Appendix A.

A.1. The relation of returns (performance guarantee) under k-step branched rollouts in Markov decision processes

In this section, we analyze the relation of a true return and a model return in an MDP, which is defined by a tuple $\langle \mathcal{S}, \mathcal{A}, r, \gamma, p_{\mathrm{st}} \rangle$. Here, $\mathcal{S}$ is a set of states, $\mathcal{A}$ is a set of actions, $p_{\mathrm{st}} = \mathcal{S} \times \mathcal{S} \times \mathcal{A} \to [0, 1]$ is the state transition probability, $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is a reward function and $\gamma \in [0, 1)$ is a discount factor. At time step t, the state transition probability and reward function are used as $p(s_t|s_{t-1}, a_{t-1})$ and $r(s_t,\ a_t)$, respectively. The true return is defined as $\mathbb{E}_{a\sim\pi, s\sim p}\left[R = \sum_{t=0}^{\infty} \gamma^t r_t\right]$, where π is the agent's current policy. In addition, the model return is defined by $\mathbb{E}_{(a,s)\sim m_b(\pi, p_\theta, \mathcal{D}_{\mathrm{env}})}[R]$, where $m(\pi, p_\theta, \mathcal{D}_{\mathrm{env}})$ is the state-action visitation probability based upon an abstract model-based rollout method, which can be calculated on the basis of the π, p_θ, and $\mathcal{D}_{\mathrm{env}}$. p_θ is the predictive model for the next state. $\mathcal{D}_{\mathrm{env}}$ is the dataset in which the real trajectories collected by a data collection policy $\pi_{\mathcal{D}}$ is stored.

We analyze the relation of the returns, which takes the form of

$$\mathbb{E}_{a\sim\pi, s\sim p}[R] \geq \mathbb{E}_{(a,s)\sim m(\pi, p_\theta, \mathcal{D}_{\mathrm{env}})}[R] - C(\epsilon_m, \epsilon_\pi),$$

where $C(\epsilon_m, \epsilon_\pi)$ is the discrepancy between the returns, which can be expressed as the function of two error quantities ϵ_m and ϵ_π. Here, $\epsilon_m = \max_t \mathbb{E}_{a_t\sim\pi_{\mathcal{D}}, s_t\sim p}\left[D_{\mathrm{TV}}\left(p(s_{t+1}|s_t, a_t)||p_\theta(s_{t+1}|s_t, a_t)\right)\right]$ and $\epsilon_\pi = \max_{s_t} D_{\mathrm{TV}}\left(\pi(a_t|s_t)||\pi_{\mathcal{D}}(a_t|s_t)\right)$. In addition, we define the upper bounds of the reward scale as $r_{\max} > \max_{s,a}|r(s, a)|$. Note that, in this section, to discuss the MDP case, we are overriding the definition of the variables and functions that were defined for the POMDP case in the main content.

Janner et al. (2019) analyzed the relations of the returns under the full-model-based rollout and that under the branched rollout in the MDP:

Theorem 4.1. in Janner et al. (2019). *Under the full-model-based rollout in the MDP, the following inequality holds,*

$$\mathbb{E}_{a\sim\pi, s\sim p}[R] \geq \mathbb{E}_{(a,s)\sim m_f(\pi, p_\theta, \mathcal{D}_{\mathrm{env}})}[R] - 2r_{\max}\left\{\frac{\gamma}{(1-\gamma)^2}(\epsilon_m + 2\epsilon_\pi) + \frac{2}{(1-\gamma)}\epsilon_m\right\}, \quad (2)$$

where $\mathbb{E}_{(a,s)\sim m_f(\pi, p_\theta, \mathcal{D}_{env})}[R]$ is the model return under the full-model-based rollout. $m_f(\pi_\phi, p_\theta, \mathcal{D}_{env})$ is the state-action visitation probability under the full model-based rollout method.

Theorem 4.2. in Janner et al. (2019). *Under the branched rollout in the MDP, the following inequality holds,*

$$\mathbb{E}_{a\sim\pi, s\sim p}[R] \geq \mathbb{E}_{(a,s)\sim m_b(\pi, p_\theta, \mathcal{D}_{\mathrm{env}})}[R] - 2r_{\max}\left\{\frac{\gamma^{k+1}}{(1-\gamma)^2}\epsilon_\pi + \frac{\gamma^k + 2}{(1-\gamma)}\epsilon_\pi + \frac{k}{1-\gamma}(\epsilon_m + 2\epsilon_\pi)\right\}, \quad (3)$$

where $\mathbb{E}_{(a,s)\sim m_b(\pi, p_\theta, \mathcal{D}_{env})}[R]$ is the model return under the branched rollout. $m_b(\pi_\phi, p_\theta, \mathcal{D}_{env})$ is the state-action visitation probability under the branched rollout method.

The notion of $m_b(\pi, p_\theta, \mathcal{D}_{\mathrm{env}})$ in the analyses in Janner et al. (2019) and our case are summarized in Figure 10. In the branched rollout method, the real trajectories are uniformly sampled from $\mathcal{D}_{\mathrm{env}}$, and then starting from the sampled trajectories, k-step model-based

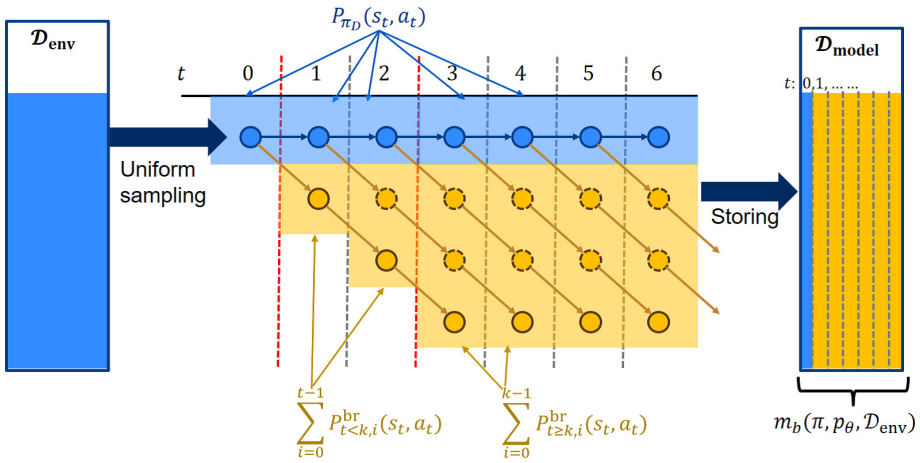

Figure 10: The notion of $m_b(\pi_\phi, p_\theta, \mathcal{D}_{\text{env}})$. The figure shows the case of the branched rollout method with $k = 3$. The blue nodes represent real trajectories contained in $\mathcal{D}_{\text{env}}$, and the yellow nodes represent fictitious trajectories generated by k-step model-based rollouts with π and p_θ. The fictitious trajectories are stored in D_{model}. The distribution of the trajectories stored in D_{model} is used as $m_b(\pi, p_\theta, \mathcal{D}_{\text{env}})$. **In the analysis (derivation of Theorem 4.2) in Janner et al. (2019), fictitious trajectories of dashed yellow nodes are not stored in $\mathcal{D}_{\text{model}}$, and thus the state-action visitation probability at each time step is calculated based solely on a single model-based rollout factor. To contrast, in our analysis, these fictitious trajectories are stored in $\mathcal{D}_{\text{model}}$, and thus the state-action visitation probability at each time step is calculated on the basis of multiple model-based rollout factors.**

rollouts under π and p_θ are run. Then, the fictitious trajectories generated by the branched rollout are stored in a model dataset D_{model} [3]. The distribution of the trajectories stored in D_{model} is used as $m_b(\pi, p_\theta, \mathcal{D}_{\text{env}})$. In Figure 10, $p_{\pi_\mathcal{D}}(s_t, a_t)$ is the state-action visitation probability under p_{st} and π_D. This can be used for the initial state distribution for k-steps model-based rollouts since the real trajectories are uniformly sampled from $\mathcal{D}_{\text{env}}$. In addition, $p^{\text{br}}_{t<k,i}(s_t, a_t)$ and $p^{\text{br}}_{t\geq k,i}(s_t, a_t)$ are the state-action visitation probabilities that the i-th yellow fictitious trajectories (nodes) from the bottom at t follow.

In the derivation of Theorem 4.2 (more specifically, the proof of Lemma B.4) in Janner et al. (2019), important premises are not properly taken into consideration. In the derivation, state-action visitation probabilities under the branched rollout are affected only by a single model-based rollout factor (see Figure 10). For example, a state-action visitation probability at t (s.t. $t > k$) is affected only by the model-based rollout branched from real trajectories at $t - k$ (i.e., $p^{\text{br}}_{t\geq k,0}(s_t, a_t)$). However, state-action visitation probabilities (except for ones at $t = 0$ and $t = 1$) should be affected by multiple past model-based rollouts.

3. Here, when the trajectories are stored in D_{model}, the states in the trajectories are augmented with time step information to deal with the state transition depending on the time step.

For example, a state-action visitation probability at t (s.t. $t > k$) should be affected by the model-based rollout branched from real trajectories at $t-k$ and ones from $t-k+1$ to $t-1$ (i.e., $p^{\mathrm{br}}_{t\geq k,0}(s_t,a_t)...,p^{\mathrm{br}}_{t\geq k,k-1}(s_t,a_t)$). In addition, in their analysis, they consider that k is an element of the set of non-negative integers. However, if $k=0$, the fictitious trajectories for $\mathcal{D}_{\mathrm{model}}$ are not generated, and the distribution of trajectories in $\mathcal{D}_{\mathrm{model}}$ cannot be built. Therefore, k's value should not be 0 (it should be an element of the set of non-zero natural numbers $\mathbb{N}_{>0}$). These oversights of important premises in their analysis induce a large mismatch between those for their theorem (Theorem 4.2) and those made for the actual implementation of the branched rollout (lines 5–8 in Algorithm 2 in Janner et al. (2019)).

Hence, we will newly analyze the relation of the returns under the branched rollout method, considering the aforementioned premises more properly. Concretely, we consider the multiple model-based rollout factors for $m_b(\pi,p_\theta,\mathcal{D}_{\mathrm{env}})$ (See Figure 10) [4]. With this consideration, we define the model-return under the branched rollout $\mathbb{E}_{(a,s)\sim m_b(\pi,p_\theta,\mathcal{D}_{\mathrm{env}})}[R]$ as:

$$\begin{aligned}
\mathbb{E}_{(a,s)\sim m_b(\pi,p_\theta,\mathcal{D}_{\mathrm{env}})}[R] &= \sum_{s_0,a_0} p_{\pi_\mathcal{D}}(s_0,a_0)r(s_0,a_0) + \sum_{t=1}^{k-1}\sum_{s_t,a_t}\gamma^t p^{\mathrm{br}}_{t<k}(s_t,a_t)r(s_t,a_t) \\
&\quad + \sum_{t=k}^{\infty}\sum_{s_t,a_t}\gamma^t p^{\mathrm{br}}_{t\geq k}(s_t,a_t)r(s_t,a_t) \qquad (4)\\
p^{\mathrm{br}}_{t<k}(s_t,a_t) &= \frac{1}{t}\sum_{i=0}^{t-1} p^{\mathrm{br}}_{t<k,i}(s_t,a_t) \qquad (5)\\
p^{\mathrm{br}}_{t\geq k}(s_t,a_t) &= \frac{1}{k}\sum_{i=0}^{k-1} p^{\mathrm{br}}_{t\geq k,i}(s_t,a_t) \qquad (6)\\
p^{\mathrm{br}}_{t<k,i}(s_t,a_t) &= \sum_{s_i,\ldots,s_{t-1}}\sum_{a_i,\ldots,a_{t-1}} p_{\pi_\mathcal{D}}(s_i)\Pi_{j=i}^{t-1}p_\theta(s_{j+1}|s_j,a_j)\pi(a_j|s_j) \qquad (7)\\
p^{\mathrm{br}}_{t\geq k,i}(s_t,a_t) &= \sum_{s_{t-k+i},\ldots,s_{t-1}}\sum_{a_{t-k+i},\ldots,a_{t-1}} \\
&\quad p_{\pi_\mathcal{D}}(s_{t-k+i})\Pi_{j=t-k+i}^{t-1}p_\theta(s_{j+1}|s_j,a_j)\pi(a_j|s_j) \qquad (8)
\end{aligned}$$

Here, $p_{\pi_\mathcal{D}}$ is the state visitation probability under $\pi_\mathcal{D}$ and p_{st}. The ones modified from Janner et al. (2019) are highlighted in red. In the remaining paragraphs in this section, we will derive the new theorems for the relation of the returns under this definition of the model return.

Before starting the derivation of our theorem, we introduce a useful lemma.

Lemma 1 *Assume that the rollout process in which the policy and dynamics can be switched to other ones at time step t_{sw}. Letting two probabilities be p_1 and p_2, for $1 \leq t' \leq t_{sw}$, we assume that the dynamics distributions are bounded as* $\epsilon_{m,pre} = \max_{t'} E_{s\sim p_1}\left[D_{TV}(p_1(s_{t'}|s_{t'-1},a_{t'-1})||p_2(s_{t'}|s_{t'-1},a_{t'-1}))\right]$. *In addition, for $t_{sw} < t' \leq t$, we assume that the dynamics distributions are bounded as*

4. Considering the discussion in the last paragraph, we also limit the range of k as $k \in \mathbb{N}_{>0}$ in our analysis.

$\epsilon_{m,post} = \max_{t'} E_{s\sim p_1}\left[D_{TV}(p_1(s_{t'}|s_{t'-1}, a_{t'-1})||p_2(s_{t'}|s_{t'-1}, a_{t'-1}))\right]$. *Likewise, the policy divergence is bounded by* $\epsilon_{\pi,pre}$ *and* $\epsilon_{\pi,post}$. *Then, the following inequality holds*

$$\sum_{s_t,a_t} |p_1(s_t, a_t) - p_2(s_t, a_t)| \leq 2(t - t_{sw})(\epsilon_{m,post} + \epsilon_{\pi,post}) + 2t_{sw}(\epsilon_{m,pre} + \epsilon_{\pi,pre}) \qquad (9)$$

Proof The proof is done in a similar manner to those of Lemma B.1 and B.2 in Janner et al. (2019).

$$\begin{aligned}
&\sum_{s_t,a_t} |p_1(s_t, a_t) - p_2(s_t, a_t)| \\
&= \sum_{s_t,a_t} |p_1(s_t)p_1(a_t|s_t) - p_2(s_t)p_2(a_t|s_t)| \\
&= \sum_{s_t,a_t} |p_1(s_t)p_1(a_t|s_t) - p_1(s_t)p_2(a_t|s_t) + (p_1(s_t) - p_2(s_t))p_2(a_t|s_t)| \\
&\leq \sum_{s_t,a_t} p_1(s_t)\,|p_1(a_t|s_t) - p_2(a_t|s_t)| + \sum_{s_t} |p_1(s_t) - p_2(s_t)| \\
&\leq \sum_{s_t,a_t} p_1(s_t)\,|p_1(a_t|s_t) - p_2(a_t|s_t)| + \sum_{s_t,a_{t-1}} |p_1(s_t, a_{t-1}) - p_2(s_t, a_{t-1})| \\
&= \sum_{s_t,a_t} p_1(s_t)\,|p_1(a_t|s_t) - p_2(a_t|s_t)| \\
&\quad + \sum_{s_t,a_{t-1}} |p_1(a_{t-1})p_1(s_t|a_{t-1}) - p_1(a_{t-1})p_2(s_t|a_{t-1}) + (p_1(a_{t-1}) - p_2(a_{t-1}))p_2(s_t|a_{t-1})| \\
&\leq \sum_{s_t,a_t} p_1(s_t)\,|p_1(a_t|s_t) - p_2(a_t|s_t)| + \sum_{s_t,a_{t-1}} p_1(a_{t-1})\,|p_1(s_t|a_{t-1}) - p_2(s_t|a_{t-1})| \\
&\quad + \sum_{a_{t-1}} |p_1(a_{t-1}) - p_2(a_{t-1})| \\
&\leq \sum_{s_t,a_t} p_1(s_t)\,|p_1(a_t|s_t) - p_2(a_t|s_t)| + \sum_{s_t,a_{t-1}} p_1(a_{t-1})\,|p_1(s_t|a_{t-1}) - p_2(s_t|a_{t-1})| \\
&\quad + \sum_{s_{t-1},a_{t-1}} |p_1(s_{t-1}, a_{t-1}) - p_2(s_{t-1}, s_{t-1})| \\
&\leq 2\epsilon_{m,\text{post}} + 2\epsilon_{\pi,\text{post}} + \sum_{s_{t-1},a_{t-1}} |p_1(s_{t-1}, a_{t-1}) - p_2(s_{t-1}, s_{t-1})| \\
&\leq 2(t - t_{\text{sw}})(\epsilon_{m,\text{post}} + \epsilon_{\pi,\text{post}}) + \sum_{s_{t_{\text{sw}}},a_{t_{\text{sw}}}} |p_1(s_{t_{\text{sw}}}, a_{t_{\text{sw}}}) - p_2(s_{t_{\text{sw}}}, s_{t_{\text{sw}}})| \\
&\leq 2(t - t_{\text{sw}})(\epsilon_{m,\text{post}} + \epsilon_{\pi,\text{post}}) + 2t_{\text{sw}}(\epsilon_{m,\text{pre}} + \epsilon_{\pi,\text{pre}})
\end{aligned} \qquad (10)$$

■

Now, we start the derivation of our theorems.

Theorem 3 *Under the $k \in \mathbb{N}_{>0}$ steps branched rollouts in the MDP $\langle \mathcal{S}, \mathcal{A}, r, \gamma, p_{st} \rangle$, given the bound of the errors $\epsilon_m = \max_t \mathbb{E}_{a_t \sim \pi_{\mathcal{D}}, s_t \sim p} \left[D_{TV}(p(s_{t+1}|s_t, a_t) || p_\theta(s_{t+1}|s_t, a_t)) \right]$ and $\epsilon_\pi = \max_{s_t} D_{TV}(\pi(a_t|s_t) || \pi_{\mathcal{D}}(a_t|s_t))$, the following inequality holds,*

$$\mathbb{E}_{a \sim \pi, s \sim p}[R] \geq \mathbb{E}_{(a,s) \sim m_b(\pi, p_\theta, \mathcal{D}_{env})}[R] - r_{max} \left\{ \frac{1+\gamma^2}{(1-\gamma)^2} 2\epsilon_\pi + \frac{\gamma - k\gamma^k + (k-1)\gamma^{k+1}}{(1-\gamma)^2} (\epsilon_\pi + \epsilon_m) + \frac{\gamma^k - \gamma}{\gamma - 1}(\epsilon_\pi + \epsilon_m) + \frac{\gamma^k}{1-\gamma}(k+1)(\epsilon_\pi + \epsilon_m) \right\}. \tag{11}$$

Proof

$$\begin{aligned}
\left| \mathbb{E}_{a \sim \pi, s \sim p}[R] - \mathbb{E}_{(a,s) \sim m_b(\pi, p_\theta, \mathcal{D}_{\text{env}})}[R] \right| &= \left| \begin{array}{c} \sum_{s_0, a_0} \left\{ p_\pi(s_0, a_0) - p_{\pi_{\mathcal{D}}}(s_0, a_0) \right\} r(s_0, a_0) \\ + \sum_{t=1}^{k-1} \sum_{s_t, a_t} \gamma^t \left\{ p_\pi(s_t, a_t) - p^{\text{br}}_{t<k}(s_t, a_t) \right\} r(s_t, a_t) \\ + \sum_{t=k}^{\infty} \sum_{s_t, a_t} \gamma^t \left\{ p_\pi(s_t, a_t) - p^{\text{br}}_{t \geq k}(s_t, a_t) \right\} r(s_t, a_t) \end{array} \right| \\
&\leq \left\{ \begin{array}{c} \sum_{s_0, a_0} \left| p_\pi(s_0, a_0) - p_{\pi_{\mathcal{D}}}(s_0, a_0) \right| |r(s_0, a_0)| \\ + \sum_{t=1}^{k-1} \gamma^t \sum_{s_t, a_t} \left| p_\pi(s_t, a_t) - p^{\text{br}}_{t<k}(s_t, a_t) \right| |r(s_t, a_t)| \\ + \sum_{t=k}^{\infty} \gamma^t \sum_{s_t, a_t} \left| p_\pi(s_t, a_t) - p^{\text{br}}_{t \geq k}(s_t, a_t) \right| |r(s_t, a_t)| \end{array} \right\} \\
&\leq \left\{ \begin{array}{c} r_{\max} \underbrace{\sum_{s_0, a_0} \left| p_\pi(s_0, a_0) - p_{\pi_{\mathcal{D}}}(s_0, a_0) \right|}_{\text{term A}} \\ + r_{\max} \sum_{t=1}^{k-1} \gamma^t \underbrace{\sum_{s_t, a_t} \left| p_\pi(s_t, a_t) - p^{\text{br}}_{t<k}(s_t, a_t) \right|}_{\text{term B}} \\ + r_{\max} \sum_{t=k}^{\infty} \gamma^t \underbrace{\sum_{s_t, a_t} \left| p_\pi(s_t, a_t) - p^{\text{br}}_{t \geq k}(s_t, a_t) \right|}_{\text{term C}} \end{array} \right\}
\end{aligned} \tag{12}$$

Here, $p_\pi(s_t, a_t)$ is the state-action visitation probability under p_{st} and π.

For **term A**, we can bound the value in similar manner to the derivation of Lemma 1:

$$\begin{aligned}
\sum_{s_0, a_0} \left| p_\pi(s_0, a_0) - p_{\pi_{\mathcal{D}}}(s_0, a_0) \right| &= \sum_{s_0, a_0} \left| p_\pi(a_0) p(s_0) - p_{\pi_{\mathcal{D}}}(a_0) p(s_0) \right| \\
&= \sum_{s_0, a_0} \left| p_\pi(a_0) p(s_0) - p_\pi(a_0) p(s_0) + \left(p_\pi(a_0) - p_{\pi_{\mathcal{D}}}(a_0) \right) p(s_0) \right| \\
&\leq \underbrace{\sum_{s_0, a_0} p_\pi(a_0) \left| p(s_0) - p(s_0) \right|}_{=0} + \underbrace{\sum_{a_0} \left| p_\pi(a_0) - p_{\pi_{\mathcal{D}}}(a_0) \right|}_{\leq 2\epsilon_\pi} \\
&\leq 2\epsilon_\pi
\end{aligned} \tag{13}$$

For **term B**, we can apply Lemma 1 to bound the value, but it requires the bounded model error under the current policy π. Thus, we need to decompose the distance into two

by adding and subtracting $p_{\pi_\mathcal{D}}$:

$$\begin{aligned}\sum_{s_t,a_t}|p_\pi(s_t,a_t)-p_{\mathrm{br},t<k}(s_t,a_t)| &= \sum_{s_t,a_t}\left|\begin{array}{l}p_\pi(s_t,a_t)-p_{\pi_\mathcal{D}}(s_t,a_t)\\+p_{\pi_\mathcal{D}}(s_t,a_t)-p^{\mathrm{br}}_{t<k}(s_t,a_t)\end{array}\right| \\ &\le \underbrace{\sum_{s_t,a_t}|p_\pi(s_t,a_t)-p_{\pi_\mathcal{D}}(s_t,a_t)|}_{\le 2t\epsilon_\pi} \\ &\quad+\sum_{s_t,a_t}\left|p_{\pi_\mathcal{D}}(s_t,a_t)-p^{\mathrm{br}}_{t<k}(s_t,a_t)\right| \end{aligned} \tag{14}$$

$$\begin{aligned}\sum_{s_t,a_t}\left|p_{\pi_\mathcal{D}}(s_t,a_t)-p^{\mathrm{br}}_{t<k}(s_t,a_t)\right| &= \sum_{s_t,a_t}\left|\tfrac{1}{t}\textstyle\sum_{i=0}^{t-1}p_{\pi_\mathcal{D}}(s_t,a_t)-\tfrac{1}{t}\sum_{i=0}^{t-1}p^{\mathrm{br}}_{t<k,i}(s_t,a_t)\right| \\ &\le \frac{1}{t}\sum_{i=0}^{t-1}\sum_{s_t,a_t}\left|p_{\pi_\mathcal{D}}(s_t,a_t)-p^{\mathrm{br}}_{t<k,i}(s_t,a_t)\right| \\ &\overset{(A)}{\le} \frac{1}{t}\sum_{i=0}^{t-1}\{2\,(t-i)\cdot(\epsilon_\pi+\epsilon_m)\} \\ &= \frac{1}{t}\left\{t^2(\epsilon_\pi+\epsilon_m)+t(\epsilon_\pi+\epsilon_m)\right\} \end{aligned} \tag{15}$$

For (A), we apply Lemma 1 with setting $\epsilon_{m,\mathrm{post}}=\epsilon_m$ and $\epsilon_{\pi,\mathrm{post}}=\epsilon_\pi$ for the rollout following π and p_θ, and $\epsilon_{m,\mathrm{pre}}=0$ and $\epsilon_{\pi,\mathrm{pre}}=0$ for the rollout following $\pi_\mathcal{D}$ and p_{st}, respectively. To recap **term B**, the following inequality holds:

$$\sum_{s_t,a_t}|p_\pi(s_t,a_t)-p_{\mathrm{br},t<k}(s_t,a_t)| \le 2t\epsilon_\pi+t(\epsilon_\pi+\epsilon_m)+(\epsilon_\pi+\epsilon_m) \tag{16}$$

For **term C**, we can derive the bound in a similar manner to the term B case:

$$\begin{aligned}\sum_{s_t,a_t}|p_\pi(s_t,a_t)-p_{\mathrm{br},t\ge k}(s_t,a_t)| &= \sum_{s_t,a_t}\left|\begin{array}{l}p_\pi(s_t,a_t)-p_{\pi_\mathcal{D}}(s_t,a_t)\\+p_{\pi_\mathcal{D}}(s_t,a_t)-p^{\mathrm{br}}_{t\ge k}(s_t,a_t)\end{array}\right| \\ &\le \underbrace{\sum_{s_t,a_t}|p_\pi(s_t,a_t)-p_{\pi_\mathcal{D}}(s_t,a_t)|}_{\le 2t\epsilon_\pi} \\ &\quad+\sum_{s_t,a_t}\left|p_{\pi_\mathcal{D}}(s_t,a_t)-p^{\mathrm{br}}_{t\ge k}(s_t,a_t)\right| \end{aligned} \tag{17}$$

$$
\begin{aligned}
\sum_{s_t,a_t}\left|p_{\pi_{\mathcal{D}}}(s_t,a_t)-p^{\text{br}}_{t\geq k}(s_t,a_t)\right| &= \sum_{s_t,a_t}\left|\tfrac{1}{k}\textstyle\sum_{i=0}^{k-1}p_{\pi_{\mathcal{D}}}(s_t,a_t)-\tfrac{1}{k}\sum_{i=0}^{k-1}p^{\text{br}}_{t\geq k,i}(s_t,a_t)\right| \\
&\leq \frac{1}{k}\sum_{i=0}^{k-1}\sum_{s_t,a_t}\left|p_{\pi_{\mathcal{D}}}(s_t,a_t)-p^{\text{br}}_{t\geq k,i}(s_t,a_t)\right| \\
&\leq \frac{1}{k}\sum_{i=0}^{k-1}\left\{2\left(k-i\right)\cdot(\epsilon_\pi+\epsilon_m)\right\} \\
&= \frac{1}{k}\left\{k^2(\epsilon_\pi+\epsilon_m)+k(\epsilon_\pi+\epsilon_m)\right\} \qquad (18)
\end{aligned}
$$

To recap **term C**, the following equation holds:

$$
\sum_{s_t,a_t}\left|p_{\pi}(s_t,a_t)-p^{\text{br}}_{t\geq k}(s_t,a_t)\right| \leq 2t\epsilon_\pi+k(\epsilon_\pi+\epsilon_m)+(\epsilon_\pi+\epsilon_m) \qquad (19)
$$

By substituting Eqs. 13, 16, and 19, into Eq. 12, we obtain the result:

$$
\begin{aligned}
\left|\mathbb{E}_{a\sim\pi,s\sim p}\left[R\right]-\mathbb{E}_{(a,s)\sim m_b(\pi,p_\theta,\mathcal{D}_{\text{env}})}\left[R\right]\right| &\leq \left\{\begin{array}{c} r_{\max}2\epsilon_\pi \\ +r_{\max}\sum_{t=1}^{k-1}\gamma^t\left\{2t\epsilon_\pi+t(\epsilon_\pi+\epsilon_m)+(\epsilon_\pi+\epsilon_m)\right\} \\ +r_{\max}\sum_{t=k}^{\infty}\gamma^t\left\{2t\epsilon_\pi+k(\epsilon_\pi+\epsilon_m)+(\epsilon_\pi+\epsilon_m)\right\}\end{array}\right\} \\
&= r_{\max}\left\{\begin{array}{c} 2\epsilon_\pi+\frac{1-k\gamma^{(k-1)}+(k-1)\gamma^k}{(1-\gamma)^2}\gamma\left(3\epsilon_\pi+\epsilon_m\right)+\frac{\gamma^k-\gamma}{\gamma-1}(\epsilon_\pi+\epsilon_m) \\ +\sum_{t=k}^{\infty}\gamma^t\left\{2t\epsilon_\pi+k(\epsilon_\pi+\epsilon_m)+(\epsilon_\pi+\epsilon_m)\right\}\end{array}\right\} \\
&= r_{\max}\left\{\begin{array}{c} 2\epsilon_\pi+\frac{1-k\gamma^{(k-1)}+(k-1)\gamma^k}{(1-\gamma)^2}\gamma\left(3\epsilon_\pi+\epsilon_m\right)+\frac{\gamma^k-\gamma}{\gamma-1}(\epsilon_\pi+\epsilon_m) \\ +\sum_{t=1}^{\infty}\gamma^t\left\{2t\epsilon_\pi+k(\epsilon_\pi+\epsilon_m)+(\epsilon_\pi+\epsilon_m)\right\} \\ -\sum_{t=1}^{k-1}\gamma^t\left\{2t\epsilon_\pi+k(\epsilon_\pi+\epsilon_m)+(\epsilon_\pi+\epsilon_m)\right\}\end{array}\right\} \\
&= r_{\max}\left\{\begin{array}{c} 2\epsilon_\pi+\frac{1-k\gamma^{(k-1)}+(k-1)\gamma^k}{(1-\gamma)^2}\gamma\left(3\epsilon_\pi+\epsilon_m\right)+\frac{\gamma^k-\gamma}{\gamma-1}(\epsilon_\pi+\epsilon_m) \\ +\frac{2}{(1-\gamma)^2}\gamma\epsilon_\pi+\frac{\gamma}{1-\gamma}\left\{k(\epsilon_\pi+\epsilon_m)+(\epsilon_\pi+\epsilon_m)\right\} \\ -\sum_{t=1}^{k-1}\gamma^t\left\{2t\epsilon_\pi+k(\epsilon_\pi+\epsilon_m)+(\epsilon_\pi+\epsilon_m)\right\}\end{array}\right\} \\
&= r_{\max}\left\{\begin{array}{c} 2\epsilon_\pi+\frac{1-k\gamma^{(k-1)}+(k-1)\gamma^k}{(1-\gamma)^2}\gamma\left(3\epsilon_\pi+\epsilon_m\right)+\frac{\gamma^k-\gamma}{\gamma-1}(\epsilon_\pi+\epsilon_m) \\ +\frac{2}{(1-\gamma)^2}\gamma\epsilon_\pi+\frac{\gamma}{1-\gamma}\left\{k(\epsilon_\pi+\epsilon_m)+(\epsilon_\pi+\epsilon_m)\right\} \\ -\frac{1-k\gamma^{(k-1)}+(k-1)\gamma^k}{(1-\gamma)^2}2\gamma\epsilon_\pi \\ -\frac{\gamma^k-\gamma}{\gamma-1}\left\{k(\epsilon_\pi+\epsilon_m)+(\epsilon_\pi+\epsilon_m)\right\}\end{array}\right\} \\
&= r_{\max}\left\{\begin{array}{c} \frac{1+\gamma^2}{(1-\gamma)^2}2\epsilon_\pi+\frac{\gamma-k\gamma^k+(k-1)\gamma^{k+1}}{(1-\gamma)^2}\left(\epsilon_\pi+\epsilon_m\right) \\ +\frac{\gamma^k-\gamma}{\gamma-1}(\epsilon_\pi+\epsilon_m)+\left(\frac{\gamma}{1-\gamma}-\frac{\gamma^k-\gamma}{\gamma-1}\right)(k+1)(\epsilon_\pi+\epsilon_m)\end{array}\right\} \\
&= r_{\max}\left\{\begin{array}{c} \frac{1+\gamma^2}{(1-\gamma)^2}2\epsilon_\pi+\frac{\gamma-k\gamma^k+(k-1)\gamma^{k+1}}{(1-\gamma)^2}\left(\epsilon_\pi+\epsilon_m\right) \\ +\frac{\gamma^k-\gamma}{\gamma-1}(\epsilon_\pi+\epsilon_m)+\frac{\gamma^k}{1-\gamma}(k+1)(\epsilon_\pi+\epsilon_m)\end{array}\right\} \qquad (20)
\end{aligned}
$$

■

A.2. Proofs of theorems for performance guarantee in the model-based meta-RL setting

Before starting the derivation of the main theorems, we first introduce a lemma useful for bridging POMDPs, our meta-RL setting, and MDPs.

Lemma 2 (Silver and Veness (2010)) *Given a POMDP $\langle \mathcal{O}, \mathcal{S}, \mathcal{A}, p_{ob}, r, \gamma, p_{st} \rangle$, consider the derived MDP with histories as states, $\langle \mathcal{H}, \mathcal{A}, \gamma, \bar{r}, p_{hi} \rangle$, where $p_{hi} = p(h_{t+1}|a_t, h_t) = \sum_{s_t} \sum_{s_{t+1}} p(s_t|h_t) p(s_{t+1}|s_t, a_t) p(o_{t+1}|s_{t+1}, a_t)$ and $\bar{r}(h_t, a_t) = \sum_{s_t} p(s_t|h_t) r(s_t, a_t)$. Then, the value function of the derived MDP is equal to that of the POMDP.*

Proof The statement can be derived by backward induction on the value functions. See the proof of Lemma 1 in Silver and Veness (2010) for details. ■

Lemma 3 *Given meta-RL setting in Section 4, consider the derived MDP with histories as states, $\langle \mathcal{H}, \mathcal{A}, \gamma, \bar{r}, p_{hi} \rangle$, where $p_{hi} = p(h_{t+1}|a_t, h_t) = \sum_{\tau_t} p(\tau_t|h_t) p(o_{t+1}|\tau_t, o_t, a_t)$ and $\bar{r}(h_t, a_t) = \sum_{\tau_t} p(\tau_t|h_t) r(\tau_t, o_t, a_t)$. Then, the value function of the derived MDP is equal to that in the meta-RL setting in Section 4.*

Proof

By Lemma 2, the value function of the POMDP can be mapped into that of the derived MDP with histories as states, $\langle \mathcal{H}, \mathcal{A}, \gamma, \bar{r}, p_{\text{hi}} \rangle$, where
$p_{\text{hi}} = p(h_{t+1}|a_t, h_t) = \sum_{s_t} \sum_{s_{t+1}} p(s_t|h_t) p(s_{t+1}|s_t, a_t) p(o_{t+1}|s_{t+1}, a_t)$ and $\bar{r}(h_t, a_t) = \sum_{s_t} p(s_t|h_t) r(s_t, a_t)$.

By considering that the hidden state is defined as $\mathcal{S} = \mathcal{T} \times \mathcal{O}$ in the meta-RL setting in Section 4, $p(h_{t+1}|a_t, h_t)$ and $\bar{r}(h_t, a_t)$ can be transformed as:

$$\begin{aligned} p(h_{t+1}|a_t, h_t) &= \sum_{s_t} \sum_{s_{t+1}} p(s_t|h_t) p(s_{t+1}|s_t, a_t) p(o_{t+1}|s_{t+1}, a_t) \\ &= \sum_{s_t} p(s_t|h_t) p(o_{t+1}|s_t, a_t) \\ &= \sum_{\tau_t, o_t} p(\tau_t, o_t|h_t) p(o_{t+1}|\tau_t, o_t, a_t) \\ &= \sum_{\tau_t} p(\tau_t|h_t) p(o_{t+1}|\tau_t, o_t, a_t) \end{aligned} \tag{21}$$

$$\bar{r}(h_t, a_t) = \sum_{s_t} p(s_t|h_t) r(s_t, a_t) = \sum_{\tau_t, o_t} p(\tau_t, o_t|h_t) r(\tau_t, o_t, a_t) = \sum_{\tau_t} p(\tau_t|h_t) r(\tau_t, o_t, a_t) \tag{22}$$

■

Now, we provide the proof of Theorems 1 and 2 in Section 5.

Proof of Theorem 1:

Proof By Lemma 3 our problem of meta-RL can be mapped into the problem in the derived MDP $\langle \mathcal{H}, \mathcal{A}, \gamma, \bar{r}, p_{\text{hi}} \rangle$.

By applying Theorem 4.1 in Janner et al. (2019) to the derived MDP $\langle \mathcal{H}, \mathcal{A}, \gamma, \bar{r}, p_{\text{hi}} \rangle$ with defining ϵ_m, and ϵ_π by Definitions 1 and 2 respectively, we obtain the following inequality in the derived MDP:

$$\mathbb{E}_{a \sim \pi, s \sim p}[R] \geq \mathbb{E}_{(a,s) \sim m_f(\pi, p_\theta, \mathcal{D}_{\text{env}})}[R] - 2r_{\max} \left\{ \frac{\gamma}{(1-\gamma)^2}(\epsilon_m + 2\epsilon_\pi) + \frac{2}{(1-\gamma)}\epsilon_m \right\}, \quad (23)$$

■

Proof of Theorem 2:

Proof By Lemma 3 our problem of meta-RL can be mapped into that in the derived MDP $\langle \mathcal{H}, \mathcal{A}, \gamma, \bar{r}, p_{\text{hi}} \rangle$.

By applying Theorem 3 in Appendix A.1 to the derived MDP $\langle \mathcal{H}, \mathcal{A}, \gamma, \bar{r}, p_{\text{hi}} \rangle$ with defining ϵ_m, and ϵ_π by Definitions 1 and 2 respectively, we obtain the following inequality in the derived MDP:

$$\begin{aligned}\mathbb{E}_{a \sim \pi, s \sim p}[R] \geq \mathbb{E}_{(a,h) \sim m_b(\pi, p_\theta, \mathcal{D}_{\text{env}})}[R] \\ - r_{\max} \left\{ \frac{1+\gamma^2}{(1-\gamma)^2} 2\epsilon_\pi + \frac{\gamma - k\gamma^k + (k-1)\gamma^{k+1}}{(1-\gamma)^2}(\epsilon_\pi + \epsilon_m) \right. \\ \left. + \frac{\gamma^k - \gamma}{\gamma - 1}(\epsilon_\pi + \epsilon_m) + \frac{\gamma^k}{1-\gamma}(k+1)(\epsilon_\pi + \epsilon_m) \right\}. \quad (24)\end{aligned}$$

■

A.3. The discrepancy factors relying on the model error ϵ_m in Theorem 1 and those in Theorem 2

Proof of Corollary 1:

Proof Let the terms relying on ϵ_m in C_{Th1} as $C_{\text{Th1},m}$. Let the terms relying on ϵ_m at $k = 1$ in C_{Th2} as $C_{\text{Th2},m}$.

By Theorems 1 and 2,

$$C_{\text{Th1},m} = r_{\max} \frac{2\gamma\epsilon_m}{(1-\gamma)^2}. \quad (25)$$

$$C_{\text{Th2},m} = r_{\max} \frac{\gamma}{1-\gamma} 2\epsilon_m. \quad (26)$$

Given that $\gamma \in [0, 1)$, $r_{\max} > 0$ and $\epsilon_m \geq 0$,

$$C_{\mathrm{Th2},m} - C_{\mathrm{Th1},m} = r_{\max} \frac{-2\gamma^2 \epsilon_m}{(1-\gamma)^2} \leq 0. \tag{27}$$

■

A.4. Baseline methods for our experiment

PEARL: The model-free meta-RL method proposed in Rakelly et al. (2019). This is an off-policy method and implemented by extending Soft Actor-Critic (Haarnoja et al., 2018). By leveraging experience replay, this method shows high sample efficiency. We reimplemented the PEARL method on TensorFlow, referring to the original implementation on PyTorch (https://github.com/katerakelly/oyster).
Learning to adapt (L2A): The model-based meta-RL proposed in Nagabandi et al. (2019a). In this method, the model is implemented with MAML (Finn et al., 2017) and the optimal action is found by the model predictive path integral control (Williams et al., 2015). We adapt the following implementation of L2A to our experiment: https://github.com/iclavera/learning_to_adapt

A.5. Environments for our experiments

For our experiments in Section 7, we prepare simulated robot environments using the MuJoCo physics engine (Todorov et al., 2012):
Halfcheetah-fwd-bwd: In this environment, policies are used to control the half-cheetah, which is a planar biped robot with eight rigid links, including two legs and a torso, along with six actuated joints. Here, the half-cheetah's moving direction is randomly selected from "forward" and "backward" around every 15 seconds (in simulation time). If the half-cheetah moves in the correct direction, a positive reward is fed to the half-cheetah in accordance with the magnitude of movement, otherwise, a negative reward is fed.
Halfcheetah-pier: In this environment, the half-cheetah runs over a series of blocks that are floating on water. Each block moves up and down when stepped on, and the changes in the dynamics are rapidly changing due to each block having different damping and friction properties. These properties are randomly determined at the beginning of each episode.
Ant-fwd-bwd: Same as Halfcheetah-fwd-bwd except that the policies are used for controlling the ant, which is a quadruped robot with nine rigid links, including four legs and a torso, along with eight actuated joints.
Ant-crippled-leg: In this environment, we randomly sample a leg on the ant to cripple. The crippling of the leg causes unexpected and drastic changes to the underlying dynamics. One of the four legs is randomly crippled every 15 seconds.
Walker2D-randomparams: In this environment, the policies are used to control the walker, which is a planar biped robot consisting of seven links, including two legs and a torso, along with six actuated joints. The walker's torso mass and ground friction is randomly determined every 15 seconds.

Humanoid-direc: In this environment, the policies are used to control the humanoid, which is a biped robot with 13 rigid links, including two legs, two arms and a torso, along with 17 actuated joints. In this task, the humanoid moving direction is randomly selected from two different directions around every 15 seconds. If the humanoid moves in the correct direction, a positive reward is fed to the humanoid in accordance with the magnitude of its movement, otherwise, a negative reward is fed.

A.6. Complementary experimental results

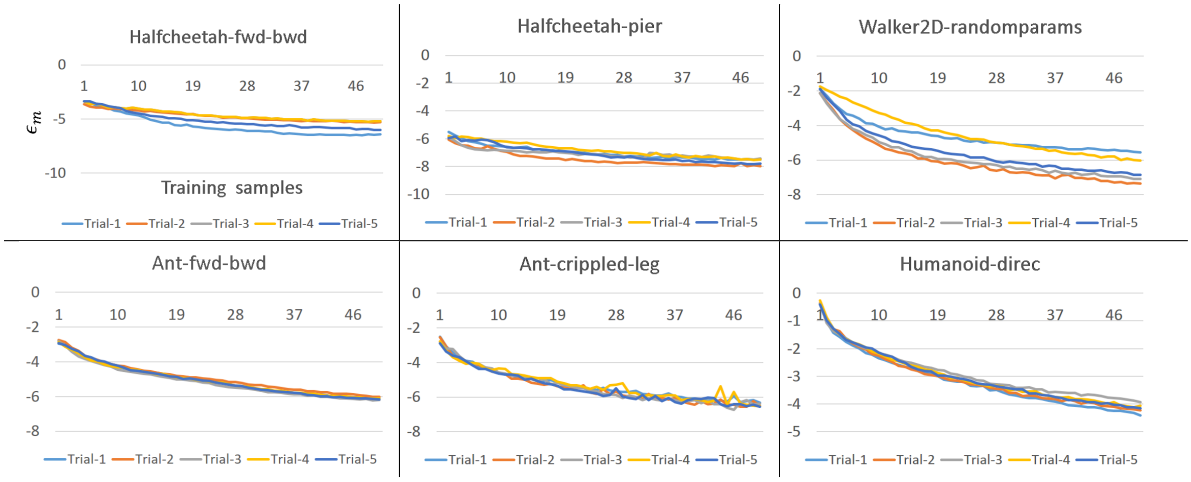

Figure 11: Transition of model errors on training. In each figure, the vertical axis represents empirical values of ϵ_m and the horizontal axis represents the number of training samples (x1000). We ran five trials with different random seeds. The result of the x-th trial is denoted by Trial-x. We used the negative of log-likelihood of the model on validation samples as the approximation of ϵ_m. The figures show that the model error tends to decrease as epochs elapse.

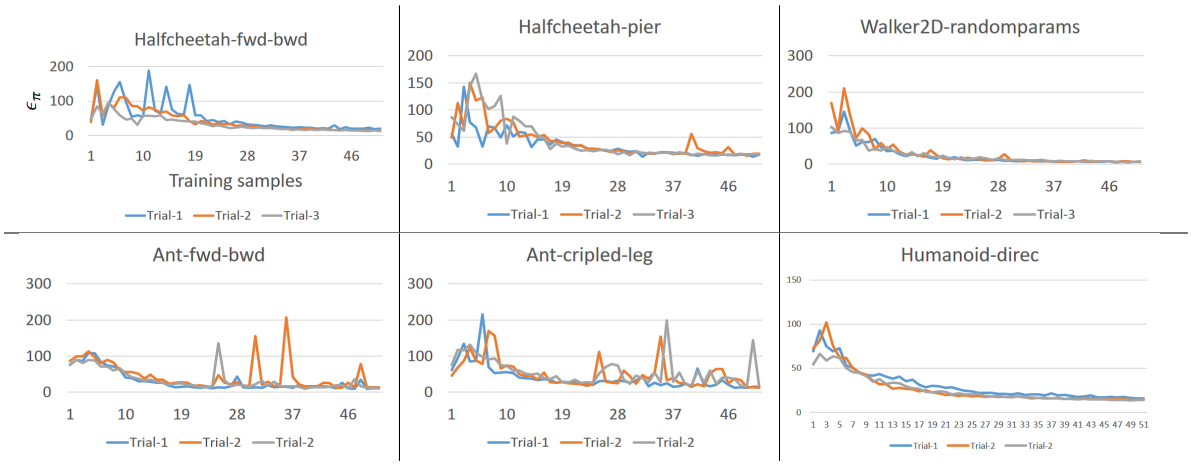

Figure 12: Transition of policy divergence on training. In each figure, the vertical axis represents empirical values of ϵ_π and the horizontal axis represents the number of training samples (x1000). We ran three trials with different random seeds. The result of the x-th trial is denoted by Trial-x. For ϵ_π, we used the empirical Kullback-Leibler divergence of π_θ and $\pi_\mathcal{D}$. Here, $\pi_\mathcal{D}$ has the same policy network architecture with π_θ and is learned by maximum likelihood estimation with the trajectories stored in $\mathcal{D}_{\text{env}}$.

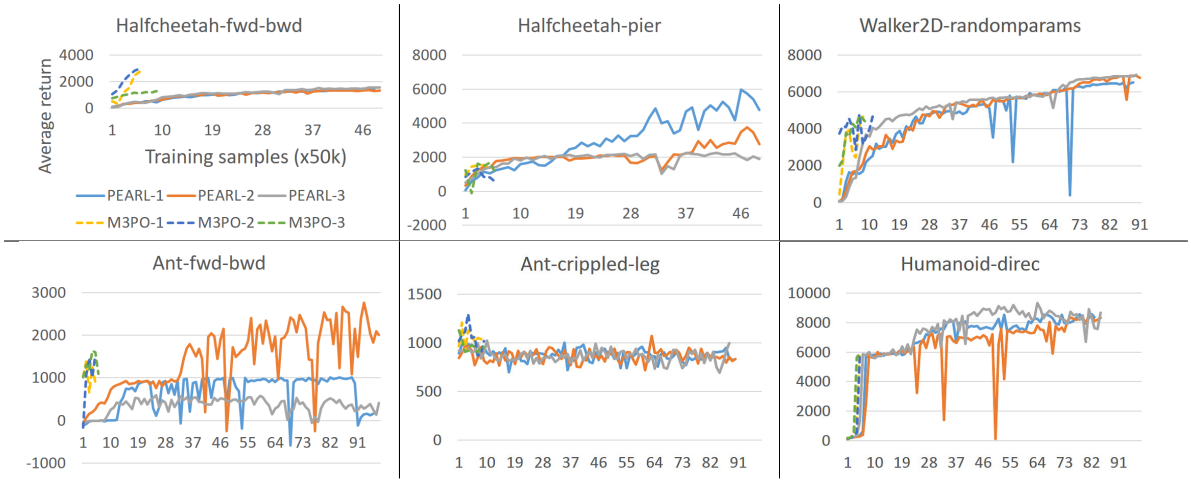

Figure 13: Learning curve of PEARL and M3PO in a long-term training. In each figure, the vertical axis represents expected returns and the horizontal axis represents the number of training samples (**x50000**). The policy and model were fixed and their expected returns were evaluated on 50 test episodes at every 50,000 training samples. Each method was evaluated in three trials, and the result of the x-th trial is denoted by method-x. **Note that the scale of the horizontal axis is larger than that in Figure 7 by 50 times (i.e., 4 in this figure is equal to 200 in Figure 7).**

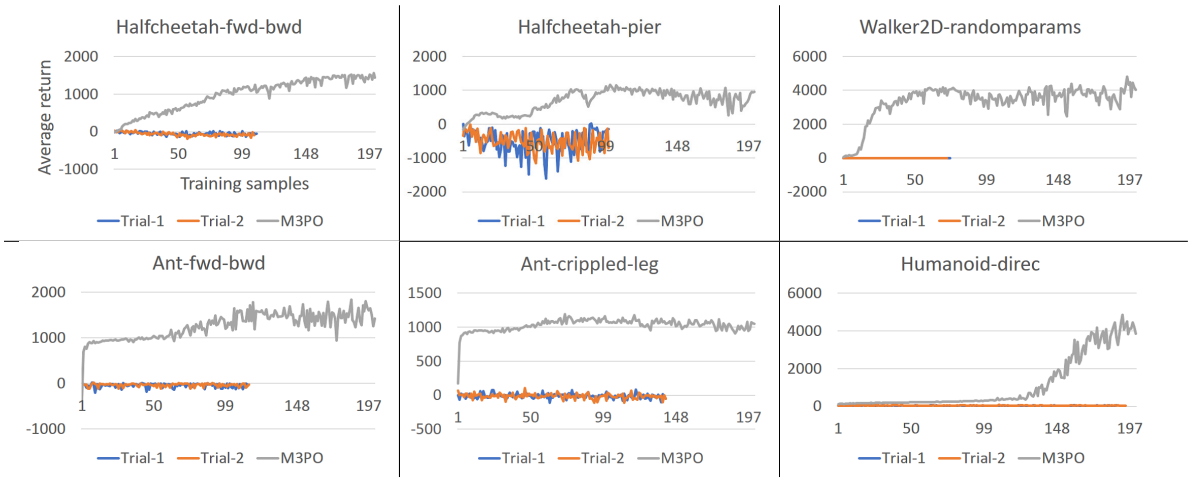

Figure 14: Comparison of GHP-MDP (Algorithm 1 in Perez et al. (2020)) and M3PO. The figures show learning curves of GHP-MDP and M3PO. In each figure, the vertical axis represents expected returns and the horizontal axis represents the number of training samples (x1000). GHP-MDP was evaluated in two trials, and each trial was run for three days in real-times. Due to the limitation of computational resources, we could not run this experiment as many days and trials as other experiments. The expected returns of GHP-MDP in each trial (denoted by "Trial-1" and "Trial-2") are plotted in the figure. The results of M3PO is referred to those in Figure 7. From the comparison result, we can see that M3PO achieves better sample efficiency than GHP-MDP.

A.7. Hyperparameter setting

Table 2: Hyperparameter settings for M3PO results shown in Figure 7.

		Halfcheetah-fwd-bwd	*Halfcheetah-pier*	*Ant-fwd-bwd*	*Ant-crippled-leg*	*Walker2D-randomparams*	*Humanoid-direc*
N	epoch	200					
E	environment step per epoch	1000					
M	model-based rollouts per environment step	1e3			5e2	1e3	5e2
B	ensemble size	3					
G	policy update per environment step	40		20			
k	model-based rollout length	1					
L	history length	10					
	network architecture	GRU (Cho et al., 2014) of five units for RNN. MLP of two hidden layers of 400 swish (Ramachandran et al., 2017) units for FFNN					

Table 3: β settings for results shown in Figure 15. $a \to b$ denotes a thresholded linear function, i.e., at epoch e, $f(e) = \min(\max(1 - \frac{e-a}{b-a}, 1), 0)$.

		Halfcheetah-pier	*Walker2D-randomparams*	*Humanoid-direc*
β	mixture ratio	$80 \to 130$	$50 \to 100$	$150 \to 250$

A.8. M3PO with hybrid dataset

In Figures 7 and 13, we can see that, in a number of the environments (Halcheetah-pier, Walker2D-randomparams, and Humanoid-direc), the long-term performance of M3PO is worse than that of PEARL. This indicates that a gradual transition from M3PO to PEARL (or other model-free approaches) needs to be considered to improve overall performance. In this section, we propose to introduce such a gradual transition approach to M3PO and evaluate it on the environments where the long-term performance of M3PO is worse than that of PEARL.

For the gradual transition, we introduce a hybrid dataset $\mathcal{D}_{\text{hyb}}$. This contains a mixture of the real trajectories in $\mathcal{D}_{\text{env}}$ and the fictitious trajectories in $\mathcal{D}_{\text{model}}$, on the basis of mixture ratio $\beta \in [0, 1]$. Formally, $\mathcal{D}_{\text{mix}}$ is defined as

$$\mathcal{D}_{\text{hyb}} = \beta \cdot \mathcal{D}_{\text{model}} + (1 - \beta) \cdot \mathcal{D}_{\text{env}}. \tag{28}$$

We replace $\mathcal{D}_{\text{model}}$ in line 11 in Algorithm 2 with $\mathcal{D}_{\text{hyb}}$. We linearly reduce the value of β from 1 to 0 in accordance with the training epoch. With this value reduction, the M3PO gradually becomes less dependent on the model and is transitioned to the model-free approach.

We evaluate M3PO with the gradual transition (M3PO-h) in three environments (Halfcheetah-pier, Walker2D-randomparams and Humanoid-direc), in which the long-term performance of M3PO is worse than that of PEARL in Figures 7 and 13. The hyperparameter setting (except for the setting schedule for the value of β) for the experiment is the same as that for Figures 7 and 13 (i.e., the one shown in Table 2). Regarding the setting schedule for the value of β, we reduce it in accordance with Table 3. Evaluation results are shown in Figure 15. We can see that M3PO-h achieves the same or better scores with the long-term performances of PEARL in all environments.

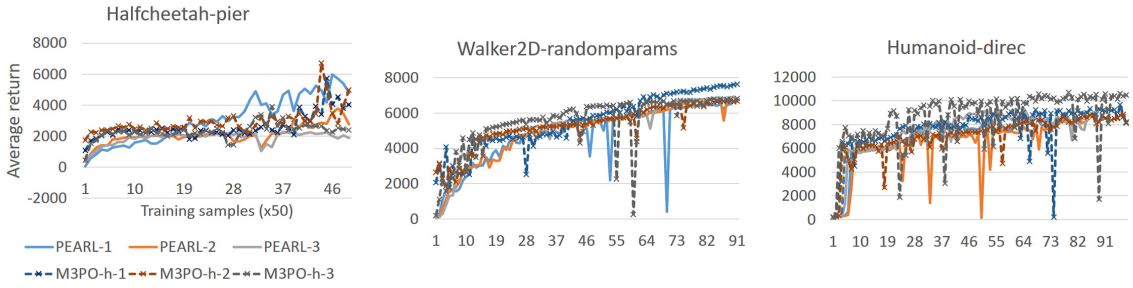

Figure 15: Learning curve of PEARL, M3PO, and M3PO-h in a long-term training. In each figure, the vertical axis represents expected returns and the horizontal axis represents the number of training samples (**x50000**). The policy and model were fixed and their expected returns were evaluated on 50 episodes at every 50,000 training samples. Each method was evaluated in three trials, and the result of the x-th trial is denoted by method-x. **Note that the scale of the horizontal axis is larger than that in Figure 7 by 50 times (i.e., 4 in this figure is equal to 200 in Figure 7).**

A.9. The effect of model adaptation

In this section, we present a complementary analysis to answer the question "Does model adaptation (i.e., the use of a meta-model) in M3PO contribute to the improvement of the meta-policy?".

We compare M3PO with **Model-based Meta-Policy Optimization (M2PO)**. M2PO is a variant of M3PO in which a non-adaptive transition model is used instead of the meta-model. The model architecture is the same as that in the MBPO algorithm (Janner et al., 2019) (i.e., the ensemble of Gaussian distributions based on four-layer feed-forward neural networks).

Our experimental result indicates that the use of a meta-model contributes to performance improvement in some of the environments. In Figure 16, we can clearly see the improvement of M3PO against M2PO in Halfcheetah-fwd-bwd. In addition, in the Ant environments, although the M3PO's performance is seemingly the same as that of M2PO, the qualitative performance is quite different; the M3PO can produce a meta-policy for walking in the correct direction, while M2PO failed to do so (M2PO produces the meta-policy "always standing" with a very small amount of control signal). For Humanoid-direc, by contrast, M2PO tends to achieve better sample efficiency than M3PO. We hypothesize that the primary reason for this is that during the plateau at the early stage of training in Humanoid-direc, the model used in M2PO generates fictitious trajectories that make meta-policy optimization more stable. To verify this hypothesis, we compare TD-errors (Q-function errors), which are an indicator of the stability of meta-policy optimization, for M3PO and M2PO. The evaluation result (Figure 17 in the appendix) shows that during the performance plateau (10–60 epoch), the TD-error in M2PO was actually lower than that in M3PO; this result supports our hypothesis. In this paper, we did not focus on the study of meta-model usage to generate the trajectories that make meta-policy optimization stable, but this experimental result indicates that such a study is important for further improving M3PO.

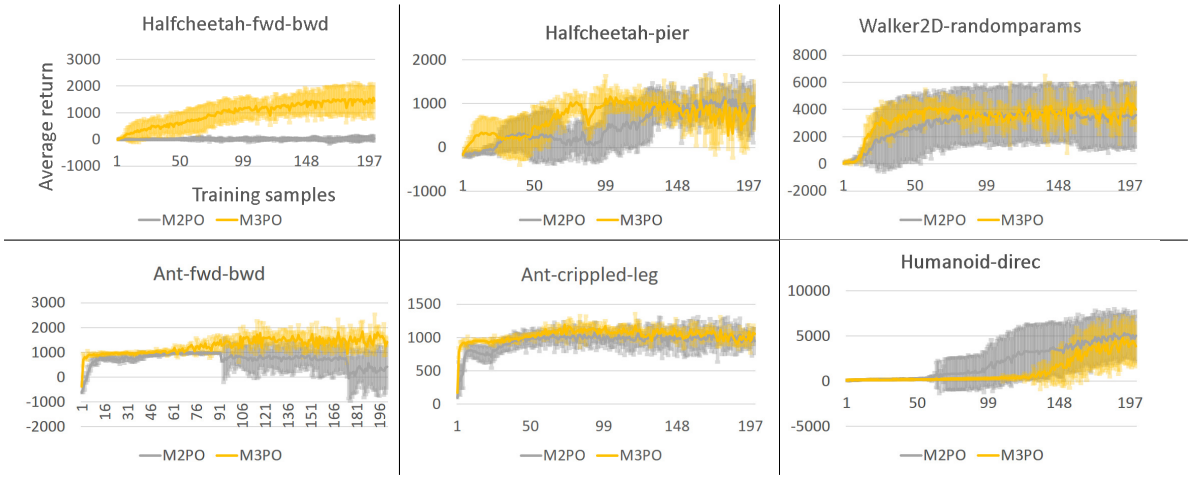

Figure 16: The learning curve of M3PO and M2PO. In each figure, the vertical axis represents expected returns and the horizontal axis represents the number of training samples (x1000). The meta-policy and models were fixed and their expected returns were evaluated on 50 episodes at every 1000 training samples for the other methods. In each episode, the task was initialized and changed randomly. Each method was evaluated in at least five trials, and the expected return on the 50 episodes was further averaged over the trials. The averaged expected returns and their standard deviations are plotted in the figures.

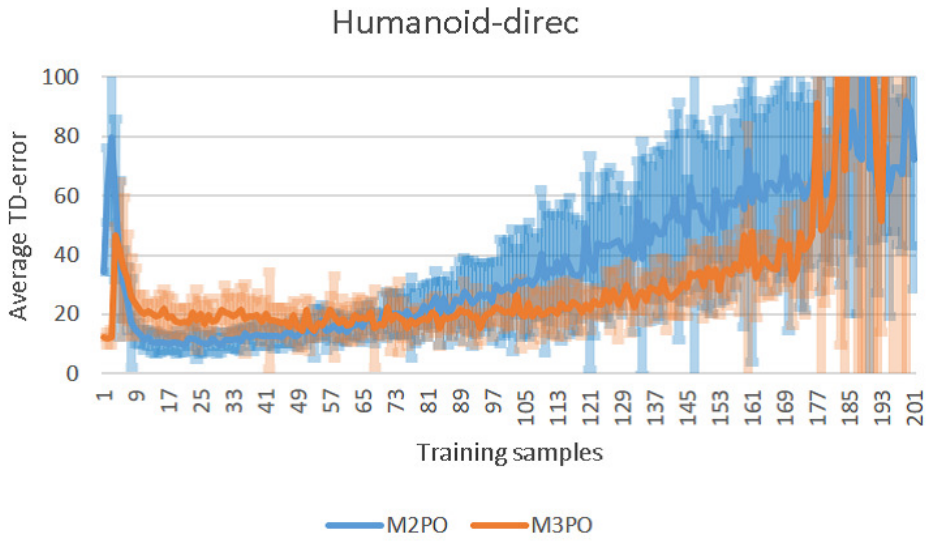

Figure 17: The transition of TD-errors (Q-function error) on training. In each figure, the vertical axis represents TD-error and the horizontal axis represents the number of training samples (x1000). We ran ten trials with different random seeds and plotted the average of their results. The error bar means one standard deviation.