Diversity-Based Trajectory and Goal Selection with Hindsight Experience Replay

Tianhong Dai[1], Hengyan Liu[1], Kai Arulkumaran[1,2], Guangyu Ren[1], and Anil Anthony Bharath[1]

November 9, 2021

[1]Imperial College London, London, United Kingdom
[2]Araya Inc., Tokyo, Japan

Table of contents

Sparse Reward Setting in Deep Reinforcement Learning

Sparse Reward Setting in DRL

In many real-world scenarios, an agent faces the challenge of sparse extrinsic reward. A typical condition is where an agent has to reach a goal and only receives a positive or non-negative reward signal when the agent is close enough to the target.

- Pros
  - No need for designing reward functions manually with specific domain knowledge.
  - Dense reward functions might only lead to specific solutions.
- Cons
  - Difficult to achieve positive feedback in the environment.
  - Lead to longer training time or even fail to get a promising policy.

Goal-Oriented RL with Sparse Reward Setting

RL can be expanded to the multi-goal setting, where the agent's policy and the environment's reward function $\mathcal{R}(s_t, a_t)$ are also conditioned on a goal g.

- Desired goal g: the desired configuration of a target object (e.g., position).
- Achieved goal g_{t+1}^{ac}: the current configuration of a target object.

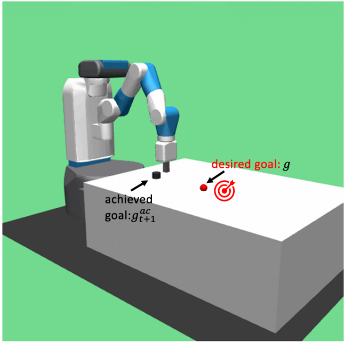


Figure 1: Description of goal-oriented environment.

Goal-Oriented RL with Sparse Reward Setting

The reward function can be defined as:

$$\mathcal{R}\left(g, g_{t+1}^{ac}\right) := \begin{cases} 0 & \text{if } \left\|g_{t+1}^{ac} - g\right\| \leq \epsilon \\ -1 & \text{otherwise.} \end{cases} \tag{1}$$

Thus, it is difficult to achieve non-negative reward from the environment.

Hindsight Experience Replay

Hindsight experience replay (HER) was proposed to improve the learning efficiency of goal-oriented RL agents in sparse reward settings: when past experience is replayed to train the agent, the desired goal g is replaced (in "hindsight") with the achieved goal g^{ac}_{t+1}, generating many positive experiences.

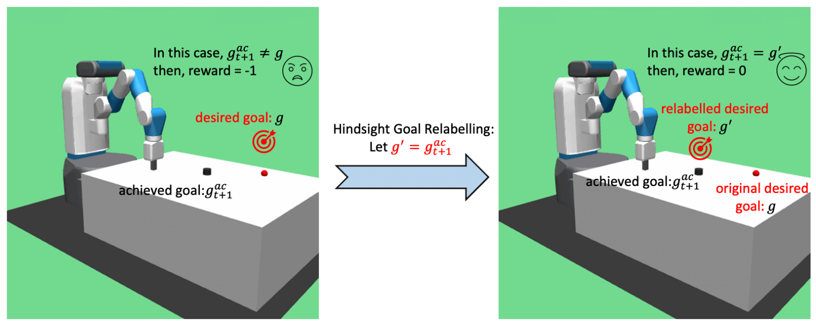


Figure 2: Illustration of hindsight goal relabelling.

Limitations of HER and Previous Works

- Limitations
  - It can be inefficient in its use of uniformly sampling transitions during training.
- HER with Energy-Based Prioritisation (HEBP) [Zhao et al., 2018]
  - Assume semantic knowledge about the goal-space and use the energy of the target objects to sample trajectories with high energies, and then samples transitions uniformly.
- Curriculum-Guided HER (CHER) [Fang et al., 2019]
  - Sample trajectories uniformly, and then sample transitions based on a mixture of proximity to the desired goal and the diversity of the samples.

Our Method

Diversity-Based Trajectory and Goal Selection with HER

In this work, we introduce diversity-based trajectory and goal selection with HER (DTGSH; See Fig. 1), which samples trajectories based on the diversity of the goals achieved within the trajectory, and then samples transitions based on the diversity of the set of samples.

- Converges faster and reaches higher rewards than prior work.
- Without requiring domain knowledge or tuning a curriculum.
- Based on a single concept — determinantal point processes (DPPs)

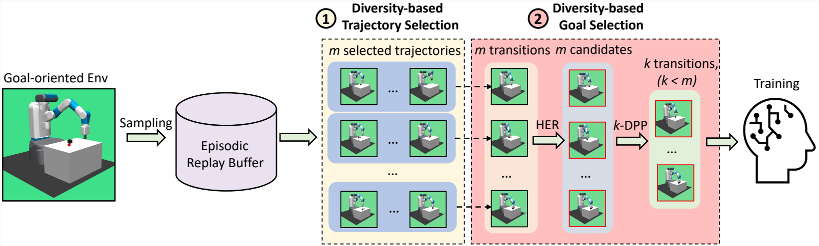


Figure 3: Illustration of DTGSH.

Determinantal Point Processes (DPPs)

Formally, for a discrete set of points $\mathcal{Y} = \{x_1, x_2, \cdots, x_N\}$, a point process $\mathcal{P}$ is a probability measure over all $2^{|\mathcal{Y}|}$ subsets. $\mathcal{P}$ is a DPP if a random subset $\mathbf{Y}$ is sampled with probability:

$$\mathcal{P}_L(\mathbf{Y} = Y) = \frac{\det(L_Y)}{\sum_{Y' \subseteq \mathcal{Y}} \det(L_{Y'})} = \frac{\det(L_Y)}{\det(L + I)}, \quad (2)$$

The kernel matrix L can be represented as the Gram matrix $L = X^T X$, where each column of X is the feature vector of an item in $\mathcal{Y}$. The determinant, $\det(L_Y)$, represents the (squared) volume spanned by vectors $x_i \in Y$.

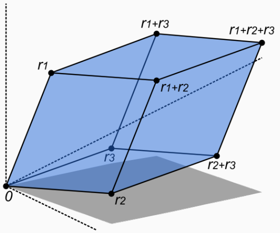


Figure 4: Illustration of determinant.

Diversity-Based Trajectory Selection

We propose a diversity-based prioritization method to select valuable trajectories for efficient training. We hypothesise that trajectories that achieve diverse goal states g_t^{ac} are more valuable for training.

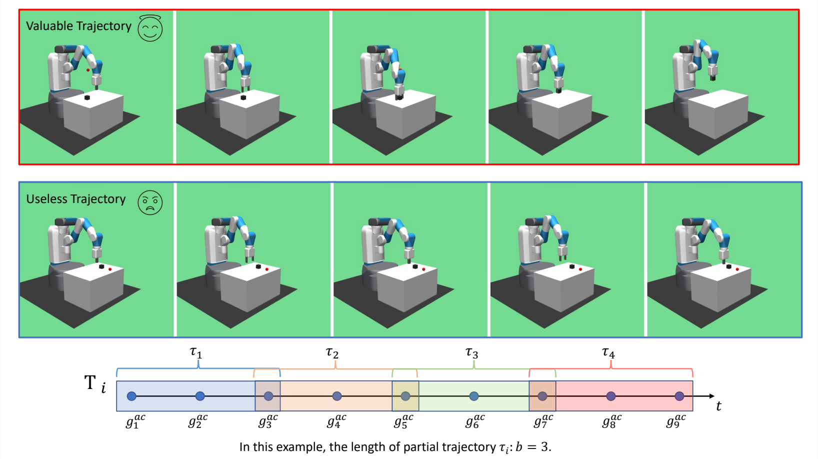


Figure 5: Illustration of diversity-based trajectory selection.

Diversity-Based Trajectory Selection

The diversity d_{τ_j} of each partial trajectory τ_j can be computed as:

$$d_{\tau_j} = \det(L_{\tau_j}), \tag{3}$$

where L_{τ_j} is the kernel matrix of partial trajectory τ_j:

$$L_{\tau_j} = M^T M, \tag{4}$$

and $M = [\hat{g}_n^{ac}, \hat{g}_{n+1}^{ac}, \cdots, \hat{g}_{n+b-1}^{ac}]$, where each $\hat{g}^{ac}$ is the ℓ_2-normalised version of the achieved goal g^{ac}. Finally, the diversity $d_{\mathcal{T}}$ of trajectory $\mathcal{T}$ is the sum of the diversity of its N_p constituent partial trajectories:

$$d_{\mathcal{T}} = \sum_{j=1}^{N_p} d_{\tau_j}. \tag{5}$$

The probability $p(\mathcal{T}_i)$ of sampling trajectory $\mathcal{T}_i$ from a replay buffer of size N_e is:

$$p(\mathcal{T}_i) = \frac{d_{\mathcal{T}_i}}{\sum_{n=1}^{N_e} d_{\mathcal{T}_n}}. \tag{6}$$

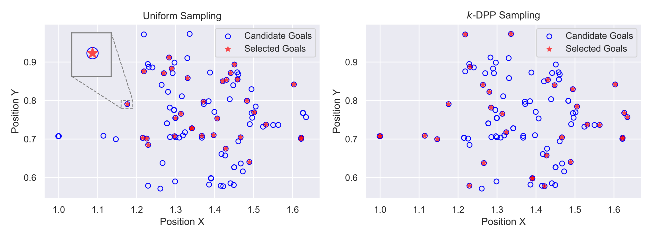
Diversity-Based Goal Selection

In order to form a minibatch with diverse goals for more efficient learning, we use k-DPPs for sampling goals. Compared to the standard DPP, a k-DPP is a conditional DPP where the subset Y has a fixed size k, with the probability distribution function:

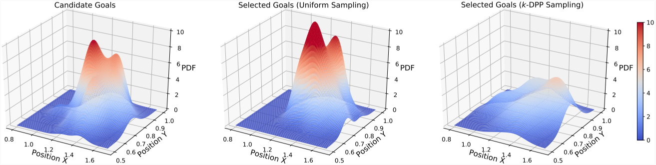
$$\mathcal{P}_L^k(\mathbf{Y} = Y) = \frac{\det(L_Y)}{\sum_{|Y'|=k} \det(L_{Y'})}. \tag{7}$$

1. Uniformly sample a transition from each of the m trajectories, and relabel them using hindsight to form m candidate transitions.
2. A k-DPP is used to sample k ($k < m$) transitions based on the relabelled goals g' (i.e. candidate goals).
3. Use k selected transitions to train the model.

Diversity-Based Goal Selection



(a) Plot of candidate goals and selected goals.



(b) Kernel density estimation of the distributions of goals.

Figure 6: Visualisation of $k = 32$ goals selected from $m = 100$ candidate goals of the Push task using either uniform sampling or k-DPP sampling.

Experiment Settings

Environments

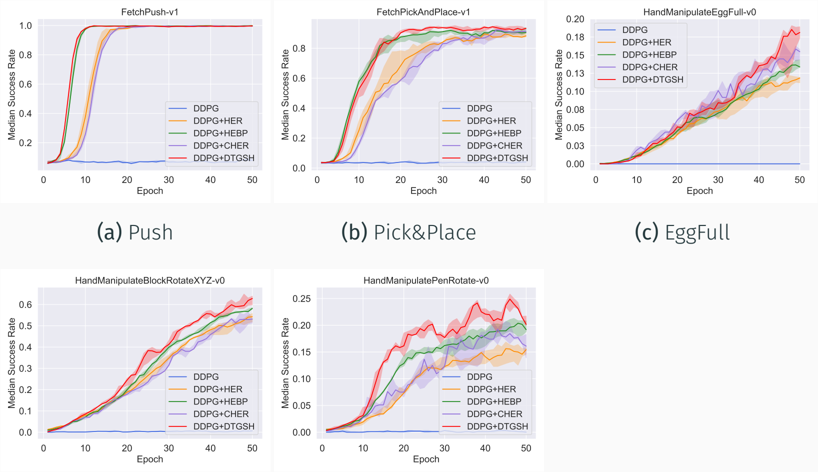
We evaluate our proposed method on five challenging robotic manipulation tasks.

(a) Push (b) Pick&Place (c) EggFull (d) BlockRotate (e) PenRotate

Figure 7: Robotic manipulation environments. (a-b) use the Fetch robot, and (c-e) use the Shadow Dexterous Hand.

Results

Comparison with Previous Works - Learning Curve



(a) Push (b) Pick&Place (c) EggFull

(d) BlockRotate (e) PenRotate

Figure 8: Success rate of DTGSH and baseline approaches.

Comparison with Previous Works - Quantitative Results

	Push	Pick&Place	EggFull	BlockRotate	PenRotate
DDPG	0.09 ± 0.01	0.04 ± 0.00	0.00 ± 0.00	0.01 ± 0.00	0.00 ± 0.00
DDPG+HER	$\mathbf{1.00\pm0.00}$	0.89 ± 0.03	0.11 ± 0.01	0.55 ± 0.04	0.15 ± 0.02
DDPG+HEBP	$\mathbf{1.00\pm0.00}$	0.91 ± 0.03	0.14 ± 0.02	0.59 ± 0.02	0.20 ± 0.03
DDPG+CHER	$\mathbf{1.00\pm0.00}$	0.91 ± 0.04	0.15 ± 0.01	0.54 ± 0.04	0.17 ± 0.03
DDPG+DTGSH	$\mathbf{1.00\pm0.00}$	$\mathbf{0.94\pm0.01}$	$\mathbf{0.17\pm0.03}$	$\mathbf{0.62\pm0.02}$	$\mathbf{0.21\pm0.02}$

Table 1: Final mean success rate $\pm$ standard deviation, with best results in **bold**.

Conclusion

Conclusion

Our experiments empirically show that DTGSH achieves:

- Faster learning speed and higher final performance.
- Does not require semantic knowledge of the goal space.
- Does not require tuning a curriculum.

Thank You For Listening.