

7. Appendix: Theoretical Derivation

Theorem 7.1 (Distinguishable Representation Property). *The similarity (defined as inner product $\langle \cdot, \cdot \rangle$) between normalized representations $\Phi(s, a; \Theta_+)$ of the Q -network and $\mathbb{E}_{s', a'} \Phi(s', a'; \Theta'_+)$ satisfies*

$$\langle \Phi(s, a; \Theta_+), \mathbb{E}_{s', a'} \Phi(s', a'; \Theta'_+) \rangle \leq \frac{1}{\gamma} - \frac{r(s, a)^2}{2\|\Theta_{-1}\|^2}, \quad (11)$$

where s, a and Θ_+ are state, action, and parameters of the Q -network except for those of the last layer. While s', a', Θ'_+ are the state, action at the next time step, and parameters of the target Q -network except for those of the last layer. And Θ_{-1} is the parameters of the last layer of Q -network.

Proof. Following eq. (3), the Bellman Equation eq. (2) can be rewritten as

$$\Phi(s, a; \Theta_+)^T \Theta_{-1} = r(s, a) + \gamma \mathbb{E}_{s', a'} \Phi(s', a'; \Theta'_+)^T \Theta_{-1}. \quad (12)$$

After the policy evaluation converges, Θ and Θ' satisfy $\Theta = \Theta'$. Thus we have

$$\begin{aligned} & (\Phi(s, a; \Theta_+)^T - \gamma \mathbb{E}_{s', a'} \Phi(s', a'; \Theta'_+)^T) \Theta_{-1} = r(s, a) \\ & \|(\Phi(s, a; \Theta_+)^T - \gamma \mathbb{E}_{s', a'} \Phi(s', a'; \Theta'_+)^T) \Theta_{-1}\| = |r(s, a)| \\ & \|(\Phi(s, a; \Theta_+)^T - \gamma \mathbb{E}_{s', a'} \Phi(s', a'; \Theta'_+)^T)\| \|\Theta_{-1}\| \cos \varphi = |r(s, a)| \\ & \|(\Phi(s, a; \Theta_+)^T - \gamma \mathbb{E}_{s', a'} \Phi(s', a'; \Theta'_+)^T)\| \|\Theta_{-1}\| \geq |r(s, a)| \\ & \|(\Phi(s, a; \Theta_+)^T - \gamma \mathbb{E}_{s', a'} \Phi(s', a'; \Theta'_+)^T)\| \geq \frac{|r(s, a)|}{\|\Theta_{-1}\|}. \end{aligned} \quad (13)$$

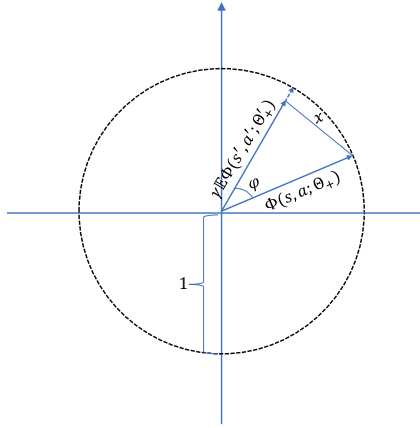


Figure 5. Normalized representation vectors.

Since the representation vectors are normalized, they should co-exist on some tangent plane as visualized in fig. 5. Let x be $\|\Phi(s, a; \Theta_+) - \gamma \mathbb{E} \Phi(s', a'; \Theta'_+)\|$, then we have $x \geq \frac{|r(s, a)|}{\|\Theta_{-1}\|}$, and

$$\cos \varphi = \frac{\|\Phi(s, a; \Theta_+)\|^2 + \|\gamma \mathbb{E} \Phi(s', a'; \Theta'_+)\|^2 - x^2}{2\|\Phi(s, a; \Theta_+)\| \|\gamma \mathbb{E} \Phi(s', a'; \Theta'_+)\|}. \quad (14)$$

Now we have

$$\begin{aligned}
\langle \Phi(s, a; \Theta_+), \gamma \mathbb{E}_{s', a'} \Phi(s', a'; \Theta'_+) \rangle &= \|\Phi(s, a; \Theta_+)\| \|\gamma \mathbb{E}_{s', a'} \Phi(s', a'; \Theta'_+)\| \cos \varphi \\
&= 1 \cdot \|\gamma \mathbb{E}_{s', a'} \Phi(s', a'; \Theta'_+)\| \cdot \frac{\|\Phi(s, a; \Theta_+)\|^2 + \|\gamma \mathbb{E} \Phi(s', a'; \Theta'_+)\|^2 - x^2}{2\|\Phi(s, a; \Theta_+)\| \|\gamma \mathbb{E} \Phi(s', a'; \Theta'_+)\|} \\
&= \frac{1 + \|\gamma \mathbb{E} \Phi(s', a'; \Theta'_+)\|^2 - x^2}{2} \\
&\leq \frac{1 + \gamma^2}{2} - \frac{r(s, a)^2}{2\|\Theta_{-1}\|^2} \\
&\leq 1 - \frac{r(s, a)^2}{2\|\Theta_{-1}\|^2}.
\end{aligned} \tag{15}$$

Thus, we have

$$\begin{aligned}
\langle \Phi(s, a; \Theta_+), \mathbb{E}_{s', a'} \Phi(s', a'; \Theta'_+) \rangle &\leq \frac{1}{\gamma} - \frac{r(s, a)^2}{2\gamma\|\Theta_{-1}\|^2} \\
&\leq \frac{1}{\gamma} - \frac{r(s, a)^2}{2\|\Theta_{-1}\|^2}.
\end{aligned} \tag{16}$$

□

In the following, for notational simplicity, we use X_i to denote S_i, A_i for all $i \in [n]$. For any $f \in \mathcal{F}$, $\|f\|_n^2 = 1/n \cdot \sum_{i=1}^n [f(X_i)]^2$. Since both \hat{O} and TQ are bounded by $V_{\max} = R_{\max}/(1 - \gamma)$, we only need to consider the case where $\log N_\delta \leq n$.

Let f_1, \dots, f_{N_δ} be the centers of minimal δ -cover the of \mathcal{F} . By the definition of δ -cover, there exists $k^* \in [N_\delta]$ such that $\|\hat{O} - f_{k^*}\|_\infty \leq \delta$. Notice that k^* is a random variable since \hat{O} is obtained from data.

Theorem 7.2 (One-step Approximation Error of PEER Update). *Suppose assumption 3.1 hold, let $\mathcal{F} \subseteq \mathcal{B}(\mathcal{S} \times \mathcal{A})$ be a class of measurable function on $\mathcal{S} \times \mathcal{A}$ that are bounded by $V_{\max} = R_{\max}/(1 - \gamma)$, and let σ be a probability distribution on $\mathcal{S} \times \mathcal{A}$. Also, let $\{(S_i, A_i)\}_{i \in [n]}$ be n i.i.d. random variables in following σ . Based on $\{(X_i, A_i, Y_i)\}_{i \in [n]}$, we define \hat{O} as the solution to the least-square with regularization problem,*

$$\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^n [f(S_i, A_i) - Y_i]^2 + \beta \Phi(s, a; \Theta) \mathbb{E} \Phi_{s', a'}(s', a'; \Theta'). \tag{17}$$

At the same time, for any $\delta > 0$, let $\mathcal{N}(\delta, \mathcal{F}, \|\cdot\|_\infty)$ be the

$$\|\hat{O} - TQ\|_\sigma^2 \leq (1 + \epsilon)^2 \cdot \omega(\mathcal{F}) + C \cdot V_{\max}^2 / (n \cdot \epsilon) + C' \cdot V_{\max} \cdot \delta + 2\beta \cdot G^2, \tag{18}$$

where C and C' are two absolute constants and is defined as

$$\omega(\mathcal{F}) = \sup_{g \in \mathcal{F}} \inf_{f \in \mathcal{F}} \|f - Tg\|_\sigma. \tag{19}$$

Proof. Step (i): We relate $\mathbb{E}[\|\hat{O} - TQ\|_n^2]$ with its empirical counterpart $\|\hat{O} - TQ\|_n^2$. Since $Y_i = R_i + \gamma \max_{a \in \mathcal{A}} Q(S_{i+1}, a)$ for each $i \in [n]$. By the definition of \hat{O} , for any $f \in \mathcal{F}$ we have

$$\sum_{i=1}^n [Y_i - \hat{O}(X_i)]^2 + \beta \Phi^\top(X_i; \Theta_{\hat{O}}) \mathbb{E} \Phi_{X_{i+1}}(X_{i+1}; \Theta'_{\hat{O}}) \leq \sum_{i=1}^n [Y_i - f(X_i)]^2 + \beta \Phi^\top(X_i; \Theta_f) \mathbb{E} \Phi_{X_{i+1}}(X_{i+1}; \Theta'_f). \tag{20}$$

For each $i \in [n]$, we define $\xi_i = Y_i - (TQ)(X_i)$. Then eq. (20) can be rewritten as

$$\|\hat{O} - TQ\|_n^2 \leq \|f - TQ\|_n^2 + \frac{1}{n} \sum_{i=1}^n \left[2\xi_i [\hat{O}(X_i) - f(X_i)] + \beta \left(\Phi^\top(X_i; \Theta_f) \mathbb{E} \Phi^\top(X_{i+1}; \Theta'_f) - \Phi^\top(X_i; \Theta_{\hat{O}}) \mathbb{E} \Phi(X_{i+1}; \Theta'_{\hat{O}}) \right) \right]. \tag{21}$$

We start by bounding the value of $\left(\Phi^\top(X_i; \Theta_f) \mathbb{E} \Phi(X_{i+1}; \Theta'_f) - \Phi^\top(X_i; \Theta_{\hat{O}}) \mathbb{E} \Phi(X_{i+1}; \Theta'_{\hat{O}})\right)$. First, by Cauchy-Schwartz Inequality, we have

$$\left|\Phi(X_i; \Theta_f) \mathbb{E} \Phi(X_{i+1}; \Theta'_f)\right| \leq \sqrt{\|\Phi(X_i; \Theta_f, +)\|^2} \cdot \sqrt{\|\mathbb{E} \Phi(X_{i+1}; \Theta'_{f,+})\|^2} \leq G^2, \quad (22)$$

where we used assumption 3.1 for the second inequality. Thus, by triangle inequality, we have

$$\left|\Phi^\top(X_i; \Theta_f) \mathbb{E} \Phi(X_{i+1}; \Theta'_f) - \Phi^\top(X_i; \Theta_{\hat{O}}) \mathbb{E} \Phi(X_{i+1}; \Theta'_{\hat{O}})\right| \leq 2G^2. \quad (23)$$

And eq. (21) reduces to

$$\|\hat{O} - TQ\|_n^2 \leq \|f - TQ\|_n^2 + \frac{2}{n} \sum_{i=1}^n [\xi_i [\hat{O}(X_i) - f(X_i)] + \beta G^2]. \quad (24)$$

Then we bound the rest on the right side of eq. (21). Since both f and Q are deterministic, we have $\mathbb{E}(\|f - TQ\|_n^2) = \|f - TQ\|_{\mathcal{T}}^2$. Moreover, since $\mathbb{E}(\xi_i | X_i) = 0$ by definition, we have $\mathbb{E}[\xi_i \cdot g(X_i)] = 0$ for any bounded and measurable function g . Thus it holds that

$$\mathbb{E} \left\{ \sum_{i=1}^n \xi_i \cdot [\hat{O}(X_i) - f(X_i)] \right\} = \mathbb{E} \left\{ \sum_{i=1}^n \xi_i \cdot [\hat{O} - (TQ)(X_i)] \right\}. \quad (25)$$

In addition, by triangle inequality and eq. (25) we have

$$\left| \mathbb{E} \left\{ \sum_{i=1}^n \xi_i \cdot [\hat{O}(X_i) - (TQ)(X_i)] \right\} \right| \leq \left| \mathbb{E} \left\{ \sum_{i=1}^n \xi_i \cdot [\hat{O} - f_{k^*}(X_i)] \right\} \right| + \left| \mathbb{E} \left\{ \sum_{i=1}^n \xi_i \cdot [f_{k^*}(X_i) - (TQ)(X_i)] \right\} \right|, \quad (26)$$

where f_{k^*} satisfies $\|f_{k^*}\| \leq \delta$. In the following, we upper bound the two terms on the right side of eq. (26) respectively. For the first term, by applying the Cauchy-Schwarz inequality twice, we have

$$\begin{aligned} \left| \mathbb{E} \left\{ \sum_{i=1}^n \xi_i \cdot [\hat{O} - f_{k^*}(X_i)] \right\} \right| &\leq \sqrt{n} \cdot \left| \mathbb{E} \left[\left(\sum_{i=1}^n \xi_i^2 \right)^{1/2} \cdot \|\hat{O} - f_{k^*}\|_n \right] \right| \\ &\leq \sqrt{n} \cdot [\mathbb{E}(\sum_{i=1}^n \xi_i^2)]^{1/2} \cdot [\mathbb{E}(\|\hat{O} - f_{k^*}\|_n^2)]^{1/2} \leq n\delta \cdot [\mathbb{E}(\xi_i^2)]^{1/2}. \end{aligned} \quad (27)$$

where we use the fact that $\{\xi_i\}_{i \in [n]}$ have the same marginal distributions and $\|\hat{O} - f_{k^*}\|_n \leq \delta$. Since both Y_i and TQ are bounded by V_{\max} , ξ_i is a bounded random variable by its definition. Thus, there exists a constant $C_\xi > 0$ depending on ξ such that $\mathbb{E}(\xi_i^2) \leq C_\xi^2 \cdot V_{\max}^2$. Then eq. (27) implies

$$\left| \mathbb{E} \left\{ \sum_{i=1}^n \xi_i \cdot [\hat{O}(X_i) - f_{k^*}(X_i)] \right\} \right| \leq C_\xi \cdot V_{\max} \cdot n\delta. \quad (28)$$

It remains to upper bound the second term on the right side of eq. (26). We define N_δ self-normalized random variables

$$Z_j = \frac{1}{\sqrt{n}} \sum_{i=1}^n \xi_i \cdot [f_j(X_i) - (TQ)(X_i)] \cdot \|f_j - (TQ)\|_n^{-1} \quad (29)$$

for all $j \in [N_\delta]$. Here recall that $\{f_j\}_{j \in [N_\delta]}$ are the centers of the minimal δ -covering of \mathcal{F} . Then we have

$$\begin{aligned} \left| \mathbb{E} \left\{ \sum_{i=1}^n \xi_i \cdot [f_{k^*}(X_i) - (TQ)(X_i)] \right\} \right| &= \sqrt{n} \cdot \mathbb{E}[\|f_{k^*} - TQ\|_n \cdot |Z_{k^*}|] \\ &\leq \sqrt{n} \cdot \mathbb{E} \{ [\|\hat{O} - TQ\|_n + \|\hat{O} - f_{k^*}\|_n] \cdot |Z_{k^*}| \} \leq \sqrt{n} \cdot \{ [\|\hat{O} - TQ\|_n + \delta] \cdot |Z_{k^*}| \}, \end{aligned} \quad (30)$$

where the first inequality follows from triangle inequality and the second follows from the fact that $\leq \delta$ eq. (30), we obtain

$$\begin{aligned} \mathbb{E} \{ [\|\hat{O} - TQ\|_n + \delta] \cdot |Z_{k^*}| \} &\leq \left(\mathbb{E} \{ [\|\hat{O} - TQ\|_n + \delta]^2 \} \right)^{1/2} \cdot [\mathbb{E}(Z_{k^*}^2)]^{1/2} \\ &\leq \left(\mathbb{E} [\|\hat{O} - TQ\|_n^2]^{1/2} + \delta \right) \cdot [\mathbb{E}(\max_{j \in [n]} Z_j^2)]^{1/2}, \end{aligned} \quad (31)$$

where the last inequality follows from

$$\mathbb{E} [\|\hat{O} - TQ\|_n] \leq \left\{ \mathbb{E} [\|\hat{O} - TQ\|_n^2] \right\}^{1/2}. \quad (32)$$

Moreover, since ξ_i is centered conditioning on $\{X_i\}$, ξ_i is a sub-Gaussian random variable. Specifically, there exists an absolute constant $H_\xi > 0$ such that $\|\xi_i\|_{\psi_2} \leq H_\xi \cdot V_{\max}$ for each $i \in [n]$. Here the ψ_2 -norm of a random variable W is defined as

$$\|W\|_{\psi_2} = \sup_{p \geq 1} p^{-1/2} [\mathbb{E}(|W|^p)]^{1/p}. \quad (33)$$

By the definition of Z_j in eq. (29), conditioning on $\{X_i\}_{i \in [n]}$, $\xi_i \cdot [f_j(X_i) - (TQ)(X_i)]$ is a centered and sub-Gaussian random variable with

$$\|\xi_i \cdot [f_j(X_i) - (TQ)(X_i)]\|_{\psi_2} \leq H_\xi \cdot V_{\max} \cdot |f_j(X_i) - (TQ)(X_i)|. \quad (34)$$

Moreover, since Z_j is a summation of independent sub-Gaussian random variables, by Lemma 5.9 of [59], the ψ_2 -norm of Z_j satisfies

$$\|Z_j\|_{\psi_2} \leq C \cdot H_\xi \cdot V_{\max} \cdot \|f_j - TQ\|_n^{-1} \cdot \left[\frac{1}{n} \sum_{i=1}^n |f_j(X_i) - (TQ)(X_i)|^2 \right]^{1/2} \leq C \cdot H_\xi \cdot V_{\max}, \quad (35)$$

where $C > 0$ is an absolute constant. Furthermore, by Lemma 5.14 and 5.15 of [59], Z_j^2 is a sub-exponential random variable, and its moment-generating function is bounded by

$$\mathbb{E} \left[\exp(t \cdot Z_j^2) \right] \leq \exp(C \cdot t^2 \cdot H_\xi^4 \cdot V_{\max}^4) \quad (36)$$

for any t satisfying $C' \cdot |t| \cdot H_\xi^2 \cdot V_{\max}^2 \leq 1$, where C and C' are two positive absolute constants. Moreover, by Jensen's Inequality, we bound the moment-generating function of $\max_{j \in [N_\delta]} Z_j^2$ by

$$\mathbb{E} \left[\exp(t \cdot \max_{j \in [N_\delta]} Z_j^2) \right] \leq \sum_{j \in [N_\delta]} \mathbb{E}[\exp(t \cdot Z_j^2)]. \quad (37)$$

Combining eq. (36) and eq. (37), we have

$$\mathbb{E}(\max_{j \in [N]} Z_j^2) \leq C^2 \cdot H_\xi^2 \cdot V_{\max}^2 \cdot \log N_\delta, \quad (38)$$

where $C > 0$ is an absolute constant. Hence, plugging eq. (38) into eq. (30) and eq. (31), we upper bound the second term of eq. (25) by

$$\left| \mathbb{E} \left\{ \sum_{i=1}^n \xi_i \cdot [f_{k^*}(X_i) - (TQ)(X_i)] \right\} \right| \leq \left(\left\{ \mathbb{E} \|\hat{O} - TQ\|_n^2 \right\}^{1/2} + \delta \right) \cdot C \cdot H_\xi \cdot V_{\max} \cdot \sqrt{n \cdot \log N_\delta}. \quad (39)$$

Finally, combining eq. (24), eq. (28) and eq. (39), we obtain the following inequality

$$\begin{aligned} \mathbb{E} [\|\hat{O} - TQ\|_n^2] &\leq \inf_{f \in \mathcal{F}} \mathbb{E} [\|f - TQ\|_n^2] + C_\xi \cdot V_{\max} \cdot \delta \\ &\quad + \left(\left\{ \mathbb{E} \|\hat{O} - (TQ)\| \right\}^{1/2} + \delta \right) \cdot C \cdot H_\xi \cdot V_{\max} + \sqrt{\log N_\delta / n} + 2 \cdot \beta \cdot G^2 \\ &\leq C \cdot V_{\max} \sqrt{\log N_\delta / n} + \inf_{f \in \mathcal{F}} \mathbb{E} [\|f - TQ\|_n^2] + C' \cdot V_{\max} \delta + 2 \cdot \beta \cdot G^2, \end{aligned} \quad (40)$$

where C and C' are two constants. Here in the first inequality we take the infimum over \mathcal{F} because eq. (20) holds for any $f \in \mathcal{F}$, and the second inequality holds because $\log N_\delta \leq n$.

Now we invoke a fact to obtain the final bound for $\mathbb{E}[\|\hat{O} - TQ\|_n^2]$ from eq. (40). Let a, b and c be positive numbers satisfying $a^2 \leq 2ab + c$. For any $\epsilon \in (0, 1]$, since $2ab \leq \frac{\epsilon}{1+\epsilon}a^2 + \frac{1+\epsilon}{\epsilon}b^2$, we have

$$a^2 \leq (1+\epsilon)^2 \cdot b^2/\epsilon + (1+\epsilon) \cdot c. \quad (41)$$

Therefore, applying eq. (41) to eq. (40) with $a^2 = \mathbb{E}[\|\hat{O} - TQ\|_n^2]$, $b = C \cdot V_{\max} \cdot \sqrt{\log N}$ and $c = \inf_{f \in \mathcal{F}} \mathbb{E}[\|f - TQ\|_n^2] + C' \cdot V_{\max} \cdot \delta$, we obtain

$$\mathbb{E}[\|\hat{O} - TQ\|_n^2] \leq (1+\epsilon) \cdot \inf_{f \in \mathcal{F}} \mathbb{E}[\|f - TQ\|_n^2] + C \cdot V_{\max}^2 \cdot \log N_\delta / (n\epsilon) + C' \cdot V_{\max} \cdot \delta + 2\beta G^2, \quad (42)$$

where C and C' are two positive absolute constants. This concludes the first step.

Step (ii): In this step, we relate the population risk $\|\hat{O} - TQ\|_\delta^2$ with $\mathbb{E}[\|\hat{O} - TQ\|_n^2]$, which is bounded in the first step. To begin with, we generate n i.i.d. random variables $\{\tilde{X}_i = (\tilde{S}_i, \tilde{A}_i)\}_{i \in [n]}$ following σ , independent of $\{(S_i, A_i, R_i, S'_i)\}_{i \in [n]}$. Since $\|\hat{O} - f_{k^*}\|_\infty \leq \delta$, for any $x \in \mathcal{S} \times \mathcal{A}$, we have

$$|[\hat{O}(x) - (TQ)(x)]^2 - [f_{k^*}(x) - (TQ)(x)]^2| = |\hat{O}(x) - f_{k^*}(x)| \cdot |\hat{O}(x) + f_{k^*}(x) - 2(TQ)(x)| \leq 4V_{\max} \cdot \delta, \quad (43)$$

where the last inequality follows from the fact that $\|TQ\|_\infty \leq V_{\max}$ and $\|f\|_\infty \leq V_{\max}$ for any $f \in \mathcal{F}$.

Then by the definition of $\|\hat{O} - TQ\|_\delta^2$ and eq. (43), we have

$$\begin{aligned} \|\hat{O} - TQ\|_\sigma^2 &= \mathbb{E} \left\{ \frac{1}{n} \sum_{i=1}^n [\hat{O}(\tilde{X}_i) - (TQ)(\tilde{X}_i)]^2 \right\} \\ &\leq \mathbb{E} \left\{ \|\hat{O} - TQ\|_n^2 + \frac{1}{n} \sum_{i=1}^n [f_{k^*}(\tilde{X}_i) - (TQ)(\tilde{X}_i)]^2 - \frac{1}{n} \sum_{i=1}^n [f_{k^*}(X_i) - (TQ)(\tilde{X}_i)]^2 \right\} + 8V_{\max} \cdot \delta \\ &= \mathbb{E}(\|\hat{O} - TQ\|_n^2) + \mathbb{E} \left[\frac{1}{n} \sum_{i=1}^n h_{k^*}(X_i, \tilde{X}_i) \right] + 8V_{\max} \cdot \delta, \end{aligned} \quad (44)$$

where we apply eq. (43) to obtain the first inequality, and in the last equality we define

$$h_j(x, y) = [f_j(y) - (TQ)(y)]^2 - [f_j(x) - (TQ)(x)]^2, \quad (45)$$

for any $x, y \in \mathcal{S} \times \mathcal{A}$ and any $j \in [N_\delta]$. Note that h_{k^*} is a random function since k^* is random. By the definition of h_j , we have $|h_j(x, y)| \leq 4V_{\max}^2$ for any $(x, y) \in \mathcal{S} \times \mathcal{A}$ and $\mathbb{E}[h_j(X_i, \tilde{X}_i)] = 0$ for any $i \in [n]$. Moreover, the variance of $h_j(X_i, \tilde{X}_i)$ satisfies

$$\begin{aligned} \text{Var}[h_j(X_i, \tilde{X}_i)] &= 2 \text{Var} \{ [f_j(X_i) - (TQ)(X_i)]^2 \} \\ &\leq 2\mathbb{E} \{ [f_j(X_i) - (TQ)(X_i)]^4 \} \leq 8Y^2 \cdot V_{\max}^2, \end{aligned} \quad (46)$$

where we define Y by letting

$$Y = \max(4V_{\max}^2 \cdot \log N_\delta / n, \max_{j \in [N_\delta]} \mathbb{E} \{ [f_j(X_i) - (TQ)(X_i)]^2 \}). \quad (47)$$

Furthermore, we define

$$T = \sup_{j \in [N_\delta]} \left| \sum_{i=1}^n h(X_i, \tilde{X}_i) / Y \right|. \quad (48)$$

Combining eq. (44) and eq. (48),

$$\|\hat{O} - TQ\|_\sigma^2 \leq \mathbb{E}[\|\hat{O} - TQ\|_n^2] + Y/n \cdot \mathbb{E}[T] + 8V_{\max} \cdot \delta. \quad (49)$$

In the following, we use Bernstein's Inequality to establish an upper bound for $\mathbb{E}(T)$:

Lemma 7.3. (Bernstein's Inequality) Let U_1, \dots, U_n be n independent random variables satisfying $\mathbb{E}(U_i) = 0$ and $\sigma^2 \leq \sigma^2$ for all $i \in [n]$. Then for any $t > 0$, we have

$$\mathbb{P}\left(\left|\sum_{i=1}^n U_i\right| \geq t\right) \leq 2 \exp\left(\frac{-t^2}{2M \cdot t/3 + 2\sigma^2}\right), \quad (50)$$

where $\sigma^2 = \sum_{i=1}^n \text{var}(U_i)$ is the variance of $\sum_{i=1}^n U_i$.

We first apply Bernstein's Inequality by setting $U_i = h_j(X_i, \tilde{X}_i)/Y$ for each $i \in [n]$. Then we take a union bound for all $j \in [N_\delta]$ to obtain

$$\mathbb{P}(T \geq t) = \mathbb{P}\left[\sup_{j \in [N_\delta]} \frac{1}{n} \left|\sum_{i=1}^n h_j(X_i, \tilde{X}_i)/Y\right| \geq t\right] \leq 2N_\delta \cdot \exp\left\{\frac{-t^2}{8V_{\max}^2 \cdot [t/(3Y) + n]}\right\}. \quad (51)$$

Since T is nonnegative, $\mathbb{E}(T) = \int_0^\infty \mathbb{P}(T \geq t) dt$. Thus, for any $u \in (0, 3Y \cdot n)$,

$$\begin{aligned} \mathbb{E}(T) &\leq u + \int_u^\infty \mathbb{P}(T \geq t) dt \leq u + 2N_\delta \int_u^{3Y \cdot n} \exp\left(\frac{-t^2}{16V_{\max}^2 \cdot n}\right) dt + 2N_\delta \int_{3Y \cdot n}^\infty \exp\left(\frac{-3Y \cdot t}{16V_{\max}^2}\right) dt \\ &\leq u + 32N_\delta \cdot V_{\max} \cdot n/u \cdot \exp\left(\frac{-u^2}{16V_{\max}^2 \cdot n}\right) + 32N_\delta \cdot V_{\max}^2/(3Y) \cdot \exp\left(\frac{-9Y^2 \cdot n}{16V_{\max}^2}\right), \end{aligned} \quad (52)$$

where in the second inequality we use the fact that $\int_s^\infty \exp(-t^2/2) dt \leq 1/s \cdot \exp(-s^2/2)$. Now we set $u = 4V_{\max} \sqrt{n \cdot \log N_\delta}$ in eq. (52) and plug in the definition of Y in eq. (46) to obtain

$$\mathbb{E} \leq 4V_{\max} \log n \cdot N_\delta + 8V_{\max} \sqrt{n/\log N_\delta} + 6V_{\max} \sqrt{n/\log N_\delta} \leq 8V_{\max} \sqrt{n \cdot \log N_\delta}, \quad (53)$$

where the last inequality holds when $\log N_\delta \geq 4$. Moreover, the definition of Y in eq. (46) implies that $Y \leq \max[2V_{\max} \sqrt{\log N_\delta/n}, \|\hat{O} - TQ\|_\sigma^2 + \delta]$. In the following, we only need to consider the case where $Y \leq \|\hat{O} - TQ\|_\sigma + \delta$, since we already have eq. (18) if $\|\hat{O} - TQ\| + \delta \leq 2V_{\max} \sqrt{\log N_\delta/n}$, which concludes the proof.

Then, when $Y \leq \|\hat{O} - TQ\|_\sigma + \delta$ holds, combining eq. (49) and eq. (53) we have,

$$\begin{aligned} \|\hat{O} - TQ\|_\delta^2 &\leq \mathbb{E}[\|\hat{O} - TQ\|_n^2] + 8V_{\max} \sqrt{\log N_\delta/n} \cdot \|\hat{O} - TQ\|_\delta + 8V_{\max} \sqrt{\log N_\delta/n} \cdot \delta + 8V_{\max} \cdot \delta \\ &\leq \mathbb{E}[\|\hat{O} - TQ\|_n^2] + 8V_{\max} \sqrt{\log N_\delta/n} \cdot \|\hat{O} - TQ\|_\sigma + 16V_{\max} \cdot \delta. \end{aligned} \quad (54)$$

We apply the inequality in eq. (41) to eq. (54) with $a = \|\hat{O} - TQ\|_\sigma$, $b = 8V_{\max} \sqrt{\log N_\delta/n}$, and $c = \mathbb{E}[\|\hat{O} - TQ\|_n^2] + 16V_{\max} \cdot \delta$ we have. Hence we found

$$\|\hat{O} - TQ\|_\sigma^2 \leq (1 + \epsilon) \cdot \mathbb{E}[\|\hat{O} - TQ\|_n^2] + (1 + \epsilon)^2 \cdot 64V_{\max} \cdot \log N_\delta/(n \cdot \epsilon) + (1 + \epsilon) \cdot 18V_{\max} \cdot \delta, \quad (55)$$

which concludes the second step of the proof.

Finally, combining steps (i) and together, i.e., eq. (42) and eq. (55), we conclude that

$$\|\hat{O} - TQ\|_\sigma^2 \leq (1 + \epsilon)^2 \cdot \inf_{f \in \mathcal{F}} \mathbb{E}[\|f - TQ\|_n^2] + C_1 \cdot V_{\max}^2 \cdot \log N_\delta/(n \cdot \epsilon) + C_2 \cdot V_{\max} \cdot \delta + 2\beta G^2, \quad (56)$$

where C_1 and C_2 are two absolute constants. Moreover, since $Q \in \mathcal{F}$

$$\inf_{f \in \mathcal{F}} \mathbb{E}[\|f - TQ\|_n^2] \leq \sup_{Q \in \mathcal{F}} \left\{ \inf_{f \in \mathcal{F}} \mathbb{E}[\|f - TQ\|_n^2] \right\}, \quad (57)$$

which concludes the proof of theorem 3.3. \square

8. Appendix: Experimental Settings

In this section, we provide the experimental settings in detail.

8.1. Code

Our project is available at <https://sites.google.com/view/peer-cvpr2023/>.

8.2. Experimental Details

Our implementation of PEER coupled with CURL/DrQ is based on the CURL/DrQ codebase.

Computational resources. All experiments are conducted on two GPU servers. The first one has 3 Titan XP GPUs and Intel(R) Xeon(R) CPU E5-2640 v4 @ 2.40GHz. The second one has 4 Titan RTX GPUs and an Intel(R) Xeon(R) Gold 6137 CPU @ 3.90GHz. Each run for DMControl takes fifty hours to finish. For PyBullet, MuJoCo, and Atari tasks, it takes 5 hours to finish a run. For PyBullet and MuJoCo suites, we simultaneously launch 70 seeds. For the DMControl and Atari suites, we simultaneously run 15 random seeds.

How to plot fig. 1. Before computing the distinguishable representation discrepancy (DRD), the representation of Q -network is normalized as shown in theorem 7.1. Then we compute DRD in mini-batch samples. We compute the average DRD of mini-batch samples, and plot fig. 1 over five random seeds.

The source of data in table 2. The evaluation results of State SAC, PlaNet, Dreamer, SAC+AE, and CURL in table 2 are taken from the original CURL paper [6]. And the results of DrQ are taken from the original DrQ paper [9]. As for the data of DrQ-v2, we took from the authors' data (link: <https://github.com/facebookresearch/drqv2>) and presented the statistics in the same way as the rest of table 2. Note that the author only provides DrQ-v2 results over nine seeds.

The source of data in table 3. The evaluation results of Human, Random, OTRainbow, Eff.Rainbow, and CURL in table 3 are taken from the original CURL paper [6]. And the results of Eff. DQN and DrQ are taken from the original DrQ paper [9].

Data in fig. 4. We do not include the DrQ-v2 results in fig. 4 because DrQ is better than DrQ-v2 as shown in table 2.

Random seeds. If not otherwise specified, we evaluate each tested algorithm over 10 random seeds to ensure the reproducibility of our experiments. Also, we set all seeds fixed in our experiments.

Grid world. The grid world is shown in fig. 3a. If the agent arrives at S_T , it gets a reward of 10, and other states get a reward of 0. We present the remaining hyper-parameters for the grid world in table 4.

PyBullet. When we train the agent on the Pybullet suite, the agent starts by randomly collecting 25,000 states and actions for better exploration. Then we evaluate the agent for ten episodes every 5,000 timesteps. We take the average return of ten episodes as a key evaluation metric. To ensure a fair evaluation of the algorithms, we do not apply any exploration tricks during the evaluation phase (e.g. injecting noise into actions in TD3), because these exploration tricks may harm the performance of tested algorithms. The complete timesteps are 1 million. The results are reported over ten random seeds. For the hyper-parameter β of PEER, we take $5e - 4$ for every task.

For all algorithms except METD3, we use the author's implementation [28] or a commonly used public repository [46]. Our implementations of PEER and METD3 are based on TD3 implementation. To fairly evaluate our algorithm, we keep all the original TD3's hyper-parameters without any modification. For the hyper-parameter of METD3, we set the dropout rate equal to 0.1 as the author [42] did. The soft update style is adopted for METD3, PEER with $\eta = 0.005$. We summarize the hyper-parameter settings for the PyBullet suite in table 5.

MuJoCo. All experiments on MuJoCo are consistent with the PyBullet settings, except for the code of SAC used. We found that the performance of SAC [45] deteriorates on the MuJoCo suite. Therefore, we use the code of Stable-Baselines3¹ [60] for SAC implementation with the same hyper-parameters under PyBullet settings.

DMControl. We utilize the authors' implementation of CURL and DrQ without any further modification as we discussed. And we do not change the default hyper-parameters for CURL². For a fair comparison, we keep the hyper-parameters of PEER the same as CURL and DrQ. And the hyper-parameter $\beta = 5e - 4$ is kept in each environment. We summarize the hyper-parameter settings for the DMControl suite in table 6 and table 7.

Atari. Our implementation PEER is based on CURL³. For a fair comparison, we keep the hyper-parameters and settings of CURL the same as CURL. And the hyper-parameter $\beta = 5e - 4$ is kept in each environment. Check table 8 and table 9 for more details.

¹Code: <https://github.com/DLR-RM/stable-baselines3>

²Code: <https://github.com/MishaLaskin/curl>

³Code: https://github.com/aravindsrinivas/curl_rainbow

8.3. Pseudocode for PEER Loss

We provide PyTorch-like pseudocode for the PEER loss as follows.

```

1 def PE_loss_with_PEER(representation, Q, target_representation, target_Q, beta):
2     """
3     representation: shape = Batch_size * N, representation of critic
4     Q: shape = Batch_size * 1, current Q value
5     target_representation: shape = Batch_size * N, representation of critic_target
6     target_Q: shape = Batch_size * 1, target Q value (  $r + \mathcal{E}Q(s', a')$  )
7     beta: a small constant, controlling the regularization effectiveness of PEER
8     """
9     PEER_loss = torch.einsum('ij,ij->i', [representation, target_representation]).mean()
10    PE_loss = torch.nn.functional.mse_loss(Q, target_Q).mean()
11
12    loss = PE_loss + beta * PEER_loss
13    return loss

```

Listing 1. Pytorch-like pseudocode for the PEER loss

Hyper-parameter	Value
<i>Shared hyper-parameters</i>	
State space	integer: from 0 to 19
Action space	Discrete(4): up, down, left, right
Discount (γ)	0.99
Replay buffer size	10^5
Optimizer	Adam [61]
Learning rate for Q-network	1×10^{-4}
Number of hidden layers for all networks	2
Number of hidden units per layer	32
Activation function	ReLU
Mini-batch size	64
Random starting exploration time steps	10^3
Target smoothing coefficient (η)	0.005
Gradient Clipping	False
Exploration Method	Epsilon-Greedy
ϵ	0.1
Evaluation Episode	10
Number of Episodes	2000
<i>PEER</i>	
PEER coefficient (β)	5×10^{-4}

Table 4. Hyper-parameters settings for Grid World experiments

Hyper-parameter	Value
<i>Shared hyper-parameters</i>	
Discount (γ)	0.99
Replay buffer size	10^6
Optimizer	Adam [61]
Learning rate for actor	3×10^{-4}
Learning rate for critic	3×10^{-4}
Number of hidden layers for all networks	2
Number of hidden units per layer	256
Activation function	ReLU
Mini-batch size	256
Random starting exploration time steps	2.5×10^4
Target smoothing coefficient (η)	0.005
Gradient Clipping	False
Target update interval (d)	2
<i>TD3</i>	
Variance of exploration noise	0.2
Variance of target policy smoothing	0.2
Noise clip range	$[-0.5, 0.5]$
Delayed policy update frequency	2
<i>PEER</i>	
PEER coefficient (β)	5×10^{-4}
<i>SAC</i>	
Target Entropy	- dim of \mathcal{A}
Learning rate for α	1×10^{-4}

Table 5. Hyper-parameters settings for PyBullet and MuJoCo experiments

Hyper-parameter	Value
PEER coefficient (β)	5×10^{-4}
Discount γ	0.99
Replay buffer size	100000
Optimizer	Adam
Learning rate	1×10^{-4}
Learning rate ($f_\theta, \pi_\psi, Q_\phi$)	2×10^{-4} cheetah, run 1×10^{-3} otherwise
Convolutional layers	4
Number of filters	32
Activation function	ReLU
Encoder EMA η	0.05
Q function EMA (η)	0.01
Mini-batch size	512
Target Update interval (d)	2
Latent dimension	50
Initial temperature	0.99
Number of hidden units per layer (MLP)	1024
Evaluation episodes	10
Random crop	True
Observation rendering	(100,100)
Observation downsampling	(84,84)
Initial steps	1000
Stacked frames	3
Action repeat	2 finger, spin; walker, walk 8 cartpole, swingup 4 otherwise
$(\beta_1, \beta_2) \rightarrow (f_\theta, \pi_\psi, Q_\phi)$	(.9, .999)
$(\beta_1, \beta_2) \rightarrow (\alpha)$	(.9, .999)

Table 6. Hyper-parameters settings for PEER (coupled with CURL) DMControl experiments.

Hyper-parameter	Value
PEER coefficient (β)	5×10^{-4}
Replay buffer capacity	100000
Seed steps	1000
Main results minibatch size	512
Discount γ	0.99
Optimizer	Adam
Learning rate	10^{-3}
Critic target update frequency	2
Critic Q-function soft-update rate τ	0.01
Actor update frequency	2
Actor log stddev bounds	$[-10, 2]$
Init temperature	0.1

Table 7. Hyper-parameters settings for PEER (coupled with DrQ) DMControl experiments.

Hyper-parameter	Value
PEER coefficient (β)	5×10^{-4}
Random crop	True
Image size	(84, 84)
Data Augmentation	Random Crop (Train)
Replay buffer size	100000
Training frames	400000
Training steps	100000
Frame skip	4
Stacked frames	4
Action repeat	4
Replay period every	1
Q network: channels	32, 64
Q network: filter size	$5 \times 5, 5 \times 5$
Q network: stride	5, 5
Q network: hidden units	256
Momentum (EMA for CURL) τ	0.001
Non-linearity	ReLU
Reward Clipping	$[-1, 1]$
Multi step return	20
Minimum replay size for sampling	1600
Max frames per episode	108K
Update	Distributional Double Q
Target Network Update Period	every 2000 updates
Support-of-Q-distribution	51 bins
Discount γ	0.99
Batch Size	32
Optimizer	Adam
Optimizer: learning rate	0.0001
Optimizer: β_1	0.9
Optimizer: β_2	0.999
Optimizer ϵ	0.000015
Max gradient norm	10
Exploration	Noisy Nets
Noisy nets parameter	0.1
Priority exponent	0.5
Priority correction	$0.4 \rightarrow 1$
Hardware	GPU

Table 8. Hyper-parameters used for Atari100K PEER (coupled with CURL) experiments.

Hyperparameter	Value
PEER coefficient (β)	5×10^{-4}
Data augmentation	Random shifts and Intensity
Grey-scaling	True
Observation down-sampling	84×84
Frames stacked	4
Action repetitions	4
Reward clipping	$[-1, 1]$
Terminal on loss of life	True
Max frames per episode	108k
Update	Double Q
Dueling	True
Target network: update period	1
Discount factor	0.99
Minibatch size	32
Optimizer	Adam
Optimizer: learning rate	0.0001
Optimizer: β_1	0.9
Optimizer: β_2	0.999
Optimizer: ϵ	0.00015
Max gradient norm	10
Training steps	100k
Evaluation steps	125k
Min replay size for sampling	1600
Memory size	Unbounded
Replay period every	1 step
Multi-step return length	10
Q network: channels	32, 64, 64
Q network: filter size	$8 \times 8, 4 \times 4, 3 \times 3$
Q network: stride	4, 2, 1
Q network: hidden units	512
Non-linearity	ReLU
Exploration	ϵ -greedy
ϵ -decay	5000

Table 9. Hyper-parameters used for Atari100K PEER (coupled with DrQ algorithm) experiments.

9. Appendix: Experimental Suites

The experimental suites we use are Bullet [29], MuJoCo[31], DMcontrol[30], and Atari[32]. We show the environments of bullet, MuJoCo, DMControl, and Atari in fig. 6, fig. 7, fig. 8, fig. 9, and fig. 10, respectively.

Besides, We list the state and action information for the four suites in table 10, table 11, table 12, and table 13. respectively.

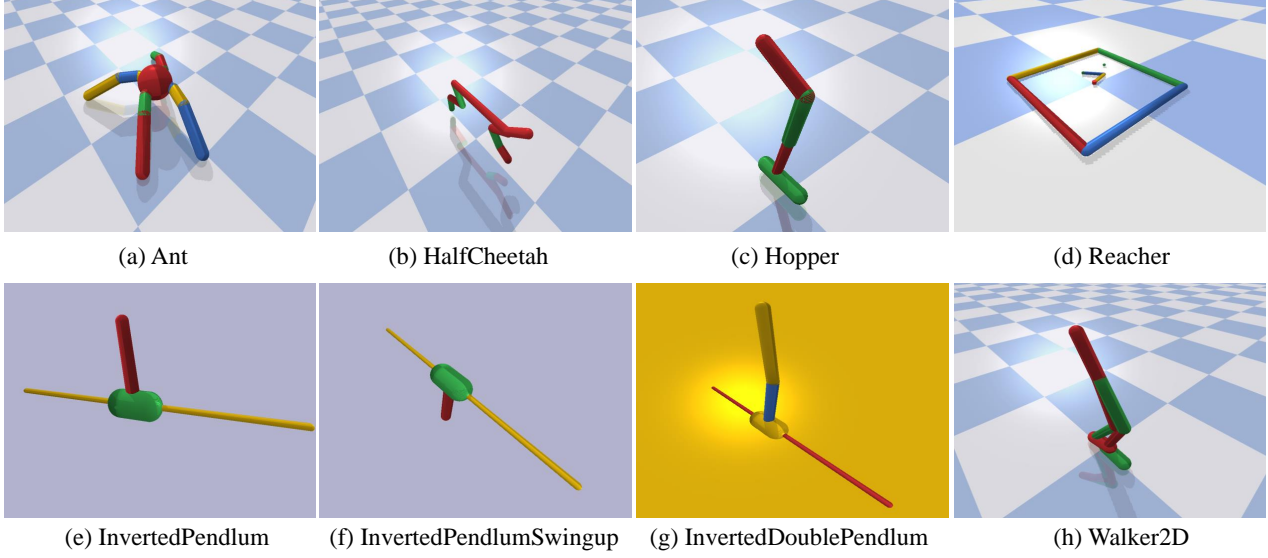


Figure 6. Images for PyBullet suite used in our experiments. The states for this suite are vectors.

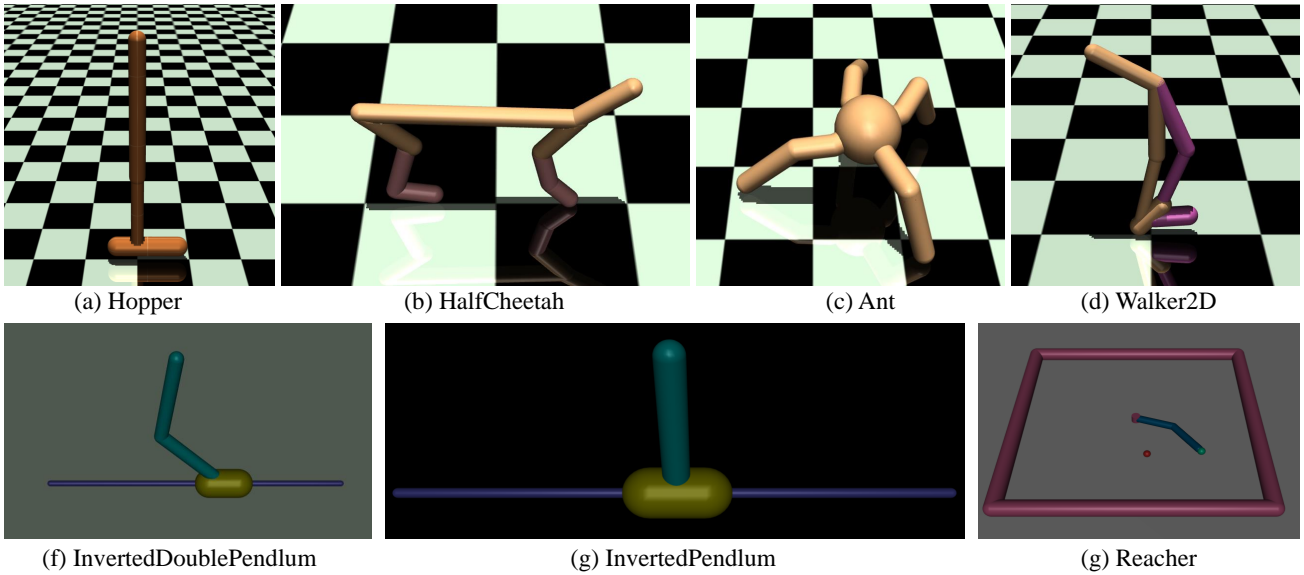


Figure 7. Images for MuJoCo suite used in our experiments. The states for this suite are vectors.

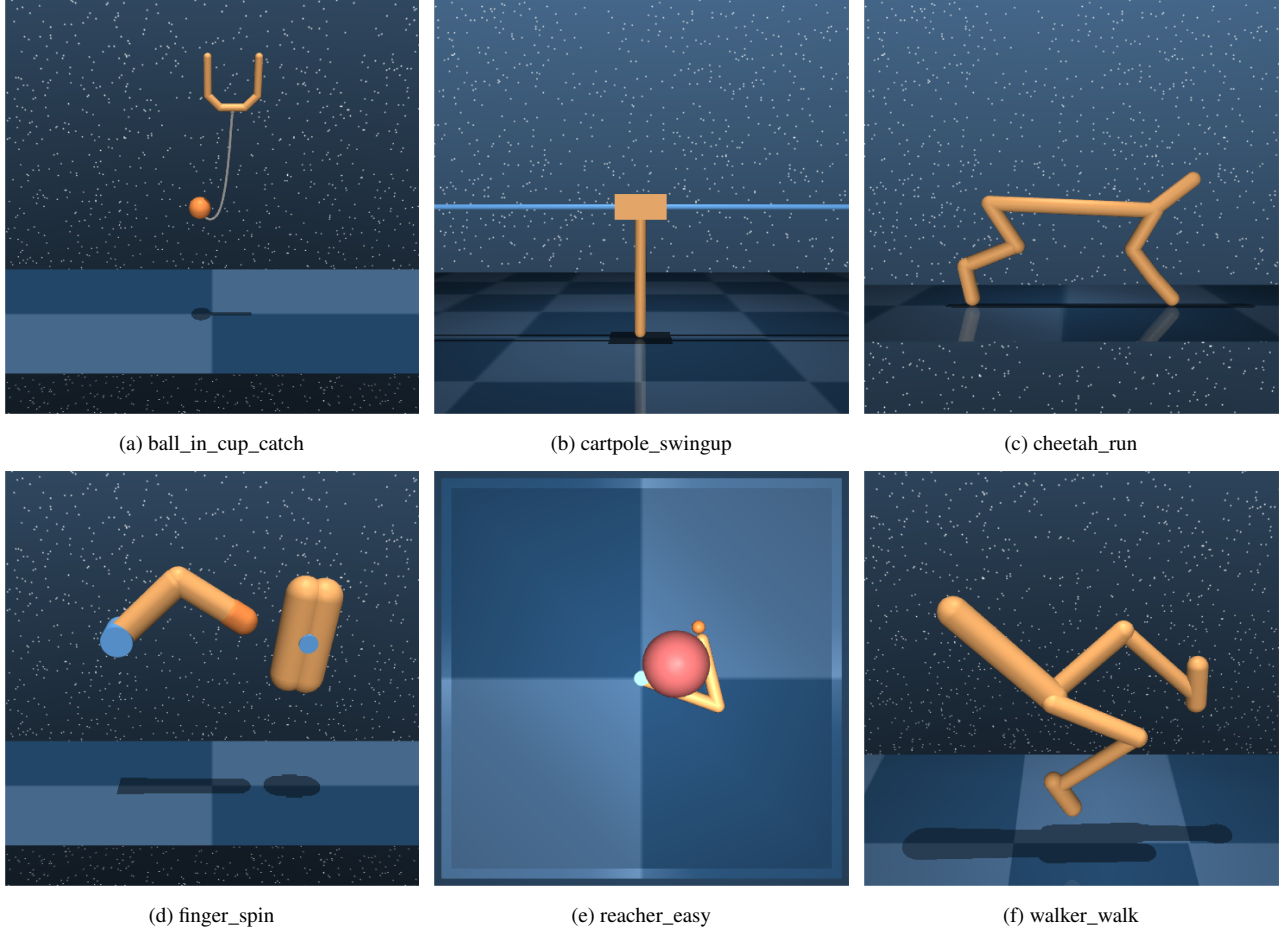


Figure 8. Images for DMControl suites used in our experiments. Each image is a frame of a specific DMControl suite.

Env	State Dimension	Action Dimension
InvertedPendulum	5	Continuous(1)
InvertedDoublePendulum	9	Continuous(1)
InvertedPendulumSwingup	5	Continuous(1)
Reacher	9	Continuous(2)
Walker2D	22	Continuous(6)
HalfCheetah	26	Continuous(6)
Ant	28	Continuous(8)
Hopper	15	Continuous(3)

Table 10. State dimension and action space for Bullet suite. Continuous(x) means the action space is continuous with dimension x .

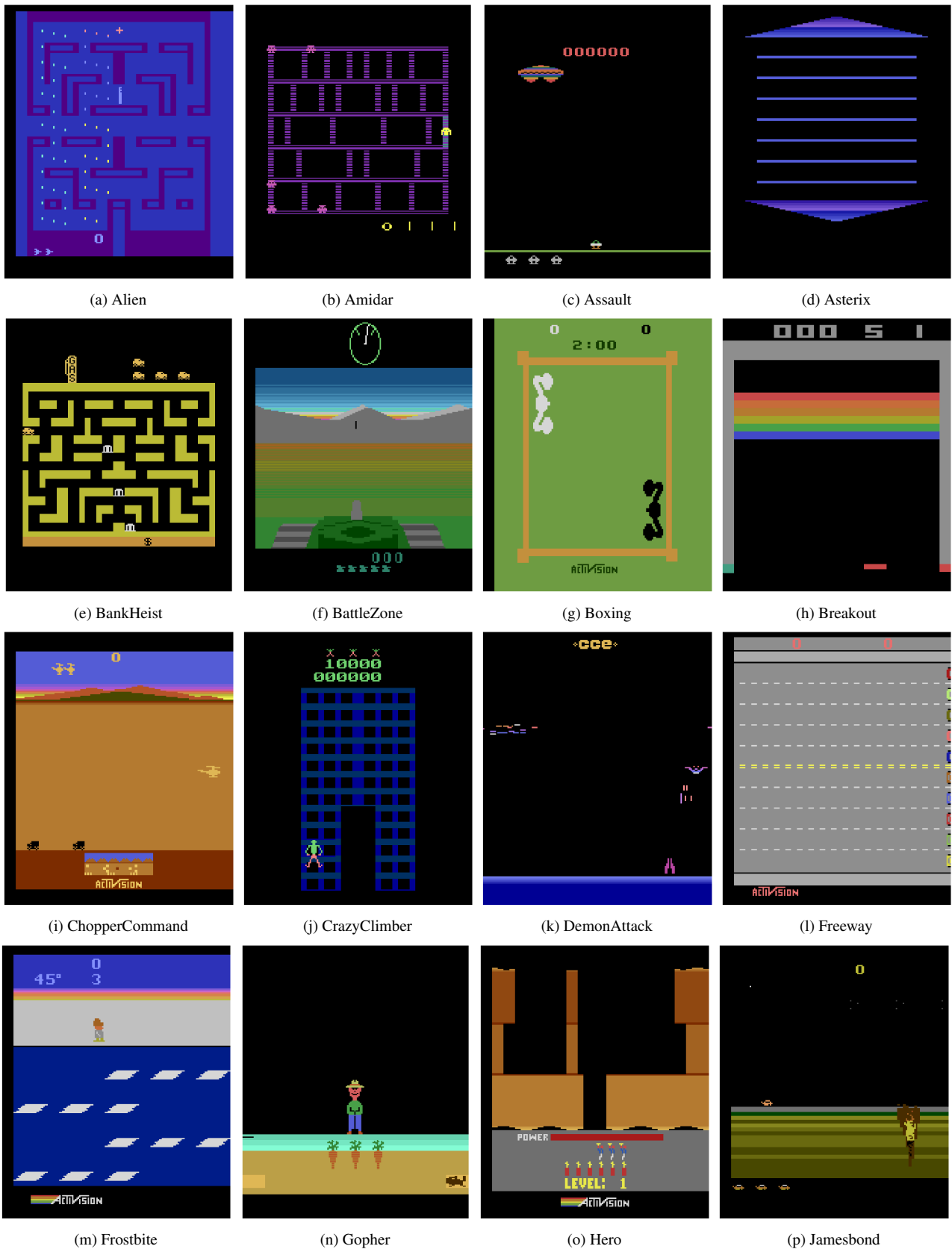


Figure 9. Images for Atari100k suites used in our experiments. Each image is a frame of a specific Atari game.

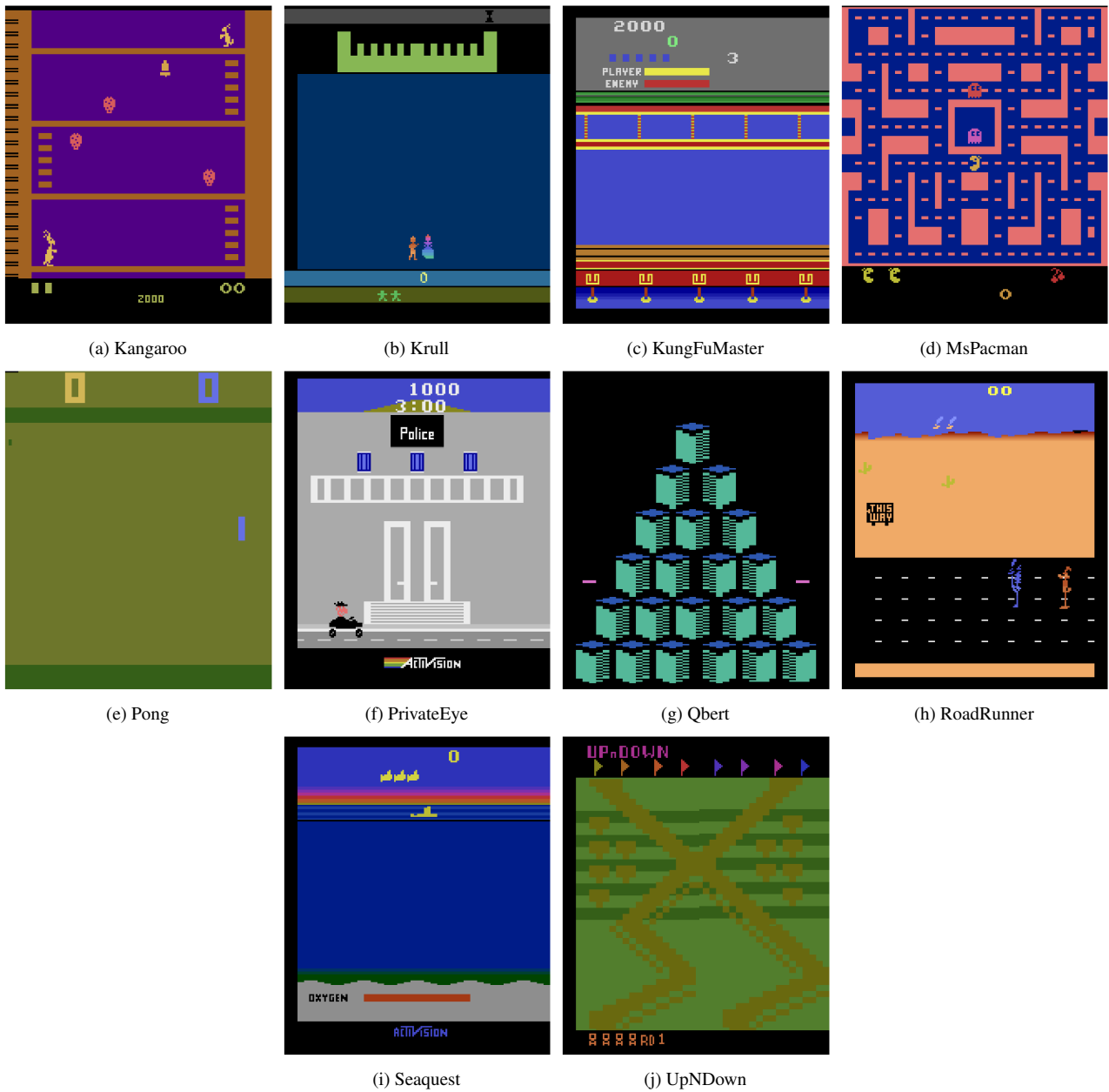


Figure 10. Images for Atari100k suites used in our experiments (continuation of fig. 9). Each image is a frame of a specific Atari game.

Env	State Dimension	Action Dimension
Reacher	11	Continuous(2)
Walker2d	17	Continuous(6)
HalfCheetah	17	Continuous(6)
Swimmer	8	Continuous(2)
Ant	111	Continuous(8)
Hopper	11	Continuous(3)
InvertedPendulum	4	Continuous(1)
InvertedDoublePendulum	11	Continuous(1)

Table 11. State space and action space for MuJoCo suite. Continuous(x) means the action space is continuous with dimension x .

Domain	Tasks	State Space	Action Space
ball_in_cup	catch	(3, 100, 100)	Continuous(2)
cartpole	balance	(3, 100, 100)	Continuous(1)
cartpole	balance_sparse	(3, 100, 100)	Continuous(1)
cartpole	swingup	(3, 100, 100)	Continuous(1)
cartpole	swingup_sparse	(3, 100, 100)	Continuous(1)
cheetah	run	(3, 100, 100)	Continuous(6)
finger	spin	(3, 100, 100)	Continuous(2)
finger	turn_easy	(3, 100, 100)	Continuous(2)
finger	turn_hard	(3, 100, 100)	Continuous(2)
hopper	hop	(3, 100, 100)	Continuous(4)
hopper	stand	(3, 100, 100)	Continuous(4)
pendulum	swingup	(3, 100, 100)	Continuous(1)
reacher	easy	(3, 100, 100)	Continuous(2)
reacher	hard	(3, 100, 100)	Continuous(2)
walker	stand	(3, 100, 100)	Continuous(6)
walker	walk	(3, 100, 100)	Continuous(6)

Table 12. State space and action space for DMControl suite. Continuous(x) means the action space is continuous with dimension x .

Game	State Space	Action Space
Alien	(210, 160, 3)	Discrete(18)
Amidar	(210, 160, 3)	Discrete(10)
Assault	(210, 160, 3)	Discrete(7)
Asterix	(210, 160, 3)	Discrete(9)
BankHeist	(210, 160, 3)	Discrete(18)
BattleZone	(210, 160, 3)	Discrete(18)
Boxing	(210, 160, 3)	Discrete(18)
Breakout	(210, 160, 3)	Discrete(4)
ChopperCommand	(210, 160, 3)	Discrete(18)
CrazyClimber	(210, 160, 3)	Discrete(9)
DemonAttack	(210, 160, 3)	Discrete(6)
Freeway	(210, 160, 3)	Discrete(3)
Frostbite	(210, 160, 3)	Discrete(18)
Gopher	(210, 160, 3)	Discrete(8)
Hero	(210, 160, 3)	Discrete(18)
Jamesbond	(210, 160, 3)	Discrete(18)
Kangaroo	(210, 160, 3)	Discrete(18)
Krull	(210, 160, 3)	Discrete(18)
KungFuMaster	(210, 160, 3)	Discrete(14)
MsPacman	(210, 160, 3)	Discrete(9)
Pong	(210, 160, 3)	Discrete(6)
PrivateEye	(210, 160, 3)	Discrete(18)
Qbert	(210, 160, 3)	Discrete(6)
RoadRunner	(210, 160, 3)	Discrete(18)
Seaquest	(210, 160, 3)	Discrete(18)
UpNDown	(210, 160, 3)	Discrete(6)

Table 13. State space and action space for Atari suite. Discrete(x) means the action space is discrete with x actions.

10. Appendix: Additional Experimental Results

In this section, we provide additional experimental results. PEER works by adding a regularization term to backbone DRL algorithms. Thus, the comparison with the backbone algorithm of PEER naturally becomes an ablation experiment. We provide more experiments to demonstrate the effectiveness of PEER.

10.1. Experiments on MuJoCo Suite

We present the performance of PEER on the MuJoCo suite in table 14. The results show that our proposed PEER outperforms or matches the compared algorithms in 5 out of 7 MuJoCo environments. Compared with its backbone algorithm TD3, PEER surpasses it in 6 out of 7 environments.

Algorithm	Ant	HalfCheetah	Hopper	InvDouPen	InvPen	Reacher	Walker
PEER	5386 ± 493	10832 ± 501	3424 ± 180	7470 ± 3721	1000 ± 0	-4 ± 1	4223 ± 655
TD3	5102 ± 787	10858 ± 637	3163 ± 367	7312 ± 3653	1000 ± 0	-4 ± 1	3762 ± 956
METD3	2256 ± 431	5696 ± 1740	804 ± 71	7815 ± 0	912 ± 71	-8 ± 3	2079 ± 1096
SAC	4233 ± 806	10482 ± 959	2666 ± 320	9358 ± 0	1000 ± 0	-4 ± 0	4187 ± 304

Table 14. The average return of the last ten evaluations over ten random seeds. The maximum average returns are bolded. PEER outperforms or matches the other tested algorithms in 5 out of 7 environments.

In table 15, we show comparisons with model-free algorithm REDQ [62] on pybullet suite.

Algo	Ant	Hopper	Walker
PEER	5386 ± 493	3424 ± 180	4223 ± 655
REDQ	3900 ± 890	2656 ± 759	4211 ± 524

Table 15. Average return for PEER and REDQ. PEER surpasses REDQ on all tested tasks. The REDQ results are obtained using the authors’ implementation and are reported over 20 trials.

10.2. Combination with Model-based Algorithm

In table 16, we show comparisons with model-based methods algorithms TDMPC and Dreamer-v2 on DMControl suites. In table 17, we show comparisons with model-based algorithms Dreamer-v2. Note that the data we take directly from the authors’ dreamer-v2 codebase (<https://github.com/danijar/dreamerv2/tree/main/scores>), the amount of data they use is 1000k, which is 10 times more than our PEER. The PEER scores in table 17 are taken as the largest of PEER+DrQ and PEER+CURL.

500K Step Scores	Finger, Spin	Cartpole, Swingup	Reacher, Easy	Cheetah, run	Walker, Walk	Ball_in_cup, Catch
PEER + CURL	864 ± 160	866 ± 17	980 ± 3	732 ± 41	946 ± 17	971 ± 5
Dreamer-V2	386 ± 83	853 ± 15	876 ± 60	610 ± 117	934 ± 16	792 ± 300
100K Step Scores						
PRER +TDMPC	772 ± 107	848 ± 25	841 ± 115	636 ± 35	876 ± 41	937 ± 96
TDMPC	943 ± 59	770 ± 70	628 ± 105	222 ± 88	577 ± 208	933 ± 24
Dreamer-V2	414 ± 93	697 ± 176	633 ± 248	501 ± 146	705 ± 232	693 ± 335

Table 16. Comparison with model-based methods. PEER outperforms Dreamer-v2 on 12 out of 12 tasks. PEER (combined with TDMPC [63]) outperforms TDMPC by on 5 out of 6 tasks.

Game	PEER	Dreamer-V2	MuZero
Alien	1218.9	384.1	530.0
Amidar	185.2	29.8	38.8
Assault	721.0	433.4	500.1
Asterix	918.2	330.6	1734.0
BHeist	78.6	127.1	192.5
BZone	15727.3	4200.0	7687.5
Boxing	14.5	37.7	15.1
Breakout	8.5	1.5	48.0
ChpCmd	1451.8	687.5	1350.0
CzClmr	18922.7	25232.5	56937.0
DmAttack	1236.7	182.9	3527.0
Freeway	30.4	11.6	21.8
Frostbite	2151.0	302.5	255.0
Gopher	681.8	820.2	1256.0
Hero	7499.9	2185.0	3095.0
Jbond	414.1	81.2	87.5
Kangaroo	1148.2	150.0	62.5
Krull	5444.7	3853.8	4890.8
KFMaster	15439.1	12420.3	18813.0
MsPacman	1768.4	647.9	1265.6
Pong	-9.5	-18.3	-6.7
PriEye	3207.7	188.8	56.3
Qbert	2197.7	318.6	3952.0
RdRunner	10697.3	3622.5	2500.0
Squest	538.5	356.0	208.0
UpNDown	7680.9	8025.1	2896.9

Table 17. PEER outperforms Dreamer-v2 and Muzero on 21 and 16 games of Atari26 where Dreamer-v2 even uses 10 times the data of PEER. Note that the data we take directly from the authors’ dreamer-v2 codebase, the amount of data for Dreamer-V2 they use is 1000k, which is 10 times more than our PEER.

10.3. Various β for Performance Improvement

Fine-tuning for hyper-parameters probably improves the performance of PEER. To see this, we select 7 Atari environments to investigate the effect of fine-tuning β , where PEER (coupled with CURL) achieves SOTA performance. We present the results in fig. 11. There is no one value taken that is significantly better than the other. We see that large β ($=1e-2$) may result in the failure of learning (on Freeway game) but may also bring the best performance improvements (on Kangaroo game). Overall, fine-tuning the hyper-parameter β may improve the empirical performance by a large margin.

10.4. Performance curves on DMControl Tasks

We present the performance curves of PEER on a total of 16 DMControl environments in fig. 12 and fig. 13. We run 10 seeds in each environment.

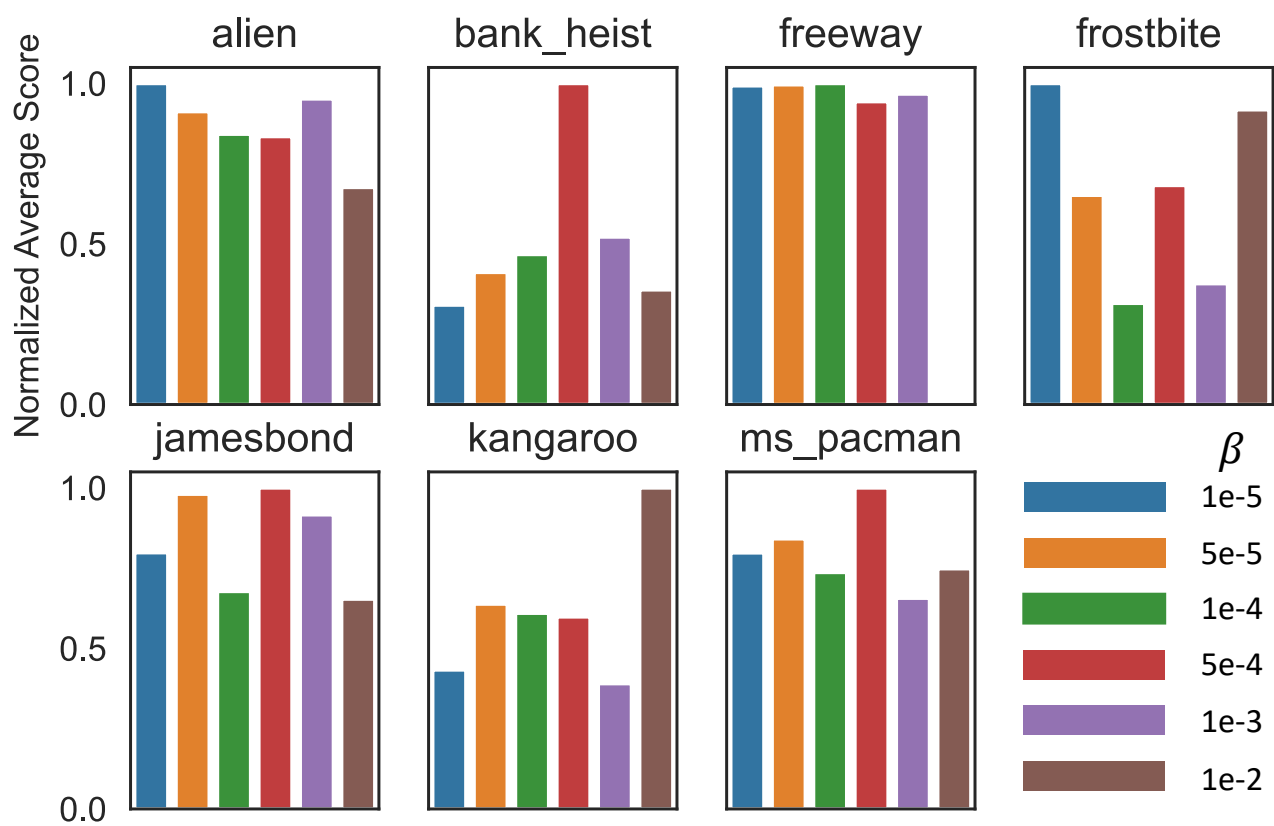


Figure 11. The average scores normalized by the max average score on the 7 Atari games for selected 6 hyper-parameter β . From the experiments, we can see that fine-tuning the β may result in performance improvements.

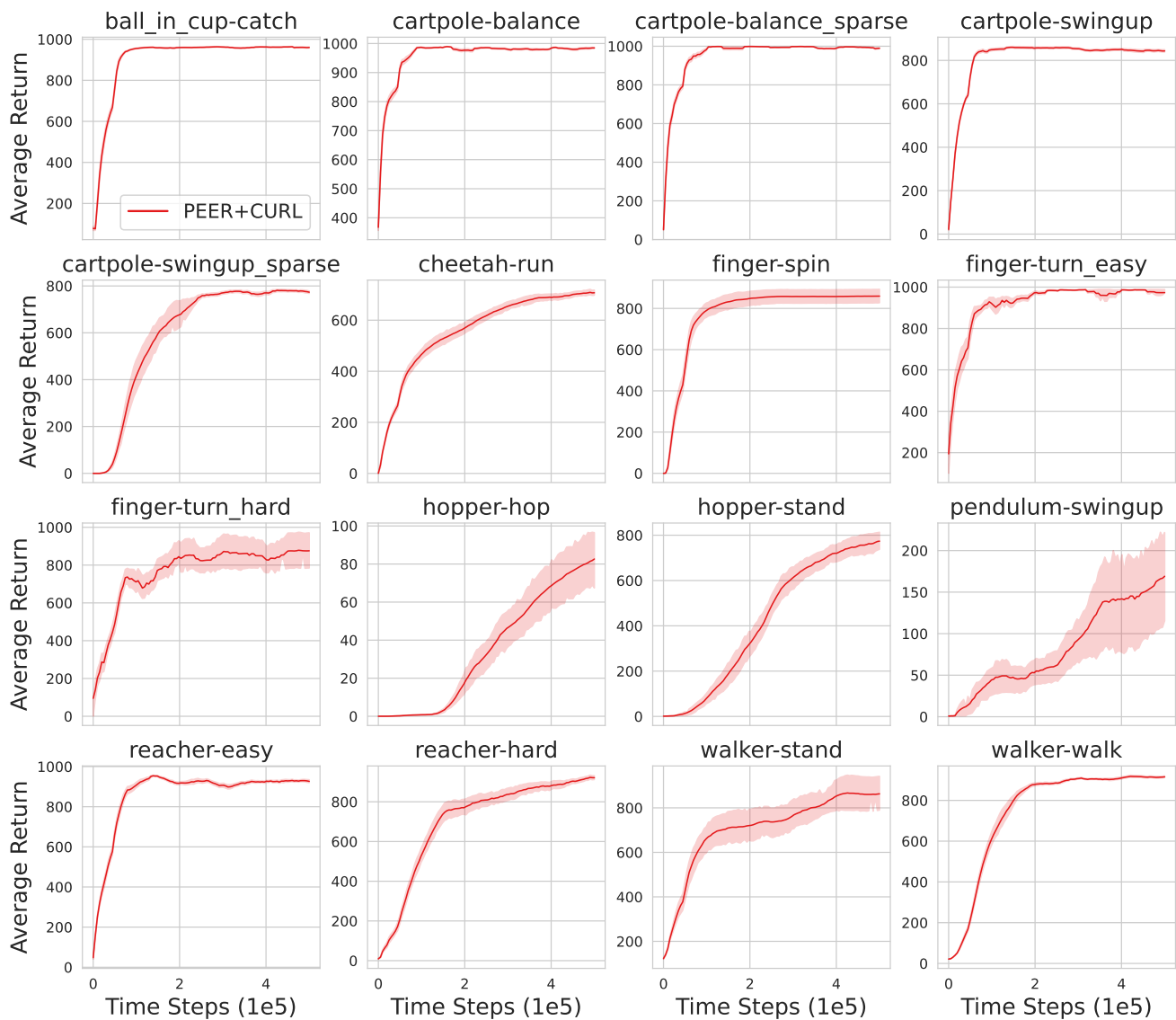


Figure 12. Performance curves for PEER (coupled with CURL) on DMControl suite. The shaded region represents half the standard deviation of the average evaluation over 10 seeds. The curves are smoothed by moving average.

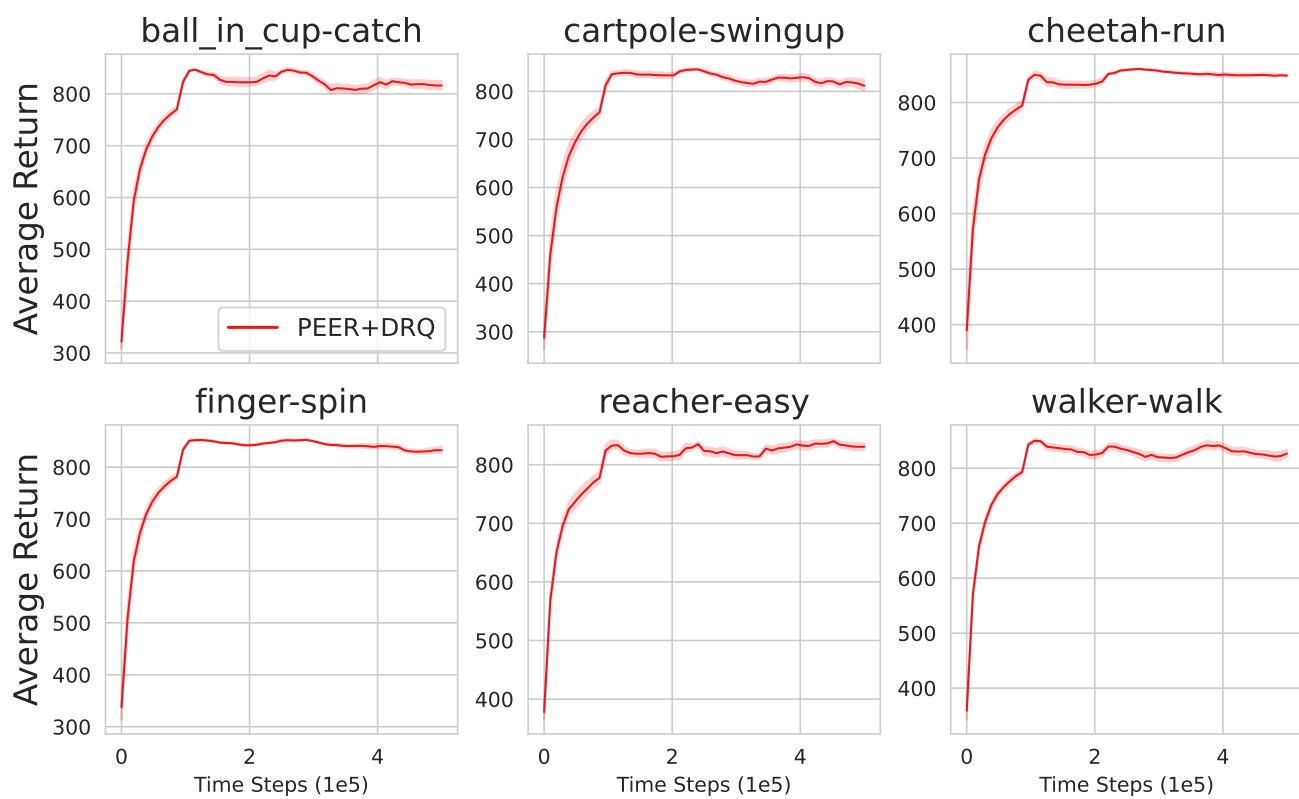


Figure 13. Performance curves for PEER (coupled with DrQ) on DMControl suite. The shaded region represents half the standard deviation of the average evaluation over 10 seeds. The curves are smoothed by moving average.