

# PRM with off-policy RL method

Saito Karuha

November 2024

## 1 Pseudo Code

---

**Algorithm 1** DDPG For PRM Training

---

```
1: Initialize: CriticNet:  $Q_{\omega}(s, a)$  , ActionNet:  $\mu_{\theta}(s)$ 
2: Initialize:  $Q_{\omega^-}(s, a) \leftarrow Q_{\omega}(s, a)$  ,  $\mu_{\theta^-}(s) \leftarrow \mu_{\theta}(s)$ 
3: for  $e = 0$  to  $E$  do
4:   Initialize initial state  $s_1$  (Randomly pick a question from UCB-Math)
5:   for  $t = 1$  to  $T$  and  $done == True$  do
6:     Choose an action  $a_t \sim \mu_{\theta}(s_t)$ 
7:      $r_t \leftarrow Env(s_t, a_t)$  ;  $s_{t+1} \leftarrow [s_t, a_t]$ 
8:     if Buffer is big enough then
9:       Randomly pick  $N$  tuples  $\{(s_i, a_i, r_i, s_{i+1})\}_{i=1, \dots, N}$ 
10:      Sample  $K$  actions:  $a_{i+1}^m \sim \mu_{\theta^-}(s_{i+1})$  ,  $(m = 1, \dots, k)$ 
11:      Calculate for every tuples:
```

$$y_i = r_i + \gamma * \max_{m \in \{1, \dots, k\}} Q_{\omega^-}(s_{i+1}, a_{i+1}^m) \quad (1)$$

$$A_i = r_i + \gamma * Random_j Q_{\omega^-}(s_{i+1}, a_{i+1}^j) - Q_{\omega}(s_i, a_i) \quad (2)$$

```
12:      Compute loss for CriticNet( $L$ ) and ActorNet( $J$ ) respectively:
13:       $L = \frac{1}{N} \sum_{i=1}^N y_i - Q_{\omega}(s_i, a_i)$ 
14:       $J = \frac{1}{N} \sum_{i=1}^N \min(\frac{\pi_{\theta}(a_i|s_i)}{\pi_{\theta^-}(a_i|s_i)} A_i, clip(\frac{\pi_{\theta}(a_i|s_i)}{\pi_{\theta^-}(a_i|s_i)}, 1 - \epsilon, 1 + \epsilon) A_i)$ 
15:      Update for Critic and Actor Network
16:      Soft update target Network
17:       $\omega^- \leftarrow \tau \omega + (1 - \tau) \omega$  ,  $\theta^- \leftarrow \tau \theta + (1 - \tau) \theta$ 
18:    end if
19:  end for
20: end for
```

---

## 2 Problems