NEURAL INFORMATION PROCESSING SYSTEMS

Exploiting the Replay Memory Before Exploring the Environment: Enhancing Reinforcement Learning Through Empirical MDP Iteration

Hongming Zhang[1], Chenjun Xiao[2], Chao Gao[3], Han Wang[1], Bo Xu[4], Martin Müller[1]

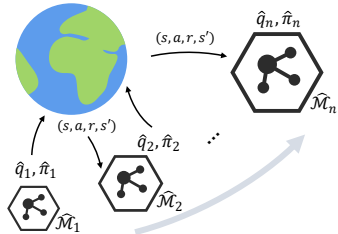
[1]Department of Computing Science and Amii, University of Alberta
[2]The Chinese University of Hong Kong, Shenzhen
[3]Edmonton Research Center, Huawei Canada
[4]Institute of Automation, Chinese Academy of Sciences

Motivation

- Reinforcement learning is typically based on the *optimal Bellman equation* in a sampled manner.
- It suffers from *estimation errors* due to the combination of *applying the max operator to out-of-sample actions* and *bootstrapping from a function approximator*.
- How about focusing on the **currently available data**?

Our Framework: Empirical MDP Iteration (EMIT)

- Constructs a sequence of empirical MDPs using data from the growing replay memory.
- Restrict the Bellman update to in-sample bootstrapping that uniquely solves each empirical MDP.
- Gradually expand from the empirical MDPs to the original MDP through new data collection.



An overview of our method.

Preliminaries

Markov Decision Process (MDP). $\mathcal{M} = (\mathcal{S}, \mathcal{A}, R, P, \gamma)$.

Empirical MDP. $\widehat{\mathcal{M}} := (\hat{\mathcal{S}}, \hat{\mathcal{A}}, \hat{R}, \hat{P}, \gamma)$. Given a dataset $\mathcal{D}$ from MDP $\mathcal{M}$, the empirical MDP has state space $\hat{\mathcal{S}} = \{s | s \in \mathcal{D}\}$, and action space $\hat{\mathcal{A}} = \{a | (s, a) \in \mathcal{D}\}$, with reward function $\widehat{R}(s, a) = R(s, a)$ if $(s, a) \in \mathcal{D}$, and $\widehat{R}(s, a) = -\infty$ otherwise. $\hat{P}(s'|s, a) = N(s, a, s') / \sum_{s'} N(s, a, s')$ is the empirical transition dynamics based on visit counts in $\mathcal{D}$, and $\gamma \in (0,1)$ is the discount factor.

Bellman Update.

$$Q(s, a) \leftarrow r + \gamma \max_{a'} Q(s', a') \qquad (1)$$

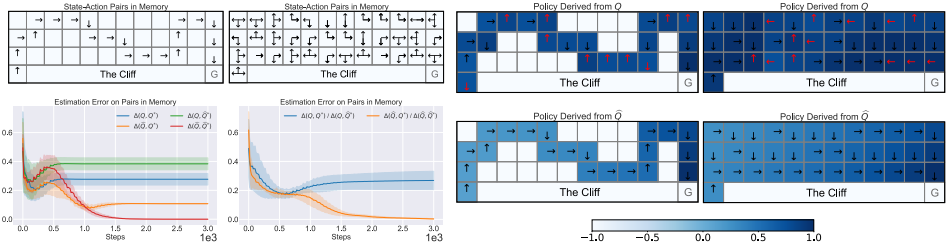
In-Sample Bellman Update.

$$\hat{Q}(s, a) \leftarrow r + \gamma \max_{a': (s', a') \in \mathcal{D}} \hat{Q}(s', a') \qquad (2)$$

Eq.(2) bootstraps only from in-sample actions.

Analysis of EMIT

Illustrative Example



(a) Value errors compared with Q^* and $\hat{Q}^*$. (b) Learned polices.

Proposition 3.2. If the data coverage $\mathcal{D}$ is incomplete, then the Bellman update Eq.(1) neither guarantees convergence to the optimal value Q^* for the original MDP $\mathcal{M}$ nor to the optimal value $\hat{Q}^*$ for the empirical MDP $\hat{\mathcal{M}}$, even in the limit of infinite updates.

Proposition 3.3. The in-sample Bellman update Eq.(2) uniquely converges to the optimal value $\hat{Q}^*$ for the empirical MDP $\hat{\mathcal{M}}$ in the limit of infinitely many updates. Furthermore, if the optimal trajectory for $\mathcal{M}$ is included in $\hat{\mathcal{M}}$, then the greedy policy derived from $\hat{Q}^*$ is also optimal for $\mathcal{M}$ following the optimal trajectory.

Proposition 3.4. Assume $\mathcal{M}$ has deterministic transitions. If $\{\mathcal{D}_i\}$ are datasets collected from $\mathcal{M}$ with $\mathcal{D}_1 \subset \mathcal{D}_2 \subset \cdots \subset \mathcal{D}_n$ and corresponding empirical MDPs $\{\hat{\mathcal{M}}_i\}$, then $\hat{Q}_1^* \leq \hat{Q}_2^* \leq \cdots \leq \hat{Q}_n^*$.

Algorithm Instantiation with EMIT

Regularization for stable learning

$$\mathcal{L} = MSE\big(Q_\theta, Q_{\text{target}}\big) + \alpha MSE(Q_\theta, \hat{Q}) \tag{4}$$

Here, $Q_{\text{target}} = r + \gamma \max_{a'} Q(s', a')$ and α is a parameter controlling the level of regularization.

Exploration to grow $\widehat{\mathcal{M}}$

Define the absolute difference between Q and $\hat{Q}$ at (s, a):

$$\delta(s, a) = |Q(s, a) - \hat{Q}(s, a)| \tag{5}$$

For algorithms in discrete action spaces such as DQN [2]:

$$\pi = \begin{cases} \text{random action}, & p = \epsilon \\ \text{argmax}_a\big(Q(s, a) + \delta(s, a)\big), & p = 1 - \epsilon \end{cases} \tag{6}$$

For algorithms in continuous action spaces such as TD3 [3]:

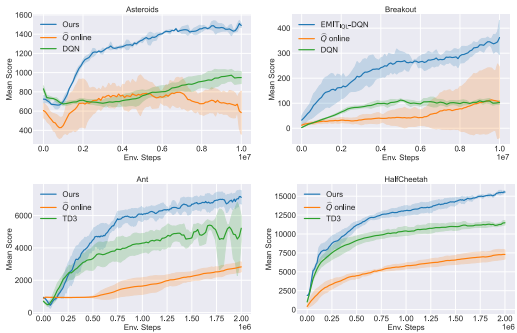
$$a \leftarrow \pi(s) + \epsilon, \quad \epsilon \sim \text{clip}(\mathcal{N}(0, \delta(s, a), -c, c) \tag{7}$$

The added noise is clipped as in TD3.

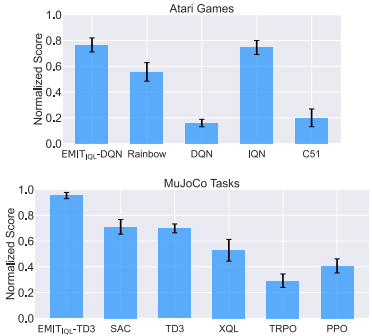
Algorithm 1 Empirical MDP Iteration for Enhancing a Q Learning Algorithm `Alg (EMIT-Alg)`

`Alg.Initialize` the replay memory $\mathcal{D}$ and action value network Q_θ
`EMIT.Initialize` $\widehat{Q}_\theta$
`Alg.Initialize` the environment $s_0 \leftarrow Env$
for environment step $t = 0$ **to** T **do**
  `Alg.Select` an action $a_t = \texttt{Alg.act}(s_t, \widehat{Q}_\theta)$ as Eq. (6) or (7) {$\widehat{Q}$ guided exploration}
  `Alg.Execute` a_t in Env and get r_t, s_{t+1}
  `Alg.Store` transition (s_t, a_t, r_t, s_{t+1}) in $\mathcal{D}$
  `Alg.Sample` random minibatch of transitions $\mathcal{B}$ from $\mathcal{D}$
  `EMIT.update`$(\widehat{Q}_\theta, \mathcal{B})$ w.r.t MSE loss derived by Eq.(2) {Learning of $\widehat{Q}$}
  `Alg.update`$(Q_\theta, \mathcal{B}, \widehat{Q}_\theta)$ as Eq.(4) {$\widehat{Q}$ regularized learning for Q}
end for

Experiments

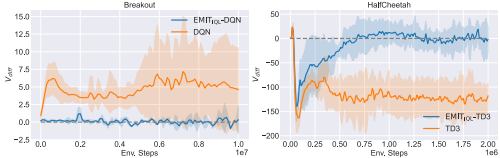


(a) Performance Enhancement of EMIT for DQN and TD3

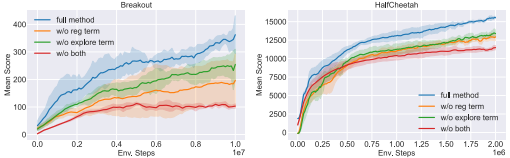


(b) Overall Performance across Various Tasks

Experiments



(a) Estimation error of Q after regularization.



(b) The contribution of each component of EMIT.

Conclusions

- Unlike previous methods that solely focus on solving the entire original MDP, we propose an Empirical MDP Iteration (EMIT) framework that uniquely solves each empirical MDP and incrementally approaches the original MDP through new data collection.
- Instantiate EMIT with DQN and TD3, and conduct extensive experiments on Atari and MuJoCo tasks. Demonstrate strong performance enhancements for both methods.

Future work

- Design a learning process that is directly based on in-sample Bellman update and also take the exploration mechanism into consideration.

THANKS !

H Zhang, C Xiao, C Gao, H Wang, B Xu, M Müller. Enhancing Reinforcement Learning Through Empirical MDP Iteration, NeurIPS 2024.