Towards Adapting Contrastive RL to the Offline Setting

Anonymous Authors[1]

Abstract

Successor measures capture long-horizon, forward-in-time state occupancy statistics for a given policy. Prior RL and neuroscience work has identified the successor measure as a sufficient statistic for estimating value functions for arbitrary rewards, making this measure an important mechanism for offline-to-online adaptation. However, modern RL methods like Contrastive RL (CRL) based off of estimating and using successor features often violate these on-policy assumptions. In this work, we identify a failure mode in offline-to-online adaptation as a result of training successor features over the mixed policy buffers. We present didactic tabular results and results in continuous, high-dimensional settings reflecting the same failure mode, partially explaining past empirical observations that vanilla CRL cannot scale in the offline setting. These results makes progress towards bridging the gap between scalable CRL methods and developing offline-to-online adaptation methods based on the successor measure.

1. Introduction

Many works in RL and neuroscience literature show empirical and theoretical evidence that intelligent agents maintain representations of the world that capture temporal relationships between states and actions (Dayan, 1993; Eysenbach et al., 2023; Barreto et al., 2018; Masset et al., 2025). Correspondingly, recent state-of-the-art RL methods learn world models to do adaptation to downstream tasks (Hafner et al., 2024). A natural long-horizon generalization of the world model is the successor measure, which captures the discounted future state occupancies of a given policy. Importantly, the successor measure is a sufficient statistic for Q-value estimation (Dayan, 1993). Thus, estimating this quantity gives a compelling route to enable scalable offline-to-online adaptation. Indeed, state-of-the-art methods learn objects like successor measures to do zero-shot RL (Touati & Ollivier, 2021; Bagatella et al., 2025).

Motivated by the compelling results of these zero-shot methods, we are particularly interested in developing scalable estimators of the successor measure that does not rely on TD-based methods or complex generative models. In this work, we aim to identify and address roadblocks in adapting scalable successor measure estimators to the offline setting (Eysenbach et al., 2023). Contrastive Reinforcement Learning (CRL) is a goal-reaching algorithm that contrastively learns a log ratio form of the discounted state occupancy measure, which is closely related to the successor measure. Importantly, CRL does not require hard-to-optimize TD objectives and has been shown to scale to neural networks 1000 layers deep in the online setting (Wang et al., 2026). However, despite estimating a quantity linked to downstream adaptation, CRL, as originally introduced and currently implemented (Eysenbach et al., 2023), is traditionally an online and on-policy method. In this work, we aim to make headway on adapting CRL to the offline setting by identifying a key failure mode in directly applying CRL to offline settings: learning representations over mixed policy buffers can lead to suboptimal policies. We identify and demonstrate this failure mode in didactic settings, tabular settings, and high-dimensional, continuous settings. In doing so, we make progress on realizing a method to learn successor measures that does not rely on hard-to-scale objectives.

2. Related Work and Background

We begin by discussing and defining the successor measure (Dayan, 1993), then introduce related objects estimated in Contrastive RL literature (Eysenbach et al., 2023). We conclude this section by identifying a tension between a definitional assumption in successor measure literature and practical implementations of Contrastive RL, queuing our hypotheses and experiments.

Successor features and measures for downstream adaptation. For a given policy, the successor measure $M^\pi(s, a, s_+, a_+)$ is the (discounted) future occupancy of

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

state-actions given a current state-action pair (Dayan, 1993; Touati & Ollivier, 2021):

$$M^\pi(s, a, s_+, a_+) \triangleq (1-\gamma)\sum_t \gamma^t p((s_t, a_t)) = (s_+, a_+) \mid (s_0, a_0) = (s, a)).$$

The successor measure captures future state statistics and policy statistics. An important property of the successor measure is that it is a sufficient statistic of the value function for arbitrary reward functions $r(s, a)$:

$$Q_r^\pi(s, a) = \mathbb{E}_{M^\pi(s,a,s_+,a_+)}[r(s_+, a_+)].$$

The overall objective J_r^π for some starting state-action distribution $p_0(s, a) = p_0(s)\pi(a \mid s)$ then takes the form

$$J(r, \pi) = \mathbb{E}_{p_0(s)}\mathbb{E}_{\pi(a|s)}\mathbb{E}_{M^\pi(s,a,s_+,a_+)}[r(s_+, a_+)].$$

Deep learning follow-ups to the original literature parameterize the successor measure as a bilinear function of representations and train the successor measure using TD-learning (Touati & Ollivier, 2021; Touati et al., 2023; Bagatella et al., 2025; Zheng et al., 2025). Linearized representations enables the use of linear regression to do zero-shot adaptation. At test time, given some reward function r, one can linearly regress onto the successor feature space and then choose the π that gives the largest objective estimate when projecting the reward onto the policy-conditioned representation space (Barreto et al., 2018). In challenging RL settings, tracking π for every policy is intractable. More recent methods parameterize π with latents pre-trained in the learned representations. In the downstream offline-to-online adaptation setting, this requires having access to the policies during the offline pre-training phase (Bagatella et al., 2025; Touati & Ollivier, 2021). Other works explcitly set the latent z, referred to as intentions, to future achieved states s_+ (Ghosh et al., 2023).

Discounted state occupancy measures. For some initial state action pair (s, a), the state-action conditioned discounted state occupancy measure (DSOM) takes the form

$$p^\pi(s_+ = s \mid s, a) \triangleq (1-\gamma)\sum_{t=0}^{\infty} \gamma^t p^\pi(s_t = s \mid s, a), \quad (1)$$

where $p^\pi(s_t = s \mid s, a)$ is the discounted probability density that policy π visits state s after t time steps, given the state-action pair (s, a).

Correspondingly, for an initial state s, the discounted state occupancy measure (DSOM) takes the form

$$p^\pi(s_+ = s \mid s) \triangleq (1-\gamma)\sum_{t=0}^{\infty} \gamma^t p^\pi(s_t = s \mid s), \quad (2)$$

which marginalizes the state-action conditioned DSOM.

Goal-conditioned contrastive RL. Contrative RL similarly estimates a sufficient statistic for Q-value estimation in goal-reaching tasks (Eysenbach et al., 2023; Liu et al., 2024). However, rather than utilize TD learning, which can be brittle, these methods directly invoke CL to learn discounted state occupancy measures without the use of e.g. generative models. Unlike standard TD-based methods, CRL has been shown to scale to neural networks of large depths in the online setting (Wang et al., 2026).

We focus on Contrastive Reinforcement Learning (CRL), a state-of-the-art method that learns to goal-reach in high-dimensional, continuous settings (Eysenbach et al., 2023; Wang et al., 2026; Liu et al., 2024). We introduce the objectives estimated by CRL and sufficient statistics intuition, and include details on the CRL loss and optimization in the appendix.

As introduced, CRL utilizes contrastive learning to fit the log probability ratio

$$f_\theta(s, a, s_+) = \psi(s_+)^T \phi(s, a) \approx \log \frac{p^\pi(s_+ \mid s, a)}{p^\pi(s)}, \quad (3)$$

then learns a goal conditioned policy $\pi(a \mid s, g)$ by maximizing

$$\mathbb{E}_{p^\pi(s,g)} f_\theta(s, \pi(\cdot \mid s, g), g) \quad (4)$$

to an on-policy buffer. Notably, the object $p^\pi(s_+ \mid s, a)$ bears close similarity to the successor measure object, and similarly captures long-horizon relationships between state-actions and other states.

However, while originally proposed as an on-policy method with provable policy improvement in the goal reaching setting (Eysenbach et al., 2023), practical implementations of CRL generally fits $f_\theta(s, a, s_+)$ to off-policy buffers (Wang et al., 2026; Liu et al., 2024). Thus, rather than capturing an object like the successor measure, CRL estimates a log ratio of objects defined over an off-policy buffer $\mathcal{B}$:

$$f_\theta(s, a, s_+) = \log \frac{p_{\mathcal{B}}(s_+ \mid s, a)}{p_{\mathcal{B}}(s)}. \quad (5)$$

This discrepancy between the assumptions and theoretical properties of the successor measure and the practical implementation of CRL forms a tension when adapting CRL to the offline setting, which we characterize in this work.

Contrastive RL in the offline setting. While prior work introduced CRL as an on-policy, online method, modifications to CRL can perform well in offline settings. For example, Park et al. (2025) considers a variant of CRL that augments the typical CRL loss with a goal-conditioned behavioral cloning (GCBC) term (Ghosh et al., 2023), regressing towards actions that *precede* desired goals.

In this work, we will argue that adding on such a regression target partially but does not fully resolve the tension between the on-policy assumptions of the successor measure and the off-policy nature of CRL.

3. Experiments and Results

To summarize, CRL is an effective, scalable method for goal-reaching that estimates a log ratio form of the discounted state occupancy measure of a policy, an object closely related to the successor measure. Like the successor measure, this occupancy measure is a sufficient statistic for value function estimation. However, in the offline setting, CRL learns the log ratio over a buffer rather than an individual policy, losing sufficiency. In this section, we characterize the failure modes when CRL fits representations over a buffer collected by multiple policies, making progress on adapting CRL as a method for offline-to-online transfer via the successor measure.

3.1. Fitting Successor Measures over Mixed Buffers Leads to Pathological Failure Modes

At a high level, the failure mode amounts to a collapsed representation structure that comes from averaging statistics over multiple, possibly optimal, policies. Thus, even if a buffer contains optimal trajectories for certain goal-reaching tasks, state occupancies and successor measures learned over this buffer will *fail* to recover the underlying optimal behavior. Indeed, our experiments show that increasing the number of underlying optimal policies in a dataset may even lead to a net *decrease* in the ability of a CRL policy to downstream generalize due to the aliasing of occupancy statistics.

3.1.1. Didactic experiment: graph and 4-rooms

Didactic Node Example We start with didactic, tabular examples. Consider the MDP (shown in Figure 1) that consists of two goals, two optimal paths that route through a shared highway, and two suboptimal paths to these goals. Then, consider the following offline dataset that consists of an equal number of trajectories gathered by two policies, respectively optimal for reaching left and right side goals of the MDP (Figure 1, right). Furthermore, assume the dataset contains trajectories along a sub-optimal but consistent route. Assume that the offline buffer is an equal mix of optimal and non-optimal trajectories.

Then, fitting the classifier

$$f_\theta(s, a, s_+) = \log \frac{p_\mathcal{B}(s_+ \mid s, a)}{p_\mathcal{B}(s)} \tag{6}$$

and optimizing the goal-reaching policy

$$\mathbb{E}_{p^\pi(s,g)} f_\theta(s, \pi(\cdot \mid s, g), g) \tag{7}$$

leads to a representation collapse that *penalizes* the optimal, shared highway route (Figure 1, right).

Commanding either goal causes the mixed buffer probability for reaching the specified goal to be at most $\frac{1}{2}$ from the middle node. Meanwhile, moving to the left (or right) side reaches the specified goal with probability greater than $\frac{1}{2}$. Therefore, the CRL agent prefers taking the sub-optimal trajectories in the right environment. By mixing the buffer, the middle node is similar to a coin-flip between the desired goal and another state. The agent then prefers sub-optimal trajectories that have a high probability of success given the offline buffer, which penalizes the shared optimal state and action and realizes the representation collapse.

Four Rooms Maze This collapse also appears in larger tabular settings. We run tabular experiments in the Four Rooms Maze and generate an optimal buffer to reach randomly-sampled goals in the maze. Each trajectory in the buffer is essentially collected by a different policy that terminates in a unique goal state (Figure 2). The mixed-policy CRL representations are fit over the buffer.

As a point of comparison, we also fit properly on-policy CRL representations on each individual trajectory in the buffer. For each goal, we use the maximum success rate among all the trained goal-reaching agents. This represents the attained success rate if we were to use all of the policies that collected the optimal trajectories to separately do goal-reaching. Figure 4 clearly shows the same failure mode as the didactic node environments: as the number of optimal policies increases, the downstream goal reaching ability of a single CRL agent trained over the mixed buffer decreases relative to the goal reaching abilities of the combined agents. Figure 3 visualizes this collapse, where the success rate drops sharply between the on-policy representations and the combined buffer representations. Both methods use the buffer illustrated in Figure 2.

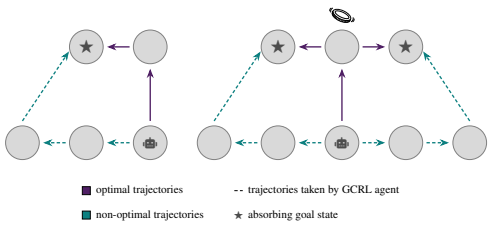

Figure 1. **The five-state and eight-state node environments.** The agent starts from the node labeled by the robot. States marked with a star are absorbing goal states that the agent is commanded to reach.

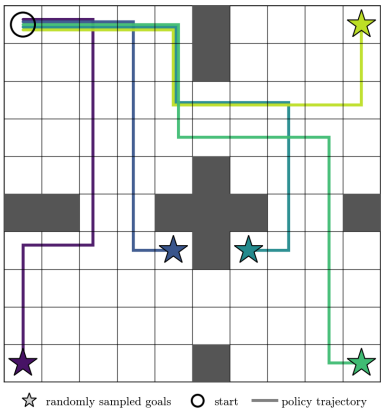

Figure 2. **Four rooms trajectory policies.** One sampled mixed policy buffer that was used to train CRL representations for the downstream goal-reaching task.

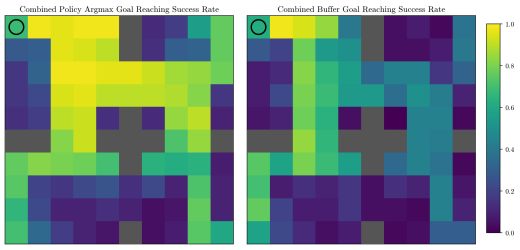

Figure 3. **Average success rates across five seeds for all goals in the Four Rooms Maze environment.** The heatmap on the left shows per-goal argmax of success rates from each CRL agent trained under one trajectory from the sampled mixed buffer while the heatmap on the right shows the success rates from one CRL agent trained under the entire mixed buffer. The agent always starts in the cell marked by a black circle.

3.1.2. Continuous example: highways and spokes in AntMaze

Finally, we consider this failure mode within a high-dimensional, continuous AntMaze setting (Figure 6) with 42 dimensional observations and 8 degrees of freedom (Park et al., 2025). All the following results are over 5 seeds, with further experiment details in the appendix.

Given a dataset of expert, goal-reaching trajectories within the AntMaze, we implement vanilla CRL following (Eysenbach et al., 2023) and evaluate on goals. Like in the didactic node setting, we construct an MDP that consists of a highway (Figure 7) where trajectories near-optimally reach goals that branch off from a shared highway. In addition to the highway setting, we also consider a spokes setting (Figure 8) where the optimal trajectories do not overlap.

Figure 5 shows that vanilla CRL collapses in the highway limit, and fails to significantly goal reach. Meanwhile, vanilla CRL can successfully goal reach when there is no overlap between the optimal trajectories in the spokes setting.

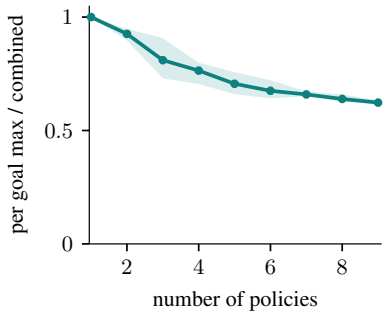

Figure 4. **Ratio between per-goal max success rate and combined buffer success rate.** The plot shows the ratio between the average success rate over all goals using per-goal argmax of success rates from each CRL agent trained under one trajectory from the sampled mixed buffer compared to using one CRL agent trained under the entire mixed buffer. Average success rate was then averaged across five seeds.

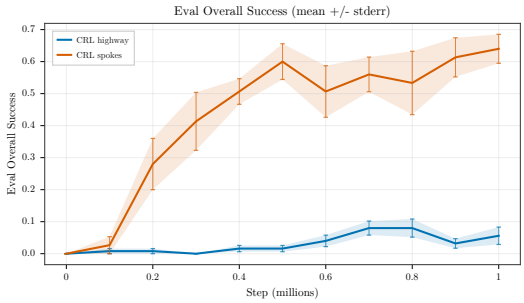

Figure 5. **Ant successfully goal reaches in the highway setting and buffer, but not the spokes setting.** Success rate over 5 seeds, 1M CRL training steps, and a buffer of 1M environment steps composed of trajectories noised around Figure 7 and Figure 8.

4. Discussion

In this paper, we identify a discrepancy between modern CRL methods and the theoretical assumptions in successor measure literature, and show that this discrepancy leads to a failure mode in simple node environments as well as tabular and continuous settings. In doing so, we make progress on current work adapting CRL to the offline setting, in order to learn successor measures for downstream adaptation with simple objectives.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Bagatella, M., Pirotta, M., Touati, A., Lazaric, A., and Tirinzoni, A. Td-jepa: Latent-predictive representations for zero-shot reinforcement learning, 2025. URL https://arxiv.org/abs/2510.00739.

Barreto, A., Dabney, W., Munos, R., Hunt, J. J., Schaul, T., van Hasselt, H., and Silver, D. Successor features for transfer in reinforcement learning, 2018. URL https://arxiv.org/abs/1606.05312.

Dayan, P. Improving generalization for temporal difference learning: The successor representation. *Neural computation*, 5(4):613–624, 1993.

Eysenbach, B., Salakhutdinov, R., and Levine, S. C-learning: Learning to achieve goals via recursive classification. *CoRR*, abs/2011.08909, 2020. URL https://arxiv.org/abs/2011.08909.

Eysenbach, B., Zhang, T., Salakhutdinov, R., and Levine, S. Contrastive learning as goal-conditioned reinforcement learning, 2023. URL https://arxiv.org/abs/2206.07568.

Ghosh, D., Bhateja, C., and Levine, S. Reinforcement learning from passive data via latent intentions, 2023. URL https://arxiv.org/abs/2304.04782.

Hafner, D., Pasukonis, J., Ba, J., and Lillicrap, T. Mastering diverse domains through world models, 2024. URL https://arxiv.org/abs/2301.04104.

Liu, G., Tang, M., and Eysenbach, B. A single goal is all you need: Skills and exploration emerge from contrastive rl without rewards, demonstrations, or subgoals, 2024. URL https://arxiv.org/abs/2408.05804.

Masset, P., Tano, P., Kim, H. R., Malik, A. N., Pouget, A., and Uchida, N. Multi-timescale reinforcement learning in the brain. *Nature*, 642(8068):682–690, 2025.

Mohamed, F., Ji, C., Eysenbach, B., and Berseth, G. Temporal representations for exploration: Learning complex exploratory behavior without extrinsic rewards. *arXiv preprint arXiv:2603.02008*, 2026.

Park, S., Frans, K., Eysenbach, B., and Levine, S. Ogbench: Benchmarking offline goal-conditioned rl, 2025. URL https://arxiv.org/abs/2410.20092.

Touati, A. and Ollivier, Y. Learning one representation to optimize all rewards, 2021. URL https://arxiv.org/abs/2103.07945.

Touati, A., Rapin, J., and Ollivier, Y. Does zero-shot reinforcement learning exist?, 2023. URL https://arxiv.org/abs/2209.14935.

van den Oord, A., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *CoRR*, abs/1807.03748, 2018. URL http://arxiv.org/abs/1807.03748.

Wang, K., Javali, I., Bortkiewicz, M., Trzciński, T., and Eysenbach, B. 1000 layer networks for self-supervised rl: Scaling depth can enable new goal-reaching capabilities, 2026. URL https://arxiv.org/abs/2503.14858.

Zheng, C., Tuyls, J., Peng, J., and Eysenbach, B. Can a misl fly? analysis and ingredients for mutual information skill learning, 2025. URL https://arxiv.org/abs/2412.08021.

A. Contrastive RL Loss and Optimization

Contrastive representation learning methods (van den Oord et al., 2018) train a critic function C_θ that takes as input pairs of positive and negative examples, and learn representations so that positive pairs have similar representations and negative pairs have dissimilar representations.

Prior works utilize contrastive learning to learn successor features and measures for control (Eysenbach et al., 2023; Liu et al., 2024; Mohamed et al., 2026).

Let s_t denote a state at time t, a_t denote the action at time t, and s_+ denote a future state. To estimate the log ratio between the probabilities, we sample positives $((s_t, a_t), s_+)$ from a joint distribution $p_{\mathcal{B}}((s_t, a_t), s_+) = p_{\mathcal{B}}(s_t, a_t)p_{\mathcal{B}}(s_+ \mid s_t, a_t)$, while the negative examples are sampled from the product of marginal distributions $p_{\mathcal{B}}(s_t, a_t)p_{\mathcal{B}}(s_+)$. Here, $p_{\mathcal{B}}(s_+)$ is the marginal discounted state occupancy $p_\tau(s_+) = \iint p_\tau(s_+ \mid s_t, a_t)p_{\mathcal{B}}(s_t, a_t)\ ds_t\, da_t$.

We use the InfoNCE loss to train the contrastive learning model (van den Oord et al., 2018). Let $\mathcal{B} = \{(s_t^{(i)}, a_t^{(i)}, s_+^{(i)})\}_{i=1}^K$ be the sampled batch, where $s_+^{(1)}$ is the positive example and $\{s_+^{(2:K)}\}$ are the $K-1$ negatives sampled independently from $(s_t^{(i)}, a_t^{(i)})$. In addition to the standard InfoNCE objective, prior work has shown that a LogSumExp regularizer is necessary for control (Eysenbach et al., 2020). The full contrastive reinforcement learning (CRL) loss $\mathcal{L}_{\text{CL}}(\theta)$ is as follows:

$$-\mathbb{E}_{\substack{(s_t,a_t)\sim p_{\mathcal{B}}(s_t,a_t) \\ s_+^{(1)}\sim p_{\mathcal{B}}(s_+|s_t,a_t) \\ s_+^{(2:K)}\sim p_{\mathcal{B}}(s_+)}}\left[\log\left(\frac{e^{C_\theta((s_t,a_t),s_+^{(1)})/\tau}}{\sum_{j=1}^{K} e^{C_\theta((s_t,a_t),s_+^{(j)})/\tau}}\right)\right] \tag{8}$$

where τ is a temperature parameter. The optimal critic $C^*((s_t, a_t), s_+)$ corresponds to a log probability ratio (**?**), $C^*((s_t, a_t), s_+) \approx \log p_{\mathcal{B}}(s_+ \mid s_t, a_t) - \log p_{\mathcal{B}}(s_+)$, where we use the negative ℓ^1 and ℓ^2 distances as the critic function. Conceptually, the critic C_θ gives a temporal similarity score between state-action pairs (s_t, a_t) and future states s_+ via learned representation ϕ_θ and ψ_θ.

B. Experiment Details

B.1. Didactic Node Environment Experiments

At each node (except the goal states), the agent has five actions: up, down, left, right, and stay. If there is not another node present in the direction that the agent chooses to move, it stays in place. For the goal states, once the agent enters the state it cannot leave.

The mixed policy buffer is built from trajectories that are padded with transitions that start at the terminal state of the given trajectory and take the stay action. Each unique trajectory is repeated in the buffer 200 times. We compute the discounted state occupancy measure M and a truncated low-rank decomposition of M to get ϕ and ψ for downstream goal reaching, sampling actions from the Boltzmann policy $\pi(a|s, g) \propto \exp(\phi(s, a)^\top \psi(g)/\tau$. We run goal conditioned evaluation for each environment 20 times for each goal. Hyperparameters are given in Table 1.

Table 1. Hyperparameters for the node environment.

HYPERPARAMETER	VALUE
TAU	0.3
MAX STEPS	20
BUFFER TRAJ LENGTH	10
GAMMA	0.99
REPRESENTATION DIMENSION	4

B.2. Tabular Environments

The optimal buffer for the environment is created by finding the shortest path to each state in the maze through breadth first search.

To compute the CRL representations ϕ and ψ, we take the truncated low-rank decomposition of the discounted state

Table 2. Hyperparameters for the four rooms maze environment.

HYPERPARAMETER	VALUE
TAU	0.3
MAX STEPS	100
BUFFER TRAJ LENGTH	20
GAMMA	0.99
REPRESENTATION DIMENSION	16

occupancy matrix M. Then, using the current state s and a commanded goal g, the goal-reaching actions are sampled from the Boltzmann policy $\pi(a|s,g) \propto \exp(\phi(s,a)^T\psi(g)/\tau)$, where τ is the temperature hyperparameter. The success rate for the given policy is the mean success rate of the CRL agent for all goals. Hyperparameters are given in Table 3.

B.3. AntMaze Environment Experiments

We include environment figures and hyperparameters for the antmaze experiment (Table 3) and use standard CRL hyperparameters included in (Park et al., 2025), while dropping the added GCBC term in the CRL loss. The trajectory buffer consists of trained SAC expert agents that approximately follow the optimal routes in Figure 7 and Figure 1.

Table 3. Hyperparameters for the AntMaze environment.

HYPERPARAMETER	VALUE
CRL TRAINING STEPS	1M
CRL BATCH SIZE	1024
BUFFER SIZE	1M
EXPERT POLICY	SAC (FOLLOWING (PARK ET AL., 2025))
REPRESENTATION DIMENSION	512
STATE DIMENSION	42
DEGREES OF FREEDOM	8
LEARNING RATE	3E-4
DISCOUNT	0.99
ACTOR AND CRITIC HIDDEN DIMS	(512, 512, 512)

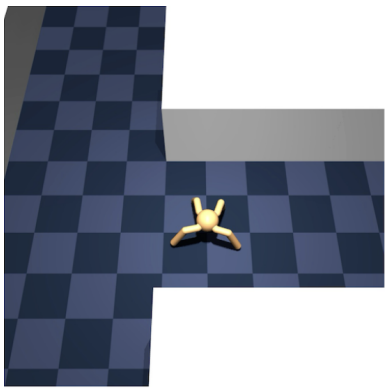

Figure 6. **AntMaze setting.** 42 dimensional state observations and 8 degrees of freedom, continuous environment. Figure and environment from (Park et al., 2025).

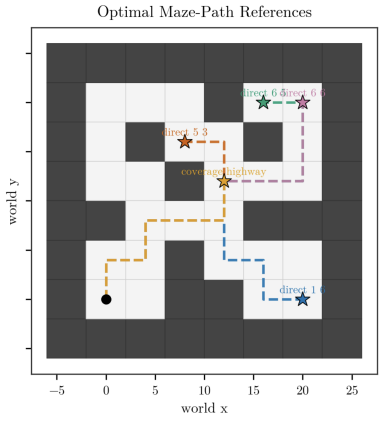

Figure 7. **The highway setting in the AntMaze environment.** Optimal trajectories in the offline buffer and goals overlap.

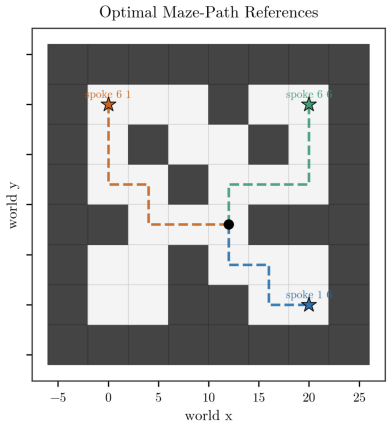

Figure 8. **The spokes setting in the AntMaze environment.** Optimal trajectories in the offline buffer do not overlap.