A Training & Experiments details

A.1 Training details

Baselines. The baseline curriculum RL algorithms are trained as follows,

- OUTPACE [5]: We follow the default setting in the original implementation from `https://github.com/jayLEE0301/outpace_official`.
- HGG [37] : We follow the default setting in the original implementation from `https://github.com/Stilwell-Git/Hindsight-Goal-Generation`.
- CURROT [18]: We follow the default setting in the original implementation from `https://github.com/psclklnk/currot`.
- PLR [16], VDS [49], ALP-GMM [36] : We follow the default setting in implementation from `https://github.com/psclklnk/currot`.

D2C and all the baselines are trained by SAC [13] with the sparse reward except for the OUTPACE which uses an intrinsic reward based on Wasserstein distance with a time-step metric.

Training details. We used NVIDIA A5000 GPU and AMD Ryzen Threadripper 3960X for training, and each experiment took about 1$\sim$2 days for training. We used small noise from a uniform distribution with an environment-specific noise scale (Table 3) for augmenting the conditioned goal in Eq (5). Also, we used the mapping $\phi(\cdot)$ that abstracts the state space into the goal space when we use the diversified conditional classifiers (i.e. $f_i(\phi(s); g)$. For example, $\phi(\cdot)$ abstracts the proprioceptive states (e.g. xy position of the agent) in navigation tasks, and abstracts the object-centric states (e.g. xyz position of the object) in robotic manipulation tasks.

Table 2: Hyperparameters for D2C

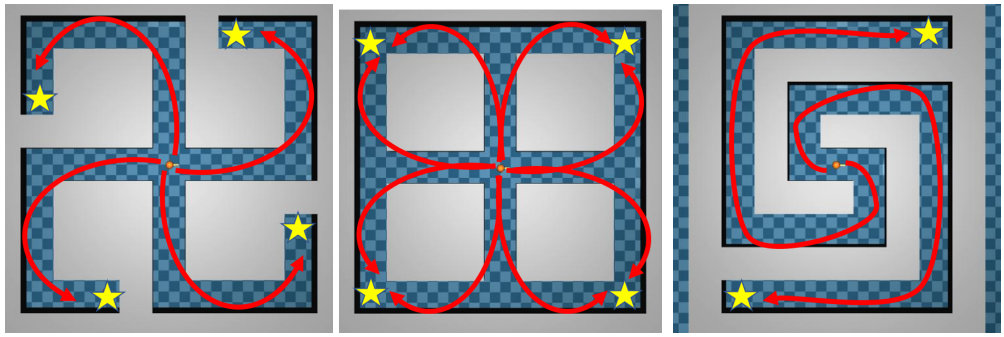
critic hidden dim	512	discount factor γ	0.99
critic hidden depth	3	batch size	512
critic target τ	0.01	init temperature α_{init} of SAC	0.3
critic target update frequency	2	replay buffer $\mathcal{B}$ size	3e6
actor hidden dim	512	learning rate for f_i	1e-3
actor hidden depth	3	learning rate for Critic & Actor	1e-4
actor update frequency	2	optimizer	adam

Table 3: Default env-specific hyperparameters for D2C

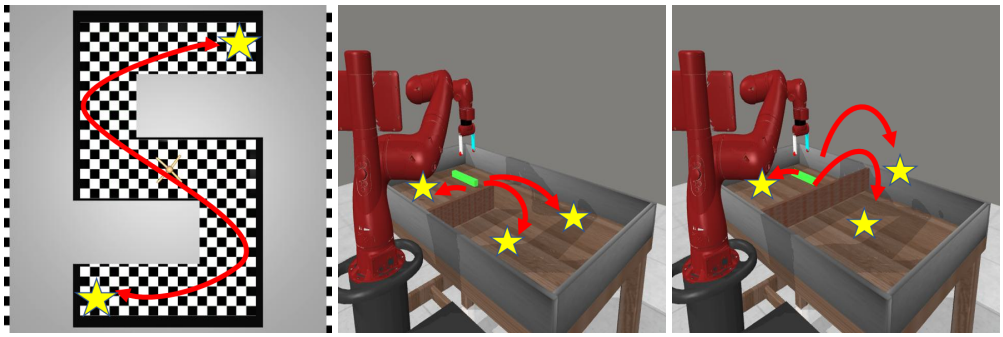
Env name	# of heads	λ	ϵ	f_i update freq (step)	f_i # of iteration per update	max episode horizon
Complex-Maze	2	1	0.5	2000	16	100
Medium-Maze	2	1	0.5	2000	16	100
Spiral-Maze	2	1	0.5	2000	16	100
Ant Locomotion	2	2	1.0	4500	16	300
Sawyer-Peg-Push	2	1	0.025	3000	16	200
Sawyer-Peg-Pick&Place	2	1	0.025	3000	16	200

A.2 Environment details

- Complex-Maze: The observation consists of the xy position, angle, velocity, and angular velocity of the 'point'. The action space consists of the velocity and angular velocity of the 'point'. The initial state of the agent is $[0, 0]$ and the desired outcome states are obtained from the default goal points $[8, 16], [-8, -16], [16, -8], [-16, 8]$. The size of the map is 36×36.
- Medium-Maze: It is the same as the Complex-Maze environment except that the desired outcome states are obtained from the default goal points $[16, 16], [-16, -16], [16, -16], [-16, 16]$.



(a) Complex-Maze (b) Medium-Maze (c) Spiral-Maze



(d) Ant Locomotion (e) Sawyer Push (f) Sawyer Pick&Place

Figure 7: Environments used for evaluation: yellow stars indicate the desired outcome examples. **(a)-(c)** the agent should navigate various maze environments with multi-modal desired outcome distribution. **(d)** the ant locomotion environment with multi-modal desired outcome distribution. **(e)** the robot has to push or pick & place a peg to the multi-modal desired locations while avoiding an obstacle at the center of the table.

- Spiral-Maze: The observation space and actions space and initial state of the agent are the same as in the Complex-Maze environment. The desired outcome states are obtained from the default goal points $[12, 16], [-12, -16]$. The size of the map is 28×36.
- Ant Locomotion: The observation consists of the xyz position, xyz velocity, joint angle, and joint angular velocity of the 'ant'. The action space consists of the torque applied on the rotor of the 'ant'. The initial state of the agent is $[0, 0]$ and the desired outcome states are obtained from the default goal points $[4, 8], [-4, -8]$. The size of the map is 12×20.
- Sawyer-Peg-Push: The observation consists of the xyz position of the end-effector, the object, and the gripper's state. The action space consists of the xyz position of the end-effector and gripper open/close control. The initial state of the object is $[0.4, 0.8, 0.02]$ and the desired outcome states are obtained from the default goal points $[-0.3, 0.4, 0.02], [-0.3, 0.8, 0.02], [0.4, 0.4, 0.02]$. The wall is located at the center of the table. Thus, the robot arm should detour the wall to reach the desired goal states. We referred to the metaworld [48] and EARL [40] environments.
- Sawyer-Peg-Pick&Place: It is the same as the Sawyer-Peg-Push environment except that the desired outcome states are obtained from the default goal points $[-0.3, 0.4, 0.2], [-0.3, 0.8, 0.2], [0.4, 0.4, 0.2]$, and the wall is located at the center of the table, fully blocking a path for pushing. Thus, the robot arm should pick and move the object over the wall to reach the desired goal states.

B Algorithm

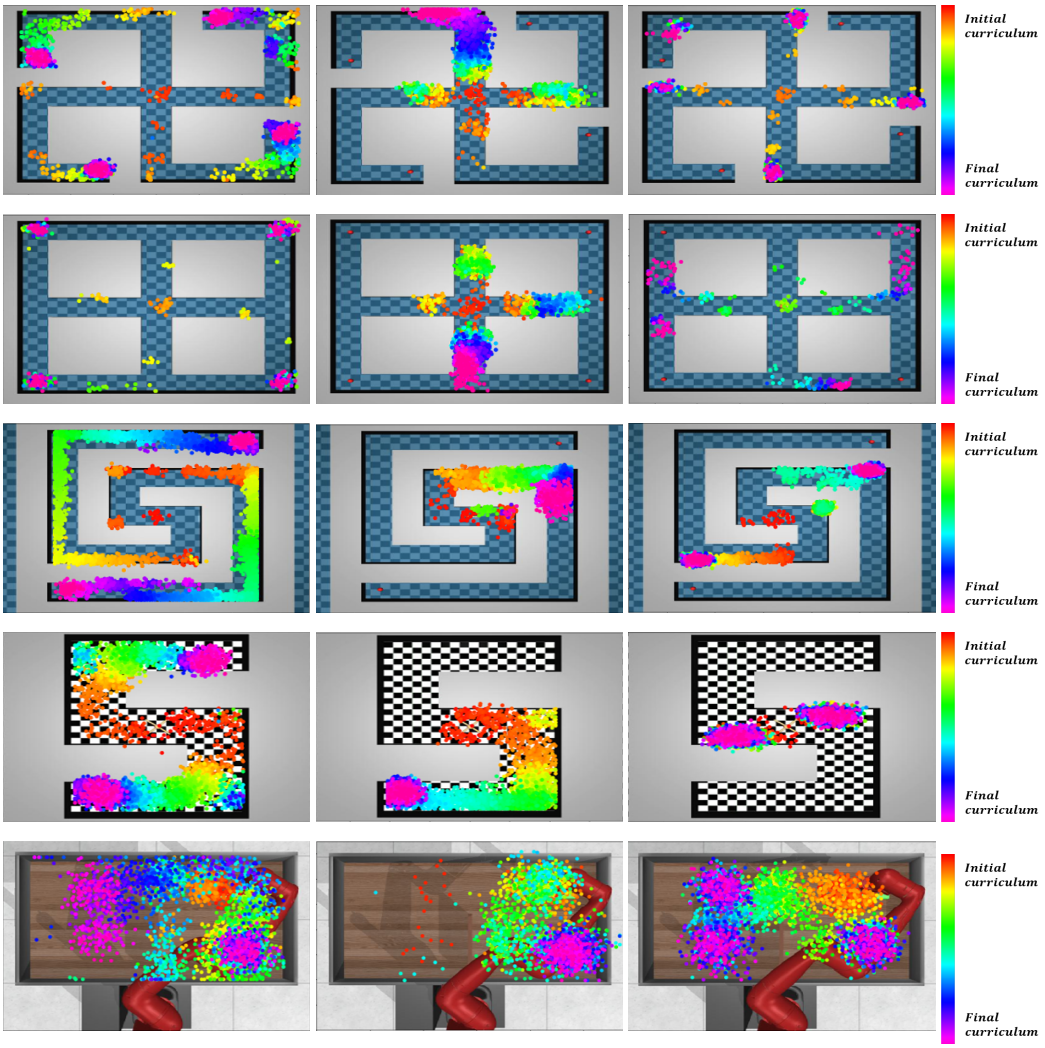
Algorithm 1 Overview of D2C algorithm

Input: desired outcome examples $\hat{p}^+(g)$, total training episodes N, Env, environment horizon H, actor π, critic Q, replay buffer $\mathcal{B}$
for iteration=1,2,...,N **do**
  $\hat{p}^c \leftarrow$ sample K curriculum goals that minimize Eq (6). We refer to HGG [37] for solving the bipartite matching problem.
  for i=1,2,...,K **do**
    `Env.reset()`
    $g \leftarrow \hat{p}^c$
    for t=0,1,...,H-1 **do**
      if achieved g **then**
        $g \leftarrow$ random goal (randomly sample a few states near s_t and measure p_{pseudo}. Then select a state with the highest value of p_{pseudo}.)
      end if
      $a_t \leftarrow \pi(\cdot|s_t, g)$
      $s_{t+1} \leftarrow$ `Env.step`(a_t)
    end for
    $\mathcal{B} \leftarrow \mathcal{B} \cup \{s_0, a_0, g, s_1 ...\}$
  end for
  for i=0,1,...,M **do**
    Sample a minibatch `b` from $\mathcal{B}$ and replace the original reward with intrinsic reward in Section 4.4 (We used relabeling technique based on [2]).
    Train π and Q with `b` via SAC [13].
    Sample another minibatch $\texttt{b}'$ from $\mathcal{D}_{\text{S}} \sim \{(\mathcal{B}, y = 0), (\mathcal{D}_{\text{G}}, y = 1)\}$ and $\mathcal{D}_{\text{T}} \sim \mathcal{U}$.
    Train f_i with $\texttt{b}'$ via Eq. (5)
  end for
end for

C More experimental results

C.1 Full results of the main script

We included the full results of the main script in this section. We include the visualization of the proposed curriculum goals in all environments in Figure 8. The visualization results of the Sawyer-Peg-Pick&Place are not included as it shares the same map with the Sawyer-Peg-Push environment.

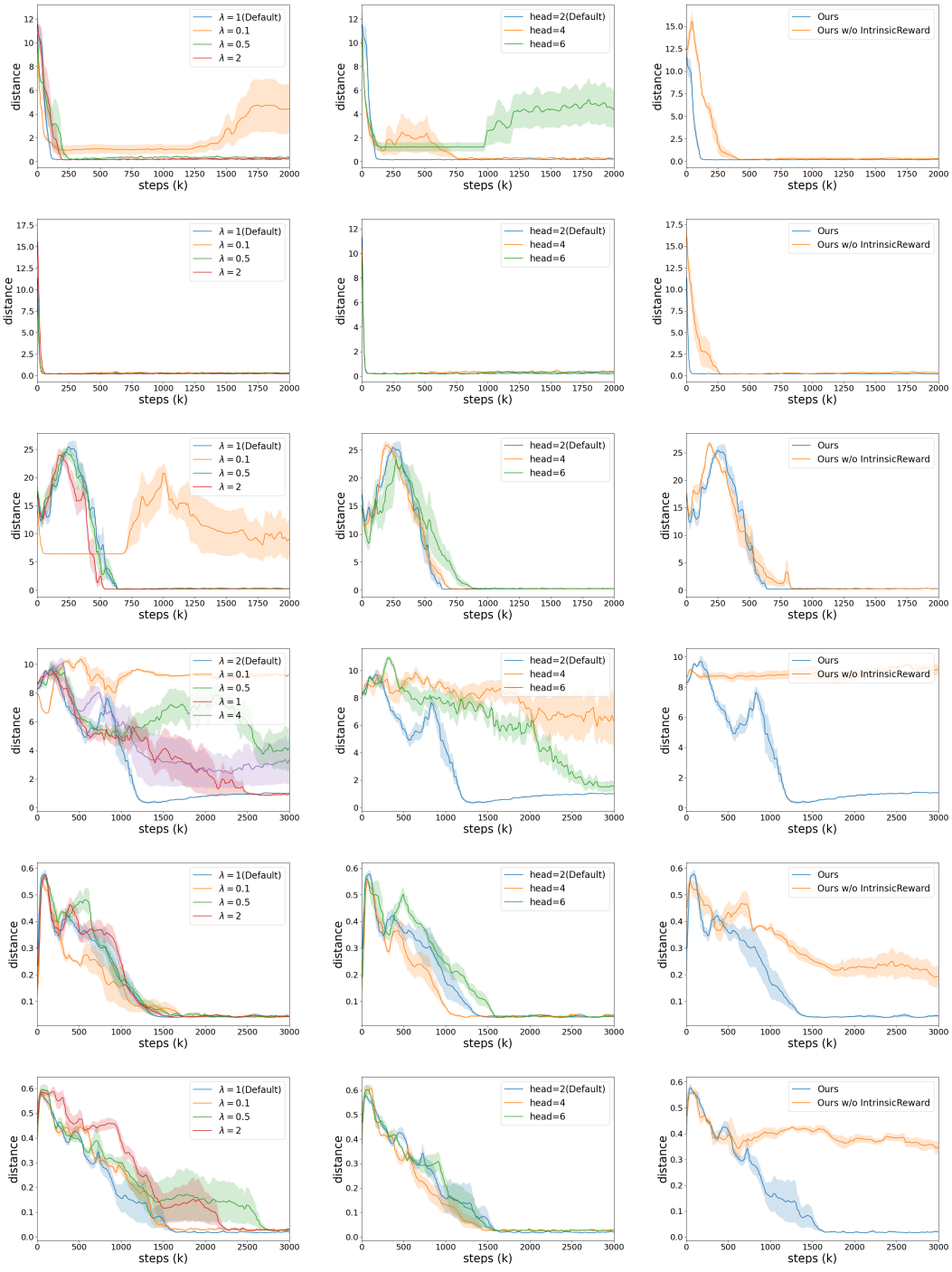


(a) Ours (b) OUTPACE (c) HGG

Figure 8: Curriculum goal visualization of the proposed method and baselines in all environments. **First row**: Complex-Maze. **Second row**: Medium-Maze. **Third row**: Spiral-Maze. **Fourth row**: Ant Locomotion. **Fifth row**: Sawyer Push. **Sixth row**: Sawyer Pick & Place.

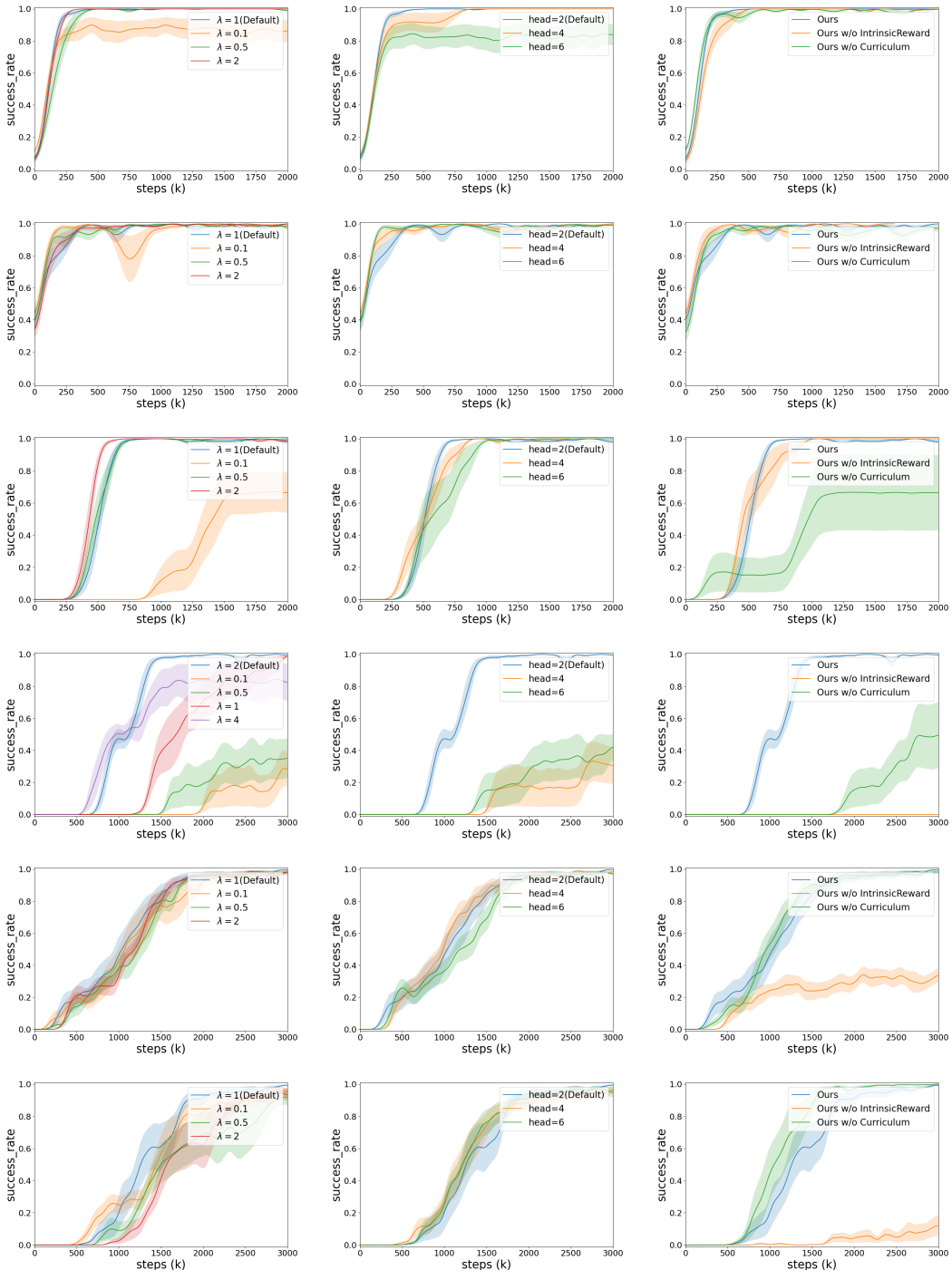
C.2 Additional ablation study results

Full ablation study results of the main script. We conducted ablation studies described in our main script in all environments. Figure 9 shows the average distance from the proposed curriculum goals to the desired final goal states along the training steps, and Figure 10 shows the episode success rates along the training steps. Note that there are no results for the ablation study without a curriculum proposal in Figure 9c (unlike Figure 10c) since there are no curriculum goals to measure the distance from the desired final goal states. As we can see in these figures, we could obtain consistent analysis with the results in the main script in most of the environments.



(a) Auxiliary loss weight λ (b) Number of prediction heads (c) Intrinsic reward

Figure 9: Ablation study in terms of the distance from the proposed curriculum goals to the desired final goal states (**Lower is better**). **First row**: Complex-Maze. **Second row**: Medium-Maze. **Third row**: Spiral-Maze. **Fourth row**: Ant Locomotion. **Fifth row**: Sawyer Push. **Sixth row**: Sawyer Pick & Place. The shaded area represents a standard deviation across 5 seeds.

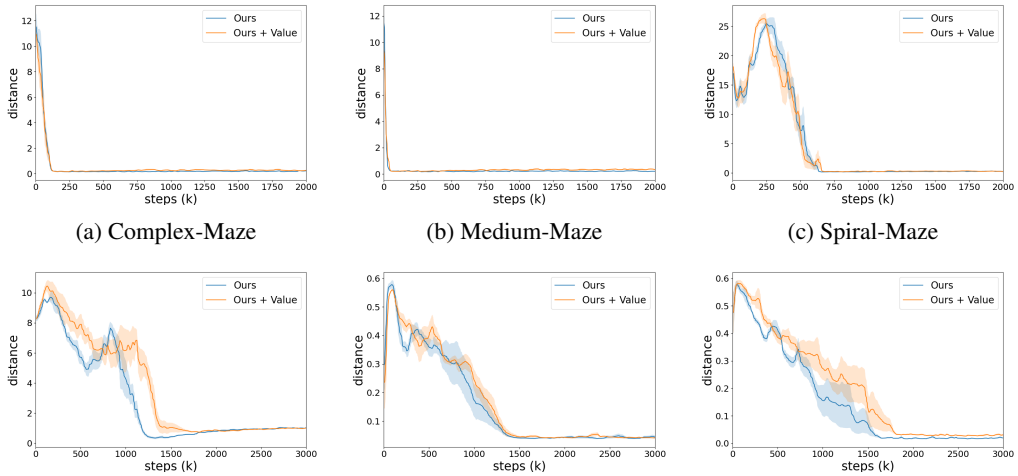


(a) Auxiliary loss weight λ (b) Number of prediction heads (c) Intrinsic reward & Curriculum

Figure 10: Ablation study in terms of the episode success rate. **First row**: Complex-Maze. **Second row**: Medium-Maze. **Third row**: Spiral-Maze. **Fourth row**: Ant Locomotion. **Fifth row**: Sawyer Push. **Sixth row**: Sawyer Pick & Place. The shaded area represents a standard deviation across 5 seeds.

Curriculum learning objective type. We conduct additional experiments to validate whether reflecting the temporal distance in a curriculum learning objective (Eq (7)) is required since there are a few works that estimate the temporal distance from the initial state distribution to propose the curriculum goals in a temporally distant region or explore based on this temporal information [5, 37]. To reflect the temporal distance in the cost function (Eq (7)), we modify it as $w(s_i, g_i^+) := \mathcal{CE}(p_{\text{pseudo}}(y = 1|s_i; g_i^+); y = p_{\text{pseudo}}(y = 1|g_i^+; g_i^+)) - V^\pi(s_0, \phi(s_i))$ ($\phi(\cdot)$ is goal space mapping) since the value function itself implicitly represents the temporal distance if we use the sparse reward or custom-defined reward similar to the sparse one. In this case, our proposed intrinsic reward outputs 1 for the desired goal and 0 for the explored states, and it works similarly to the sparse one.

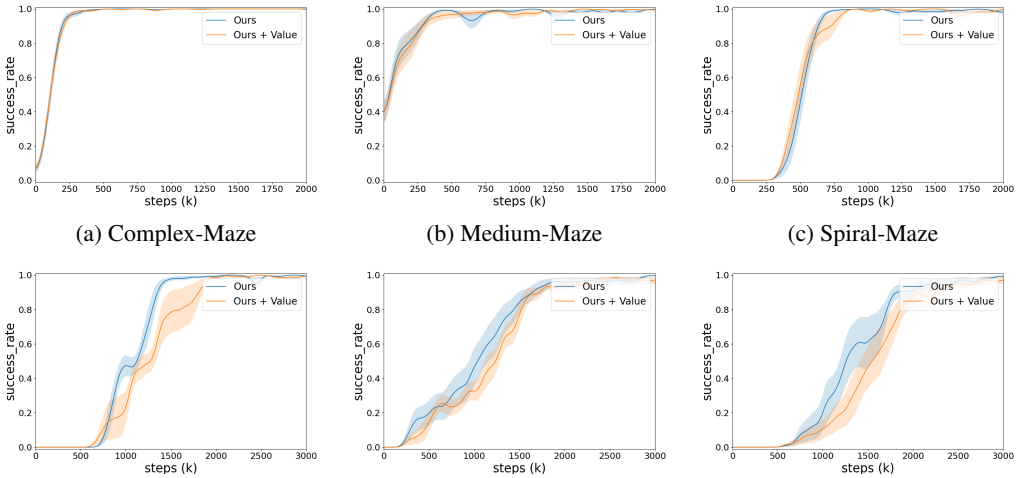
We experimented with this modified curriculum learning objective (**+Value**), and the results are shown in Figure 11, 12. It shows that there is no significant difference, which supports the superiority of our method in that our method achieves state-of-the-art results without considering additional temporal distance information.



(a) Complex-Maze (b) Medium-Maze (c) Spiral-Maze

(d) Ant Locomotion (e) Sawyer Push (f) Sawyer Pick&Place

Figure 11: Ablation study in terms of the average distance from the curriculum goals to the final goals (**Lower is better**). +Value means that we additionally consider the value function bias in the curriculum learning objective to reflect the temporal distance from the initial state distribution.

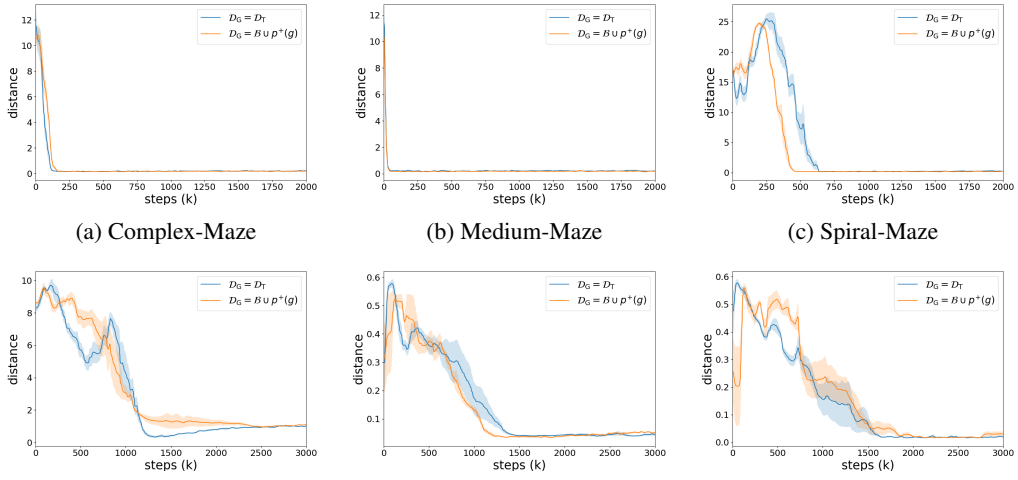


(a) Complex-Maze (b) Medium-Maze (c) Spiral-Maze

(d) Ant Locomotion (e) Sawyer Push (f) Sawyer Pick&Place

Figure 12: Ablation study in terms of the episode success rates. +Value means that we additionally consider the value function bias in the curriculum learning objective to reflect the temporal distance from the initial state distribution.

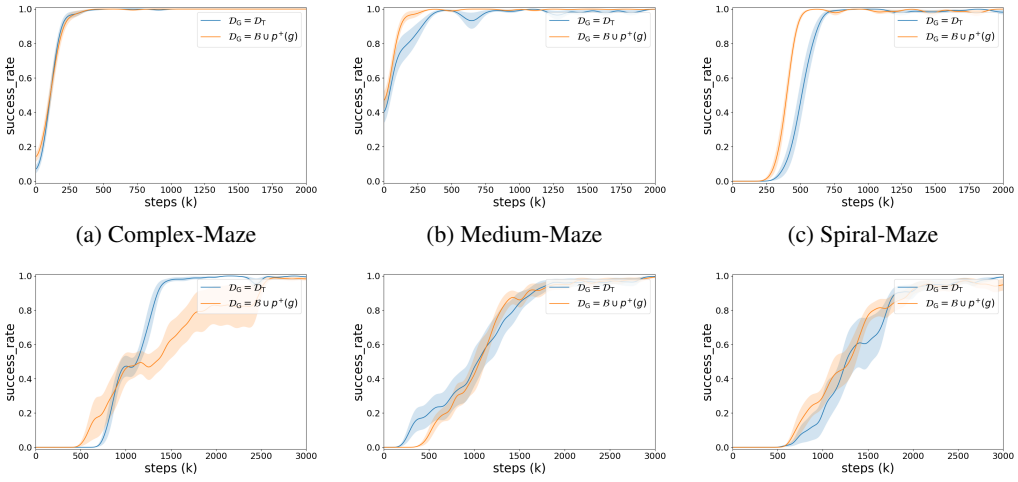
Choice of goal candidates in training conditional classifiers. As mentioned in the main script, we also experimented with different choices of the goal candidates when we train the conditional classifiers (Eq (5)). The default setting is $\mathcal{D}_{\mathrm{G}} = \mathcal{D}_{\mathrm{T}}$, and we also experimented with $\mathcal{D}_{\mathrm{G}} = \mathcal{B} \cup p^{+}(g)$. The results are shown in Figure 13, 14. It shows that there is no significant difference, which means we can even make the problem setting more strict by conditioning the classifier only with the visited states and the given desired outcome examples.



(a) Complex-Maze (b) Medium-Maze (c) Spiral-Maze

(d) Ant Locomotion (e) Sawyer Push (f) Sawyer Pick&Place

Figure 13: Ablation study in terms of the average distance from the curriculum goals to the final goals (**Lower is better**). There are no significant differences between the choice of goal candidates to train the conditional classifiers.



(a) Complex-Maze (b) Medium-Maze (c) Spiral-Maze

(d) Ant Locomotion (e) Sawyer Push (f) Sawyer Pick&Place

Figure 14: Ablation study in terms of the episode success rates. There are no significant differences between the choice of goal candidates to train the conditional classifiers.