Jun Lee[2] Sundong Kim[1]

[1]Gwangju Institute of Science and Technology [2]Korea University

Abstract

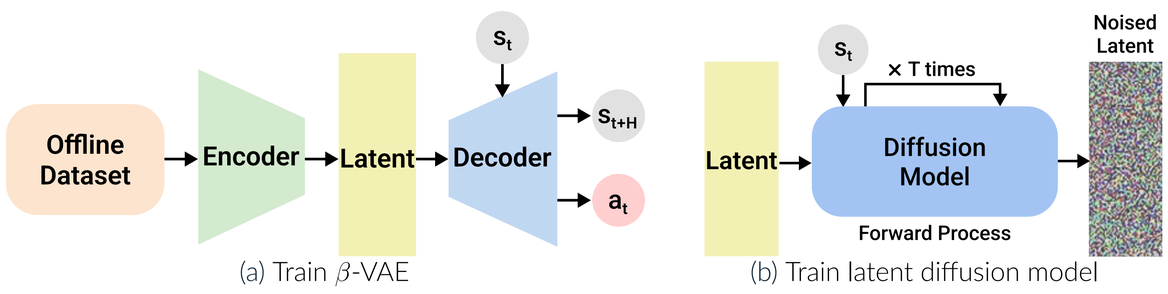
We propose a novel offline reinforcement learning (offline RL) approach, introducing the Diffusion-model-guided Implicit Q-learning with Adaptive Revaluation (DIAR) framework. We leverage diffusion models to learn state-action sequence distributions and incorporate value functions for more balanced and adaptive decision-making. DIAR introduces an Adaptive Revaluation mechanism that dynamically adjusts decision lengths by comparing current and future state values, enabling flexible long-term decision-making. Furthermore, we address Q-value overestimation by combining Q-network learning with a value function guided by a diffusion model. The diffusion model generates diverse latent trajectories, enhancing policy robustness and generalization.



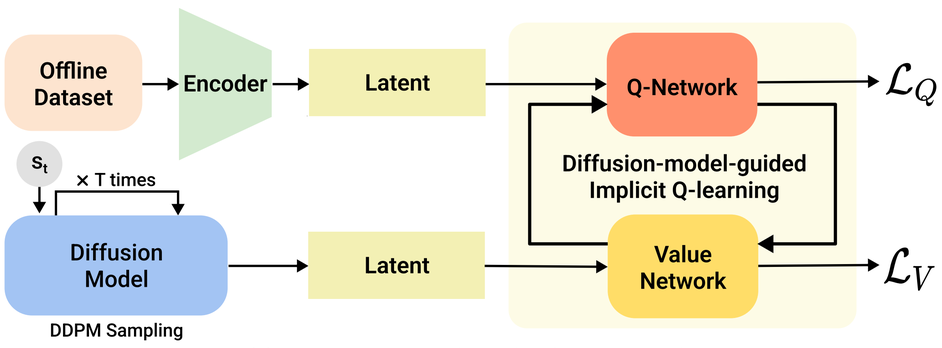
(a) Maze2D-umaze (b) Maze2D-medium (c) Maze2D-large

Figure 1. Performance comparison across D4RL environments with long-horizon and sparse-reward tasks, specifically Maze2D. Our method, DIAR, consistently outperforms other diffusion-based planning frameworks, including Diffuser and LDCQ.

DIAR Training Process



(a) Train β-VAE (b) Train latent diffusion model



(c) Train Q-network and value-network

Figure 2. Three training stages of DIAR. (a) The β-VAE is trained by encoding a state-action sequence spanning an H-length horizon into a latent space, followed by a policy decoder that outputs actions based on the encoded latent z and the state s_t contained within it. (b) A diffusion model is trained using the encoded latent and the initial state s_t. (c) The Q-network is trained on the offline dataset, while the value network is trained on data generated by the diffusion model. This interplay allows the value function and Q-function to guide each other, enabling more balanced learning across both offline samples and generated data.

DIAR Policy Execution

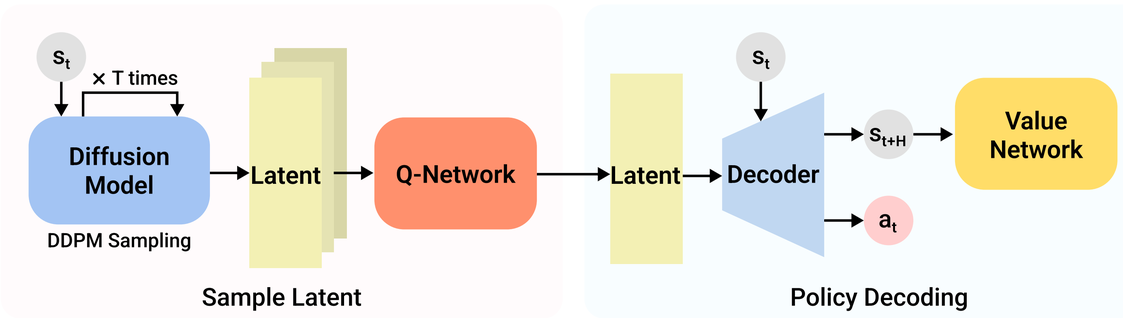
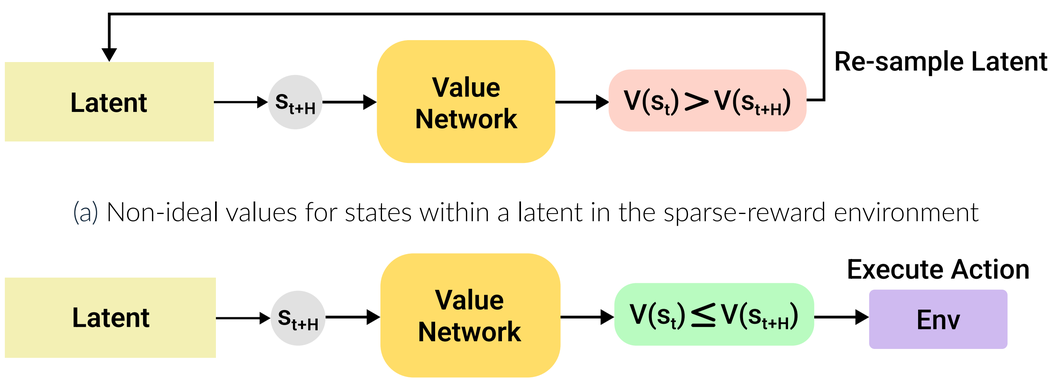


Figure 3. Inference step with DIAR. The current state s_t is put into the diffusion model to extract candidate latent vectors. Then, the latent vector z_t with the highest $Q(s_t, z_t)$ is selected as the best latent vector. This latent vector z_t is subsequently decoded to generate the action a_t. Additionally, the future state s_{t+H} is also decoded to be used for calculating the future value $V(s_{t+H})$.



(a) Non-ideal values for states within a latent in the sparse-reward environment

(b) Ideal values for states within a latent in the sparse-reward environment

Figure 4. The process of finding a better trajectory using Adaptive Revaluation. The process involves making a decision and taking action based on the skill latent z_t with the highest $Q(s_t, z_t)$. The current latent vector z_t is used to predict the future state s_{t+H}, based on which the value $V(s_{t+H})$ of the future state s_{t+H} is calculated. (a) If the value $V(s_t)$ of the current state s_t is greater than the value $V(s_{t+H})$ of the future state s_{t+H}, it is considered non-ideal, and re-sampling is performed. (b) If the value $V(s_{t+H})$ of the future state s_{t+H} is greater than or equal to the value $V(s_t)$ of the current state s_t, it is considered ideal, and the action a_t decoded by the latent vector z_t is executed continuously.

Performance of Offline RL Benchmarks

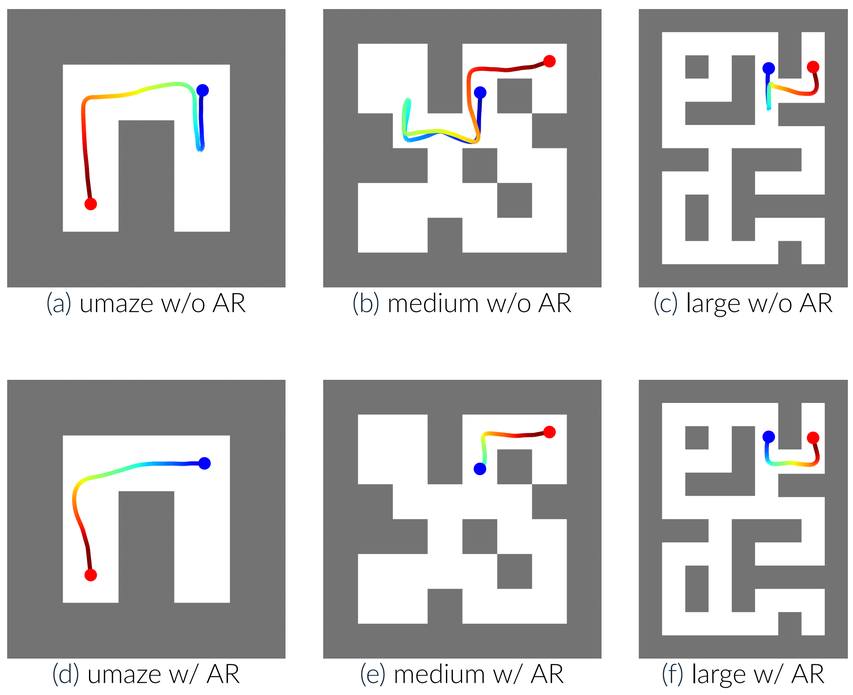
We focus on goal-based tasks in environments with long-horizons and sparse rewards. For offline RL, we use the Maze2D, AntMaze, and Kitchen datasets to test the strengths of our model in long-horizon sparse reward settings To evaluate our model, we compare it against various state-of-the-art models.

Table 1. Comparison with other methods in long horizon sparse reward D4RL environments.

Dataset	BC	IQL	DT	IDQL	Diffuser	DD	LDCQ	DIAR
maze2d-umaze-v1	3.8	47.4	27.3	57.9	113.5	-	134.2	141.8±4.3
maze2d-medium-v1	30.3	34.9	32.1	89.5	121.5	-	125.3	139.2±3.5
maze2d-large-v1	5.0	58.6	18.1	90.1	123.0	-	150.1	200.3±3.4
antmaze-umaze-diverse-v2	45.6	62.2	54.0	62.0	-	-	81.4	88.8±1.5
antmaze-medium-diverse-v2	0.0	70.0	0.0	83.5	45.5	24.6	68.9	68.2±6.7
antmaze-large-diverse-v2	0.0	47.5	0.0	56.4	22.0	7.5	57.7	60.6±2.4
kitchen-complete-v0	65.0	62.5	-	-	-	-	62.5	68.8±2.1
kitchen-partial-v0	38.0	46.3	42.0	-	-	57.0	67.8	63.3±0.9
kitchen-mixed-v0	51.5	51.0	50.7	-	-	65.0	62.3	60.8±1.4

Impact of Adaptive Revaluation

In this section, we analyze the impact of Adaptive Revaluation. We directly compare the cases where Adaptive Revaluation is used and not used in our model. The test is conducted on long-horizon sparse reward tasks, where rewards are sparse. For overall training, an expectile value of $\tau = 0.9$ was used, with $H = 30$ for Maze2D and $H = 20$ for AntMaze and Kitchen.



(a) umaze w/o AR (b) medium w/o AR (c) large w/o AR

(d) umaze w/ AR (e) medium w/ AR (f) large w/ AR

Figure 5. (a)∼(c) Three Maze2D results that only the Q-function is used without Adaptive Revaluation. (d)∼(f) Three Maze2D results for improved decision making using Adaptive Revaluation. Even without Adaptive Revaluation, our model performs well, but we can observe that using Adaptive Revaluation enables more efficient decision-making.

Table 2. Comparison of performance changes with Adaptive Revaluation (AR) in D4RL tasks.

Dataset	DIAR w/o AR	DIAR w/ AR
maze2d-umaze-v1	135.6±2.8	141.8±4.3
maze2d-medium-v1	138.2±3.1	139.2±3.5
maze2d-large-v1	193.5±4.7	200.3±3.4
antmaze-umaze-diverse-v2	88.8±1.5	85.4±2.6
antmaze-medium-diverse-v2	68.2±6.7	67.4±3.4
antmaze-large-diverse-v2	56.0±4.6	60.6±2.4
kitchen-complete-v0	68.8±2.1	63.8±3.0
kitchen-partial-v0	63.3±0.9	63.0±2.5
kitchen-mixed-v0	60.0±0.7	60.8±1.4

Comparison with Skill Latent Models

We further compare our model with other reinforcement learning methods that use skill latents. For the D4RL tasks, we selected methods that use generative models to learn skills and make decisions based on them. As performance baselines, we chose the VAE-based methods OPAL and PLAS, as well as Flow2Control, which utilizes normalizing flows. The performance comparison is shown in Table 3.

Table 3. Performance comparison with other skill latent learning methods in D4RL tasks.

Dataset	BC	PLAS	IQL+OPAL	Flow2Control	DIAR
maze2d-umaze-v1	3.8	57.0	-	-	141.8±4.3
maze2d-medium-v1	30.3	36.5	-	-	139.2±3.5
maze2d-large-v1	5.0	122.7	-	-	200.3±3.4
antmaze-umaze-diverse-v2	45.6	45.3	70.2	81.6	88.8±1.5
antmaze-medium-diverse-v2	0.0	0.7	42.8	83.7	68.2±6.6
antmaze-large-diverse-v2	0.0	0.0	52.4	52.8	60.6±2.4
kitchen-complete-v0	65.0	34.8	11.5	75.0	68.8±2.1
kitchen-partial-v0	38.0	43.9	72.5	74.9	63.3±0.9
kitchen-mixed-v0	51.5	40.8	65.7	69.2	60.8±1.4

Conclusions

- Introduced Diffusion-model-guided Implicit Q-learning with Adaptive Revaluation to enhance abstraction and decision-making in offline RL, excelling in long-horizon sparse reward tasks.
- Developed an algorithm for long-horizon predictions and flexible decision revision, enabling discovery of optimal solutions.
- Achieved leading performance in challenging tasks like Maze2D, AntMaze, and Kitchen, with robustness to hyper-parameter tuning and promising applications in fields like robotics.

jaehyun00518@gmail.com