Thank all the reviewers for the insightful comments and the helpful suggestions!

The reviewers' common concern is about *the connections between the algorithm and the theoretical analysis*. So we first outline the connections as follows.

1. Our return bound removes the term that involves the behavior policy, which is consistent with our algorithm. It fixes an inconsistency issue of MBPO: their theoretical results suggest to constrain the distance between the behavior policy and the new policy, but their algorithm does not have such a constraint, leading to bad performances in some cases.

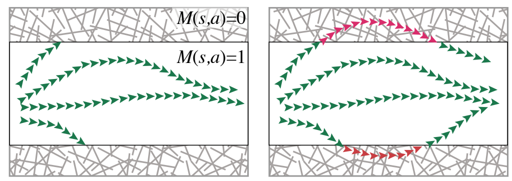
2. The theoretical results tell us that the gap between model returns and actual returns depends on the model bias in *model* rollouts instead of that in real rollouts (or in validation trajectories). Previous methods can be understood as $M(s,a) \equiv 1$ for all (s,a) during model rollouts though the model bias can be large (or even undefined, because the imaginary state may not be a valid state), so they suffer from large performance gap especially when using long model rollouts. So this result suggests to restrict model usage to reduce the gap, which leads to our actual algorithm.

3. We formulate the gap in the form of $\epsilon \cdot w$. Here ϵ is the maximum model error during *model* rollouts. So it is difficult to quantitatively constrain ϵ, if possible. However, we can always control w, the portion of model-generated samples to be used. So it leads to our rank-based heuristic that selects model-generated samples with the hyper-parameter w.

To Reviewer #1:

Q1. *A more sophisticated approach to choose w?*

A1. We agree that it is a great idea. Actually, we have tried to choose samples by using the samples whose predictive likelihoods are less than the average likelihood of a hold-out validation dataset, which had have better performance than trivial likelihood-based heuristics. However, we do not include it because 1) we want our algorithm to be easy-to-use, i.e., having competitive performance in most environments with the default hyper-parameters (set w with a linear schedule around 0.25), and 2) our rank-based heuristic with OvR uncertainty estimation can have better performance.



(a) Hard-stop mode (b) Non-stop mode

Figure 1: A demonstration of the model rollout modes in M2AC. (a) Hard-stop mode stops model rollouts once it encounters an (s,a) that $M(s,a) = 0$; (b) Non-stop mode always runs $H_{\max}$ steps and only keeps the samples that has $M(s,a) = 1$ (in green).

Q2. *Explanation of non-stop mode and hard-stop mode?*

A2. Thanks for pointing this out. As demonstrated in Figure 1, non-stop mode can provide richer samples. We will add detailed explanation and ablation studies in text and in figure in the final version.

To Reviewer #2:

Q1. *7 and 10 runs are too few, at least 30 would be better. Model-free methods in the noisy environments are missing.*

A1. Thanks for the suggestion. This was due to limited computational resources (as we have to run many environments in many different settings). We will add more runs as well as model-free baselines in the final version.

To Reviewer #3: Q1. *Meaning of "small" uncertainty score in Line-10 of Algorithm 2?*

A1. When the agent generates a batch of B imaginary samples, it aligns an uncertainty score with the OvR estimation for each sample. Then it ranks these samples by their uncertainty scores, and selects the first $\lfloor wB \rfloor$ samples (whose uncertainty scores are "smaller" than others). Here, the only hyper-parameter is w. We show by experiments that our default choice of w works fairly well across a wide range of tasks, and the algorithm is robust with varying w.

Q2. *Lines 154-164 provides more of a high level intuition rather than an properly laid out interpretation.*

A2. Thanks for pointing this out. We revised this paragraph to make it clearer.

To Reviewer #4:

Q1. *Results should include more environments (InvertedPendulum and Ant).*

A1. Thanks for the suggestion. We will add the results in the appendix. Here we report the average results of Ant in Table 1. As for InvertedPendulum, M2AC performs comparably good as MBPO (Return=1000) because the task is too simple.

Table 1: Ant-v2.

$H_{\max}$	1	4	7
MBPO	2167	3586	2394
M2AC	3907	4102	3306

Q2. *How to use the predictor $u(s,a)$ for the model-bias penalty?*

A2. For the model-bias penalty, since $D_{TV}(\cdot,\cdot) \leq \sqrt{D_{KL}(\cdot\|\cdot)/2}$ and our $u(s,a)$ in OvR uncertainty estimation in Eq.(9) is a KL-divergence, we compute the sample mean of $\alpha\sqrt{u(s,a)/2}$ as the model-bias penalty. We will add detailed explanation in the final version.