1 Appendix

A Proof Sketch

To better clarify our theoretical results, we provide a proof sketch here. Firstly, we decompose the performance difference bound under the model-based setting into three terms (Theorem 1). Secondly, by means of using Return Bound (Theorem 2), we can bound these three terms individually (Theorem 3). Then, we can do some transformation to get Unified Model Shift and Model Bias Bound (Theorem 4), which bounds the model shift term and the model bias term in total variation form. However, due to the intractable property of Δ, we further explore the upper bound of $|\Delta|$ (Theorem 5), finding that Δ can be ignored. Finally, by the Integral Probability Metrics (Lemma 3) and the property of the Wasserstein distance, we derive the target which bounds the model shift term and the model bias term in the Wasserstein distance form.

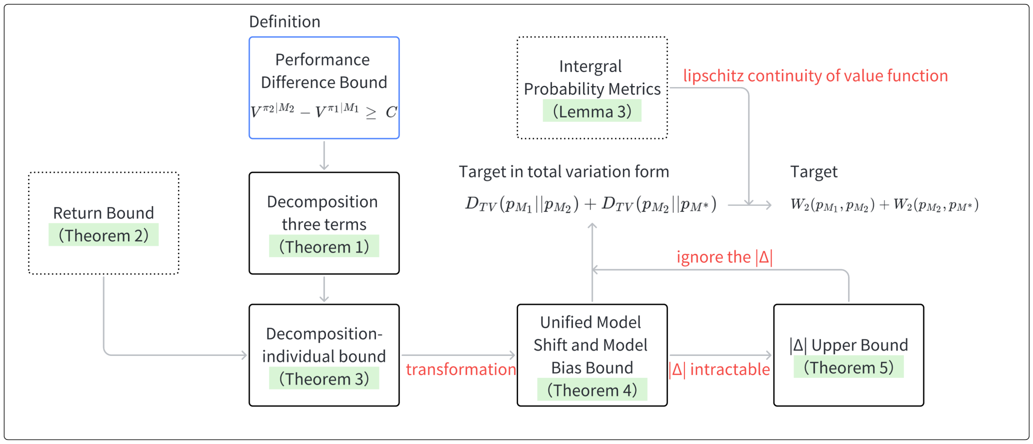
Definition
Performance Difference Bound
$V^{\pi_2|M_2} - V^{\pi_1|M_1} \geq C$
Intergral Probability Metrics
(Lemma 3)
lipschitz continuity of value function
Return Bound
(Theorem 2)
Decomposition three terms
(Theorem 1)
Target in total variation form
$D_{TV}(p_{M_1}||p_{M_2}) + D_{TV}(p_{M_2}||p_{M^*})$
Target
$W_2(p_{M_1}, p_{M_2}) + W_2(p_{M_2}, p_{M^*})$
ignore the |Δ|
Decomposition-individual bound
(Theorem 3)
transformation
Unified Model Shift and Model Bias Bound
(Theorem 4)
|Δ| intractable
|Δ| Upper Bound
(Theorem 5)

Figure 1: Theoretical sketch of USB-PO.

B Useful Lemmas

In this section, we provide some proof to support our theoretical analysis.

Lemma 1 (Total variation Distance). *Consider a measurable space* (Ω, Σ) *and probability measures* P *and* Q *are defined on* (Ω, Σ). *The total variation distance between* P *and* Q *is defined as:*

$$D_{TV}(P||Q) = \sup_{A \in \Sigma} |P(A) - Q(A)| \tag{1}$$

Eq.(1) *can be equivalently written as:*

$$D_{TV}(P||Q) = \frac{1}{2} \sum_{\omega \in \Omega} |P(\{\omega\}) - Q(\{\omega\})| \tag{2}$$

Proof: The proof of this lemma can be found in [7]. □

Lemma 2 (Total Variation Distance of Joint Distributions). *Given two distributions* $p(x, y) = p(x)p(y|x)$ *and* $q(x, y) = q(x)q(y|x)$, *the total variation distance between them can be bounded as:*

$$D_{TV}(p(x, y)||q(x, y)) \leq D_{TV}(p(x)||q(x)) + \max_x D_{TV}(p(y|x)||q(y|x)) \tag{3}$$

Proof:

$$
\begin{aligned}
D_{TV}(p(x,y)||q(x,y)) &= \frac{1}{2}\sum_{x,y}|p(x,y)-q(x,y)| \\
&= \frac{1}{2}\sum_{x,y}|p(x)p(y|x)-q(x)q(y|x)| \\
&= \frac{1}{2}\sum_{x,y}|p(x)p(y|x)-p(x)q(y|x)+p(x)q(y|x)-q(x)q(y|x)| \\
&\leq \frac{1}{2}\sum_{x,y}p(x)|p(y|x)-q(y|x)|+|p(x)-q(x)|q(y|x) \\
&= \frac{1}{2}\sum_{x,y}p(x)|p(y|x)-q(y|x)|+\frac{1}{2}\sum_{x}|p(x)-q(x)| \\
&= \mathbb{E}_{x\sim p(x)}[D_{TV}(p(y|x)||q(y|x))]+D_{TV}(p(x)||q(x)) \\
&\leq D_{TV}(p(x)||q(x))+\max_x D_{TV}(p(y|x)||q(y|x))
\end{aligned}
\tag{4}
$$

□

Lemma 3 (Integral Probability Metrics). *Consider a measurable space$(\mathcal{X},\Sigma)$. The integral probability metric associated with a class $\mathcal{F}$ of real-valued functions on $\mathcal{X}$ is defined as*

$$d_{\mathcal{F}}(P,Q)=\sup_{f\in\mathcal{F}}|\mathbb{E}_{X\sim P}[f(X)]-\mathbb{E}_{Y\sim Q}[f(Y)]| \tag{5}$$

where P and Q are probability measures on $\mathcal{X}$. We demonstrate the following special cases:

(a) If $\mathcal{F}=\{f:||f||_\infty\leq c\}$, then we have

$$d_{\mathcal{F}}(P,Q)=cD_{TV}(P||Q) \tag{6}$$

(b) If $\mathcal{F}$ is the set of $L-$ Lipschitz function with a norm $||\cdot||$, then we have

$$d_{\mathcal{F}}(P,Q)=LW_1(P,Q) \tag{7}$$

In our paper, to distinguish the dynamic transition function, we choose $\mathcal{F}$ to be the class covering V_M^π. Since the value function can converge to $\frac{r_{max}}{1-\gamma}$, it only needs to satisfy the L_v-Lipschitz continuity and thus we can get $\frac{r_{max}}{1-\gamma}D_{TV}(p_M||p_{M'})=L_vW_1(p_M,p_{M'})$ for any arbitrary model M,M'.

C Missing Proof

Theorem 1 (Performance Difference Bound Decomposition). *Let $M_i\in\mathcal{M}$ be the evaluated model and $\pi_i\in\Pi$ be the policy derived from the model. The performance difference bound can be decomposed into three terms,*

$$V^{\pi_2|M_2}-V^{\pi_1|M_1}=(V^{\pi_2|M_2}-V_{M_2}^{\pi_2})-(V^{\pi_1|M_1}-V_{M_1}^{\pi_1})+(V_{M_2}^{\pi_2}-V_{M_1}^{\pi_1}) \tag{8}$$

Proof: We introduce two additional terms $V_{M_1}^{\pi_1}$ and $V_{M_2}^{\pi_2}$ that allow the performance difference bound objective to be divided into three operators based on the return bounds, which can be reformulated separately.

$$
\begin{aligned}
V^{\pi_2|M_2}-V^{\pi_1|M_1} &= V^{\pi_2|M_2}-V^{\pi_1|M_1}+(V_{M_1}^{\pi_1}-V_{M_1}^{\pi_1})+(V_{M_2}^{\pi_2}-V_{M_2}^{\pi_2}) \\
&= (V^{\pi_2|M_2}-V_{M_2}^{\pi_2})-(V^{\pi_1|M_1}-V_{M_1}^{\pi_1})+(V_{M_2}^{\pi_2}-V_{M_1}^{\pi_1})
\end{aligned}
\tag{9}
$$

□

Theorem 2 (Return Bound). *Let R_{max} denote the bound of the reward function, ϵ_π denote $\max_s D_{TV}(\pi_1||\pi_2)$ and $\epsilon_{M_1}^{M_2}$ denote $\mathbb{E}_{(s,a)\sim d_{M_1}^{\pi_1}}[D_{TV}(p_{M_1}||p_{M_2})]$. For two arbitrary policies $\pi_1,\pi_2\in\Pi$, the expected return under two arbitrary models $M_1,M_2\in\mathcal{M}$ can be bounded as,*

$$V_{M_2}^{\pi_2}-V_{M_1}^{\pi_1}\geq-2R_{max}(\frac{\epsilon_\pi}{(1-\gamma)^2}+\frac{\gamma}{(1-\gamma)^2}\epsilon_{M_1}^{M_2}) \tag{10}$$

Proof: We give the thorough proof referring to Lemma B.4 in MBPO [4] as follows.

$$\begin{aligned} V^{\pi_2}_{M_2} - V^{\pi_1}_{M_1} &= \sum_{t=0}^{\infty} \gamma^t \sum_{s,a} (p^{\pi_2}_{t,M_2}(s,a) - p^{\pi_1}_{t,M_1}(s,a)) r(s,a) \\ &\geq -R_{max} \sum_{t=0}^{\infty} \gamma^t \sum_{s,a} |p^{\pi_2}_{t,M_2}(s,a) - p^{\pi_1}_{t,M_1}(s,a)| \\ &= -2R_{max} \sum_{t=0}^{\infty} \gamma^t D_{TV}(p^{\pi_1}_{t,M_1}(s,a) || p^{\pi_2}_{t,M_2}(s,a)) \end{aligned} \tag{11}$$

According to the Lemma 2, we have:

$$\begin{aligned} D_{TV}(p^{\pi_1}_{t,M_1}(s,a) || p^{\pi_2}_{t,M_2}(s,a)) &\leq D_{TV}(p^{\pi_1}_{t,M_1}(s) || p^{\pi_2}_{t,M_2}(s)) + \max_s D_{TV}(\pi_1(\cdot|s) || \pi_2(\cdot|s)) \\ &= D_{TV}(p^{\pi_1}_{t,M_1}(s) || p^{\pi_2}_{t,M_2}(s)) + \epsilon_\pi \end{aligned} \tag{12}$$

Further we expand the first term:

$$\begin{aligned} &D_{TV}(p^{\pi_1}_{t,M_1}(s) || p^{\pi_2}_{t,M_2}(s)) \\ &= \frac{1}{2} \sum_s |p^{\pi_1}_{t,M_1}(s) - p^{\pi_2}_{t,M_2}(s)| \\ &= \frac{1}{2} \sum_s |\sum_{s'} p^{\pi_1}_{M_1}(s|s') p^{\pi_1}_{t-1,M_1}(s') - p^{\pi_2}_{M_2}(s|s') p^{\pi_2}_{t-1,M_2}(s')| \\ &\leq \frac{1}{2} \sum_s \sum_{s'} |p^{\pi_1}_{M_1}(s|s') p^{\pi_1}_{t-1,M_1}(s') - p^{\pi_2}_{M_2}(s|s') p^{\pi_2}_{t-1,M_2}(s')| \\ &\leq \frac{1}{2} \sum_{s,s'} p^{\pi_1}_{t-1,M_1}(s') |p^{\pi_1}_{M_1}(s|s') - p^{\pi_2}_{M_2}(s|s')| + p^{\pi_2}_{M_2}(s|s') |p^{\pi_1}_{t-1,M_1}(s') - p^{\pi_2}_{t-1,M_2}(s')| \\ &= \frac{1}{2} \mathbb{E}_{s' \sim p^{\pi_1}_{t-1,M_1}(s')} [\sum_s |p^{\pi_1}_{M_1}(s|s') - p^{\pi_2}_{M_2}(s|s')|] + D_{TV}(p^{\pi_1}_{t-1,M_1}(s') || p^{\pi_2}_{t-1,M_2}(s')) \\ &= \frac{1}{2} \sum_{t'=1}^{t} \mathbb{E}_{s' \sim p^{\pi_1}_{t'-1,M_1}(s')} [\sum_s |p^{\pi_1}_{M_1}(s|s') - p^{\pi_2}_{M_2}(s|s')|] \\ &= \frac{1}{2} \sum_{t'=1}^{t} \mathbb{E}_{s' \sim p^{\pi_1}_{t'-1,M_1}(s')} [\sum_s |\sum_a p^{\pi_1}_{M_1}(s,a|s') - p^{\pi_2}_{M_2}(s,a|s')|] \\ &\leq \frac{1}{2} \sum_{t'=1}^{t} \mathbb{E}_{s' \sim p^{\pi_1}_{t'-1,M_1}(s')} [\sum_{s,a} |p^{\pi_1}_{M_1}(s,a|s') - p^{\pi_2}_{M_2}(s,a|s')|] \\ &= \sum_{t'=1}^{t} \mathbb{E}_{s' \sim p^{\pi_1}_{t'-1,M_1}(s')} D_{TV}(p^{\pi_1}_{M_1}(s,a|s') || p^{\pi_2}_{M_2}(s,a|s')) \\ &\leq \sum_{t'=1}^{t} \mathbb{E}_{s' \sim p^{\pi_1}_{t'-1,M_1}(s')} [\epsilon_\pi + \mathbb{E}_{a \sim \pi_1} [D_{TV}(p_{M_1}(s|s',a) || p_{M_2}(s|s',a))]] \\ &= t\epsilon_\pi + \sum_{t'=1}^{t} \mathbb{E}_{s',a \sim p^{\pi_1}_{t'-1,M_1}(s',a)} D_{TV}(p_{M_1}(s|s',a) || p_{M_2}(s|s',a)) \end{aligned} \tag{13}$$

Then move the result of Eq.(13) to Eq.(12), we can get:

$$D_{TV}(p^{\pi_1}_{t,M_1}(s,a) || p^{\pi_2}_{t,M_2}(s,a)) \leq (t+1)\epsilon_\pi + \sum_{t'=1}^{t} \mathbb{E}_{s',a \sim p^{\pi_1}_{t'-1,M_1}(s',a)} D_{TV}(p_{M_1}(s|s',a) || p_{M_2}(s|s',a)) \tag{14}$$

Next, we move the result of Eq.(14) to Eq.(11), we can get:

$$\begin{aligned} &V^{\pi_2}_{M_2} - V^{\pi_1}_{M_1} \\ &\geq -2R_{max}\sum_{t=0}^{\infty}\gamma^t\Big((t+1)\epsilon_\pi + \sum_{t'=1}^{t}\mathbb{E}_{s',a\sim p^{\pi_1}_{t'-1,M_1}(s',a)} D_{TV}(p_{M_1}(s|s',a)||p_{M_2}(s|s',a))\Big) \\ &= -2R_{max}\Big(\frac{\epsilon_\pi}{(1-\gamma)^2} + \frac{1}{1-\gamma}\sum_{t=1}^{\infty}\gamma^t\mathbb{E}_{s',a\sim p^{\pi_1}_{t-1,M_1}(s',a)} D_{TV}(p_{M_1}(s|s',a)||p_{M_2}(s|s',a))\Big) \end{aligned} \tag{15}$$

Here, we first simplify the second term of the Eq.(15)

$$\begin{aligned} &\frac{1}{1-\gamma}\sum_{t=1}^{\infty}\gamma^t\mathbb{E}_{s',a\sim p^{\pi_1}_{t-1,M_1}(s',a)} D_{TV}(p_{M_1}(s|s',a)||p_{M_2}(s|s',a)) \\ &= \frac{\gamma}{1-\gamma}\sum_{t=0}^{\infty}\gamma^t\mathbb{E}_{s',a\sim p^{\pi_1}_{t,M_1}(s',a)} D_{TV}(p_{M_1}(s|s',a)||p_{M_2}(s|s',a)) \\ &= \frac{\gamma}{(1-\gamma)^2}(1-\gamma)\sum_{t=0}^{\infty}\gamma^t\mathbb{E}_{s',a\sim p^{\pi_1}_{t,M_1}(s',a)} D_{TV}(p_{M_1}(s|s',a)||p_{M_2}(s|s',a)) \\ &= \frac{\gamma}{(1-\gamma)^2}\mathbb{E}_{s',a\sim d^{\pi_1}_{M_1}(s',a)} D_{TV}(p_{M_1}(s|s',a)||p_{M_2}(s|s',a)) \\ &= \frac{\gamma}{(1-\gamma)^2}\epsilon^{M_2}_{M_1} \end{aligned} \tag{16}$$

Then we bring this result back to Eq.(15) and the proof is complete.

$$V^{\pi_2}_{M_2} - V^{\pi_1}_{M_1} \geq -2R_{max}\Big(\frac{\epsilon_\pi}{(1-\gamma)^2} + \frac{\gamma}{(1-\gamma)^2}\epsilon^{M_2}_{M_1}\Big) \tag{17}$$

□

Theorem 3 (Decomposition TVD Bound)**.** *Let* $\epsilon^{\pi_i}_{M_i}$ *denote* $\mathbb{E}_{(s,a)\sim d^{\pi_i}_{M_i}}[D_{TV}(p_{M_i}||p_{M^*})]$*. Let* $M_i \in \mathcal{M}$ *be the evaluated model and* $\pi_i \in \Pi$ *be the policy derived from the model. The decomposition terms can be bounded as,*

$$V^{\pi_2|M_2} - V^{\pi_1|M_1} \geq \frac{2R_{max}\gamma}{(1-\gamma)^2}(\epsilon^{\pi_1}_{M_1} - \epsilon^{\pi_2}_{M_2} - \epsilon^{M_2}_{M_1}) - \frac{2R_{max}\epsilon_\pi}{(1-\gamma)^2} \tag{18}$$

Proof: According to CMLO [5] and Eq.(10), the term $V^{\pi_1|M_1} - V^{\pi_1}_{M_1}$ can be approximated as $-\frac{2R_{max}\gamma}{(1-\gamma)^2}\epsilon^{\pi_1}_{M_1}$, thus we only need to bound the remaining two terms.

For the term $V^{\pi_2|M_2} - V^{\pi_2}_{M_2}$, we use Eq.(10) to bound it.

$$\begin{aligned} V^{\pi_2|M_2} - V^{\pi_2}_{M_2} &\geq -2R_{max}\Big(\frac{\underset{s}{\max} D_{TV}(\pi_2||\pi_2)}{(1-\gamma)^2} + \frac{\gamma}{(1-\gamma)^2}\epsilon^{\pi_2}_{M_2}\Big) \\ &= -\frac{2R_{max}\gamma}{(1-\gamma)^2}\epsilon^{\pi_2}_{M_2} \end{aligned} \tag{19}$$

Similarly, for the term $V^{\pi_2}_{M_2} - V^{\pi_1}_{M_1}$, we can get:

$$V^{\pi_2}_{M_2} - V^{\pi_1}_{M_1} \geq -2R_{max}\Big(\frac{\epsilon_\pi}{(1-\gamma)^2} + \frac{\gamma}{(1-\gamma)^2}\epsilon^{M_2}_{M_1}\Big) \tag{20}$$

We now combine these three bounds together and complete the proof.

$$\begin{aligned} V^{\pi_2|M_2} - V^{\pi_1|M_1} &= (V^{\pi_2|M_2} - V^{\pi_2}_{M_2}) - (V^{\pi_1|M_1} - V^{\pi_1}_{M_1}) + (V^{\pi_2}_{M_2} - V^{\pi_1}_{M_1}) \\ &\geq -\frac{2R_{max}\gamma}{(1-\gamma)^2}\epsilon^{\pi_2}_{M_2} + \frac{2R_{max}\gamma}{(1-\gamma)^2}\epsilon^{\pi_1}_{M_1} - 2R_{max}\Big(\frac{\epsilon_\pi}{(1-\gamma)^2} + \frac{\gamma}{(1-\gamma)^2}\epsilon^{M_2}_{M_1}\Big) \\ &= \frac{2R_{max}\gamma}{(1-\gamma)^2}(\epsilon^{\pi_1}_{M_1} - \epsilon^{\pi_2}_{M_2} - \epsilon^{M_2}_{M_1}) - \frac{2R_{max}\epsilon_\pi}{(1-\gamma)^2} \end{aligned} \tag{21}$$

□

Theorem 4 (Unified Model Shift and Model Bias Bound)**.** *Let* κ *denote the constant* $\frac{2R_{max}}{(1-\gamma)^2}$ *and* Δ *denotes* $\mathbb{E}_{(s,a)\sim d^{\pi_1}_{M_1}}[D_{TV}(p_{M_2}||p_{M^*})] - \mathbb{E}_{(s,a)\sim d^{\pi_2}_{M_2}}[D_{TV}(p_{M_2}||p_{M^*})]$*. Let* $M_i \in \mathcal{M}$ *be the evaluated model and* $\pi_i \in \Pi$ *be the policy derived from the model. The unified model shift and model bias bound can be derived as,*

$$\begin{aligned} &V^{\pi_2|M_2} - V^{\pi_1|M_1} \\ &\geq \kappa(\gamma(\mathbb{E}_{(s,a)\sim d^{\pi_1}_{M_1}}[D_{TV}(p_{M_1}||p_{M^*}) - D_{TV}(p_{M_1}||p_{M_2}) - D_{TV}(p_{M_2}||p_{M_*})] + \Delta) - \epsilon_\pi) \end{aligned} \tag{22}$$

Proof: Based on Eq.(18), we add a new term $\kappa\mathbb{E}_{(s,a)\sim d^{\pi_1}_{M_1}}[D_{TV}(p_{M_2}||p_{M^*})]$ to reformulate the optimization objective. ☐

Theorem 5 ($|\Delta|$ Upper Bound)**.** *Let* $M_i \in \mathcal{M}$ *be the evaluated model and* $\pi_i \in \Pi$ *be the policy derived from the model. The term* Δ *can be upper bounded as:*

$$|\Delta| \leq \frac{2\gamma}{1-\gamma}\mathbb{E}_{(s,a)\sim d^{\pi_1}_{M_1}}[D_{TV}(p_{M_1}||p_{M_2})\max_{s,a} D_{TV}(p_{M_2}||p_{M^*})] + \frac{2\epsilon_\pi}{1-\gamma}\max_{s,a} D_{TV}(p_{M_2}||p_{M^*}) \tag{23}$$

Proof: First, we combine these two terms.

$$\begin{aligned} |\Delta| &= |\mathbb{E}_{(s,a)\sim d^{\pi_1}_{M_1}}[D_{TV}(p_{M_2}||p_{M^*})] - \mathbb{E}_{(s,a)\sim d^{\pi_2}_{M_2}}[D_{TV}(p_{M_2}||p_{M^*})]| \\ &= (1-\gamma)|\sum_{t=0}^{\infty}\gamma^t\sum_{s,a}(p^{\pi_1}_{t,M_1}(s,a) - p^{\pi_2}_{t,M_2}(s,a))D_{TV}(p_{M_2}(s'|s,a)||p_{M^*}(s'|s,a))| \\ &\leq (1-\gamma)\max_{s,a} D_{TV}(p_{M_2}(s'|s,a)||p_{M^*}(s'|s,a))\sum_{t=0}^{\infty}\gamma^t\sum_{s,a}|p^{\pi_1}_{t,M_1}(s,a) - p^{\pi_2}_{t,M_2}(s,a)| \\ &= 2(1-\gamma)\max_{s,a} D_{TV}(p_{M_2}||p_{M^*})\sum_{t=0}^{\infty}\gamma^t D_{TV}(p^{\pi_1}_{t,M_1}(s,a)||p^{\pi_2}_{t,M_2}(s,a)) \end{aligned} \tag{24}$$

Recalling that we get the result of the sum equation above in Eq.(14), and then we have:

$$\begin{aligned} |\Delta| &\leq 2(1-\gamma)\max_{s,a} D_{TV}(p_{M_2}||p_{M^*})(\frac{\epsilon_\pi}{(1-\gamma)^2} + \frac{\gamma}{(1-\gamma)^2}\epsilon^{M_2}_{M_1}) \\ &= \frac{2\gamma}{1-\gamma}\mathbb{E}_{s,a\sim d^{\pi_1}_{M_1}}[D_{TV}(p_{M_1}||p_{M_2})\max_{s,a} D_{TV}(p_{M_2}||p_{M^*})] + \frac{2\epsilon_\pi}{1-\gamma}\max_{s,a} D_{TV}(p_{M_2}||p_{M^*}) \end{aligned} \tag{25}$$

☐

D Experimental Details

D.1 Environment Setup

We evaluate the algorithm over a series of MuJoCo [11] continuous control benchmark tasks. To ensure fairness, we use the standard 1000-step version of all the environments. The details of the environment setup are from OpenAI Gym [1], as shown in Table 1.

Table 1: The general outline of the MuJoCo environment.

Environment-Version	State Dim	Action Dim	Termination
Ant-v2	27	8	obs[0]<0.2 or obs[0] > 1.0
HalfCheetah-v2	17	6	-
Hopper-v2	11	3	obs[1] $\geq$ 0.2 or obs[0] $\leq$ 0.7
Humanoid-v2	45	17	obs[0] < 1.0 or obs[0] > 2.0
InvertedPendulum-v2	4	1	obs[1] > 0.2 or obs[1] < -0.2
Walker2d-v2	17	6	obs[0] $\geq$ 2.0 or obs[0] $\leq$ 0.8 or obs[1] $\geq$ 1.0 or obs[1] $\leq$ -1.0

D.2 Baseline implementation

MFRL Baselines. We use two state-of-the-art model-free algorithms, i.e. SAC [3] and PPO [9], to do baseline comparison. To demonstrate the final performance and sampling efficiency of our method, we train SAC for 3M steps, which is much more than MBRL algorithms. The hyperparameters are consistent with the author's settings.

MBRL Baselines. We use several state-of-the-art model-based algorithms to do baseline comparison, covering CMLO [5], MBPO [4], SLBO [8] and STEVE [2]. The implementation of CMLO is based on the opensource repo published by the author and all of the hyperparameters are set according to the paper [5]. Our algorithm USB-PO is implemented based on the opensource repo published by Janner who is the author of MBPO.

We present the final performance on six continuous benchmark tasks in Table 2. The results demonstrate that our algorithm achieves competitive performance compared to both MBRL and MFRL baselines over these tasks. Each result in the table shows the average and standard deviation on the maximum average returns among different random seeds and we choose 250K for HalfCheetah-v2, 300K for Walker2d-v2, 300K for Humanoid-v2, 250K for Ant-v2, 15K for Inverted-Pendulum-v2, 120K for Hopper-v2.

Table 2: The final performance on six continuous benchmark tasks.

		HalfCheetah	Humanoid	Walker2d
MBRL	STEVE	12406.29±458.08	4318.32±853.60	1109.23±1163.74
	SLBO	1915.47±1398.73	459.46±34.27	3107.93±1887.09
	MBPO	12765.67±594.54	5546.77±221.72	4582.06±67.44
	CMLO	10143.55±193.82	5577.01±219.89	4807.60±99.89
	USB-PO	**15105.91±177.75**	**5973.75±110.99**	**5691.62±162.57**
MFRL(@3M steps)	SAC	15012	6207	5879
		Ant	InvertedPendulum	Hopper
MBRL	STEVE	779.72±45.67	778.54±265.51	1131.61±623.52
	SLBO	707.79±218.80	793.24±334.90	898.68±233.21
	MBPO	4926.10±818.38	1000.00±0.00	3436.00±120.72
	CMLO	5123.71±783.97	1000.00±0.00	3495.41±71.02
	USB-PO	**6340.84±119.06**	**1000.00±0.00**	**3694.22±46.19**
MFRL(@3M steps)	SAC	5934	1000	3610

D.3 Hyperparameters

Our algorithm USB-PO is based on MBPO [4] and is implemented according to the opensource repo published by the MBPO author. Except for the learning rate in phase 2 of our USB-PO algorithm, the hyperparameters are completely identical to the MBPO settings for all environments. In all benchmark tasks, we set this learning rate to 1e-4.

D.4 Computing Infrastructure

In Table 3, we list our computing infrastructure and the computational time for training USB-PO on these six continuous benchmark tasks. Note that the time we report is the cost for 4 random seeds simultaneously on one graphics card. For Humanoid, only two random seeds can be run simultaneously because of the limitation of graphics memory.

Table 3: Computing infrastructure and the computational time for each benchmark task compared to MBPO, where the time unit d denotes day and h denotes hour.

	HalfCheetah	Humanoid	Walker2d	Ant	InvertedPendulum	Hopper
CPU	AMD EPYC 7B12 64-Core Processor					
GPU	NVIDIA 2080Ti					
MBPO times	2.46d	1.64d	1.75d	2.88d	3.43h	17.81h
USB-PO times	2.29d	1.51d	1.65d	2.91d	3.42h	18.28h

E Comparison with Prior Works

In this section, we compare USB-PO with prior theoretical works to emphasize our contribution, as a complementary to the main paper. First, we give a summary and then show the details as follows. MBPO-Style does not consider model shift and CMLO-Style rely on a fixed threshold to constrain model shift. Our algorithm, USB-PO, adaptively adjusts the model updates in a unified manner (unify model shift and model bias) to get the performance improvement guarantee.

MBPO-Style [4, 10, 6, 12]. They use the return discrepancy bound $V^{\pi|M} \geq V^{\pi}_{M} - C(\epsilon_m, \epsilon_\pi)$ to improve the lower bound on the performance under the real environment, i.e. as long as improving V^{π}_{M} by more than $C(\epsilon_m, \epsilon_\pi)$ can guarantee improvement on $V^{\pi|M}$. Obviously, This scheme is guaranteed under a fixed model and it does not consider the change in model dynamic during updates nor the performance variation concerning model shift. Even worse, if the model has some excessive updates, it is impractical to find a feasible solution to meet the improvement guarantee.

CMLO-Style [5]. They use the performance difference bound under the model-based setting $V^{\pi_2|M_2} - V^{\pi_1|M_1} \geq C$ to directly consider model shift and model bias. However, they finally derive a constrained lower-bound optimization problem and use a fixed threshold to constrain model shift, i.e. $\sup_{s\in\mathcal{S},a\in\mathcal{A}} D_{TV}(P_{M_1}(\cdot|s,a)||P_{M_2}(\cdot|s,a)) \leq \sigma_{M_1,M_2}$ and determine when to update the model accordingly. Notably, we find that this fixed threshold plays a key role in the whole algorithm and needs to be carefully adjusted for each environment. If this threshold is set too low, the model bias of the following iteration will be large, which impairs the subsequent optimization process. If this threshold is set too high, the performance improvement can no longer be guaranteed. Additionally, using a fixed threshold during the whole training process makes the algorithm problematic to adjust adaptively.

USB-PO (Ours). Following CMLO-Style [5], we also use the performance difference bound under the model-based setting to directly consider model shift and model bias. Compared to relying on a fixed threshold to constrain model shift, we use a transformation to unify model shift and model bias into one formulation without the constraint (Theorem 4). Due to the intractable property of Δ, we further explore the upper bound of $|\Delta|$, finding that Δ can be ignored with respect to model shift and model bias alone (Theorem 5). Finally, the optimization objective we get can be used to fine-tune M_2 in a unified manner to adaptively adjust the model updates to get a performance improvement guarantee. Notably, our algorithm can use the same learning rate of Phase 2 and our algorithm is robust to this learning rate. To the best of our knowledge, this is the first method that unifies model shift and model bias and adaptively fine-tunes the model updates during the training process.

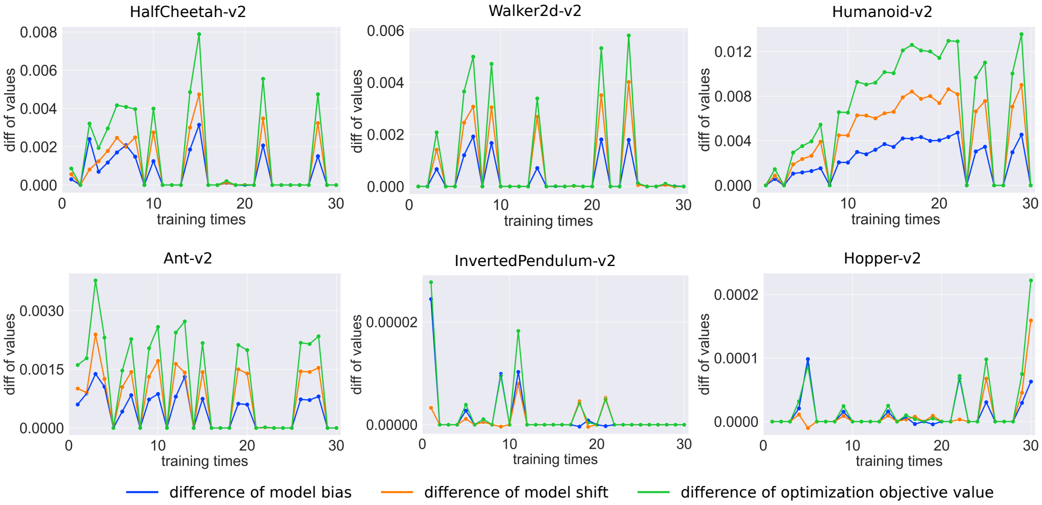

Figure 2: We choose a specific random seed to show the details of the first 30 training times on all benchmark tasks, covering the difference of optimization objective value, the model shift term and the model bias term before and after the fine-tuning process.

F Additional Experiment

F.1 Working Mechanism Extension

To illustrate that the ability of USB-PO to reduce both model shift and model bias potentially is not a coincidence that exists only in the Walker2d environment, we add experimental results in other MuJoCo [11] environments. As shown in Figure 2, when the fine-tuning actually operates, the difference of the model shift term and the model bias term among all of the benchmark tasks are generally both positive, further validating our superiority.

F.2 Ablation Study Extension

Here, we show the results of the ablation study on all of the MuJoCo benchmark tasks.

As shown in Figure 3, only optimizing the model shift term results in a drop in sample efficiency while only optimizing the model bias term leads to performance deterioration. Only fine-tuning the model updates in a unified manner can achieve excellent performance.

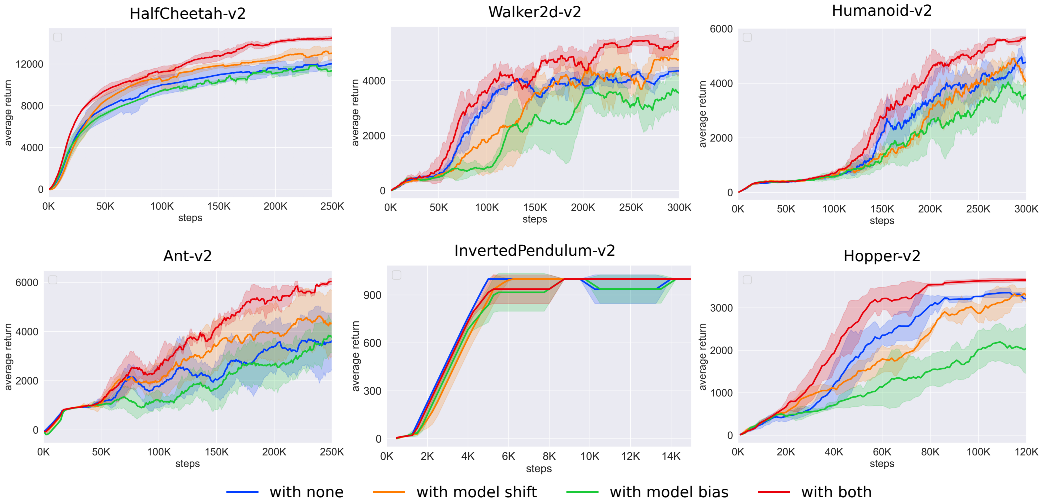

Figure 3: **Optimization Objective Variants** on all MuJoCo benchmark tasks.

F.3 Ensemble numbers

To illustrate the justification of the ensemble numbers we set, we conduct the ablation experiment on more ensemble numbers containing 3, 5, and 7. As Figure 4 shows, as the number of ensemble models goes up, the performance will be higher and more stable, but it will cost more time. To maintain the balance between performance and time, we finally set the value of this parameter to 7, which is recommended by the MBPO [4] original repo.

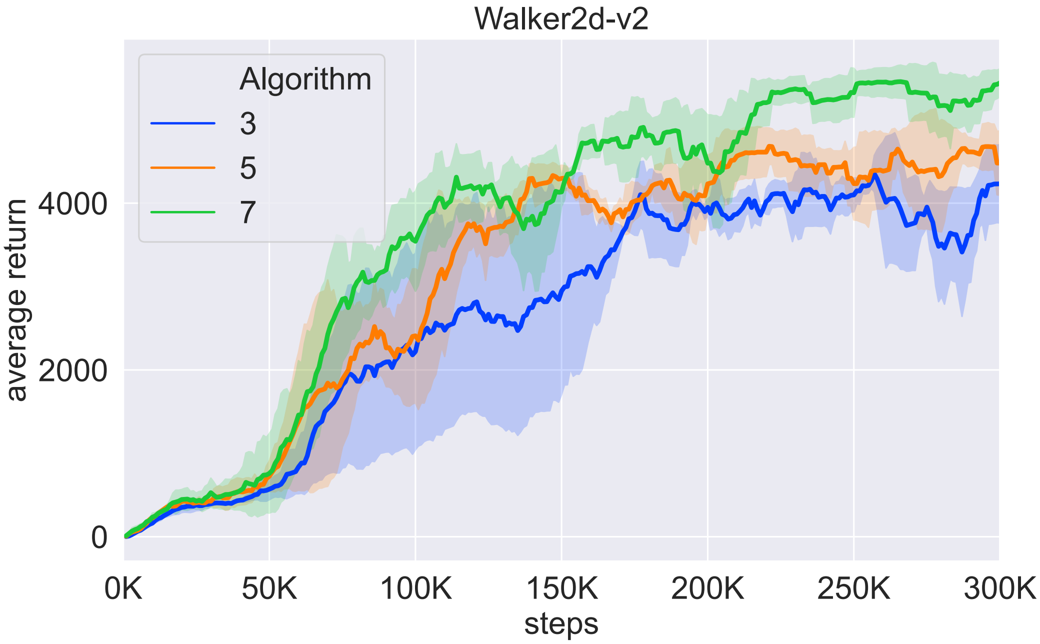

Figure 4: Ablation experimental results of the ensemble model numbers.