A Note on Using Branched Rollout to Improve The Efficiency of The Model-based Algorithm

Kuan-Chen, Pan
Department of Computer Science
National Yang Ming Chiao Tung University
penny644.cs07@nycu.edu.tw

1 Introduction

Reinforcement learning algorithms can be mainly divided into two types: model-based and model-free. Although model-free algorithms can directly map from state to action, it is relatively inefficient in learning. Model-based algorithms no need to collect a lot of data in the real world, but the accuracy of the model often becomes the bottleneck of the algorithm. In this paper, the authors design an algorithm, named as MBPO(Janner et al. [2019]), which is a model-based algorithm with monotonic improvement.

MBPO algorithm is based on the previous papers and proposed branched rollouts which replace long rollout with short rollout. The steps of MBPO algorithm is to collect data in the environment and use the data to train the model, then performs k steps model rollouts, and finally updates the parameters in the model. Starting k steps rollout from a state under the state distribution of previous policy is the concept of branched rollouts.

Compared with other papers, the main contribution of this paper is to design a model-based algorithm that guarantees monotonic improvement and come up with the idea of branched rollouts. While there are many model-based algorithms before, few guarantee monotonic improvement. In addition, compared with the previous method, MBPO uses branch rollout to start rollout under different state distributions, avoiding the entanglement of rollout length and task horizon.

Regarding the MBPO algorithm, I think that using a short rollout can indeed effectively avoid the drawbacks that a long rollout may amplify the inaccuracy of the model. In this method, k is an important parameter to adjust whether it depends on the model. Therefore, the value of k has a great influence on the performance of the method, and determining the value of hyperparameter k will become an important issue for this method.

2 Problem Formulation

I consider a MDP, defined by the tuple $(S, A, p, r, \gamma, \rho_0)$. S is state space and A is action space. p is transition distribution and r is reward function. γ is the discount factor between 0 and 1. ρ_0 is initial state distribution. In addition, I use π to denote the policy. The authors assume that MBPO is used in the non-linear system. In order to prove that this method guarantees monotonic improvement, it is necessary to give a bound

$$\eta[\pi] \geq \hat{\eta}[\pi] - \mathrm{C}.$$

$\eta[\pi]$ and $\hat{\eta}[\pi]$ is the return of the policy in the true MDP and under the model, respectively.

In this paper, the authors want to use branched rollout to remove the limitation of rollout length and task horizon entanglement and reduce the impact of model inaccuracy on learning efficiency. In addition, whether it can continue to optimize the bound that will be mentioned in the next chapter to make learning more efficient is a very interesting problem. In the theoretical analysis section, further derivations will be made to obtain more precise lower bound.

3 Theoretical Analysis

In this section, I will derive the precise lower bound according to the bound which is showed in the problem formulation section:

$$\eta[\pi] \geq \hat{\eta}[\pi] - \mathrm{C}.$$

C in the above bound is mainly affected by generalization error and distribution shift. The generalization error can be expressed as

$$\epsilon_m = max_t \; E_{s\sim\pi_{D,t}}[D_{TV}(p(s',r|s,a)||p_\theta(s',r|s,a))].$$

The distribution shift can be denoted by

$$\epsilon_\pi \geq max_s[D_{TV}(\pi||\pi_D)].$$

Therefore, we can change the above bound to

$$\eta[\pi] \geq \hat{\eta}[\pi] - [\frac{2\gamma r_{max}(\epsilon_m + 2\epsilon_\pi)}{(1-\gamma)^2} + \frac{4r_{max}\epsilon_\pi}{1-\gamma}].$$

Before we start to prove this formula, we have to list some useful lemmas and the proof of lemma 1,2 and 4 are correct in the original paper so we omit. However, the proof of lemma 3 on the original paper is somewhat wrong, so we prove it again.

Lemma 1. $D_{TV}(p_1(x,y)||p_2(x,y)) \leq D_{TV}(p_1(x)||p_2(x)) + max_t D_{TV}(p_1(y|x)||p_2(y|x))$

Lemma 2. $D_{TV}(p_1^t(s)||p_2^t(s)) \leq t\delta$

Lemma 3. $|\eta_1 - \eta_2| \leq \frac{2R\gamma(\epsilon_m+\epsilon_\pi)}{(1-\gamma)^2} + \frac{2R\epsilon_\pi}{1-\gamma}$

Proof. We prove the lemma 3 again and mark symbols which are missed in the original paper in red.

$$\begin{aligned}
|\eta_1 - \eta_2| &= |\sum_{s,a}(p_1(s,a) - p_2(s,a))r(s,a)\,| \\
&= |\sum_{s,a}(\sum_t \gamma^t p_1^t(s,a) - \gamma^t p_2^t(s,a))r(s,a))\,| \\
&\leq \sum_{s,a}\sum_t |\gamma^t p_1^t(s,a) - \gamma^t p_2^t(s,a)|r(s,a) \\
&\quad (by\ triangle\ inequality) \\
&\leq r_{max}\sum_{s,a}\sum_t |\gamma^t p_1^t(s,a) - \gamma^t p_2^t(s,a)| \\
&\leq 2r_{max}\sum_t \gamma^t t(\epsilon_m + \epsilon_\pi) + \gamma^t\epsilon_\pi \\
&\quad (by\ lemma\ 2\ and\ setting\ \delta = \epsilon_m + \epsilon_\pi) \\
&\leq 2r_{max}(\frac{\gamma(\epsilon_m+\epsilon_\pi)}{(1-\gamma)^2} + \frac{\epsilon_\pi}{1-\gamma}) \\
&\quad (by\ the\ formula\ of\ infinite\ geometric\ progression)
\end{aligned}$$

□

Lemma 4. $|\eta_1 - \eta_2| \leq 2r_{max}[\frac{\gamma^{k+1}(\epsilon_m^{pre}+\epsilon_\pi^{pre})}{(1-\gamma)^2} + \frac{k(\epsilon_m^{post}+\epsilon_\pi^{post})}{1-\gamma} + \frac{\gamma^k\epsilon_\pi^{pre}}{1-\gamma} + \frac{\epsilon_\pi^{post}}{1-\gamma}]$

After we have the above four lemmas, we can start to prove the bound

$$\eta[\pi] \geq \hat{\eta}[\pi] - [\frac{2\gamma r_{max}(\epsilon_m + 2\epsilon_\pi)}{(1-\gamma)^2} + \frac{4r_{max}\epsilon_\pi}{1-\gamma}].$$

Proof.

$$\begin{aligned}\eta[\pi] - \hat{\eta}[\pi] &= \eta[\pi] - \eta[\pi_D] + \eta[\pi_D] - \hat{\eta}[\pi] \\ &\geq -2r_{max}(\frac{\gamma\epsilon_\pi}{(1-\gamma)^2} + \frac{\epsilon_\pi}{1-\gamma}) - 2r_{max}(\frac{\gamma(\epsilon_m + \epsilon_\pi)}{(1-\gamma)^2} + \frac{\epsilon_\pi}{1-\gamma}) \\ &\quad (by\ lemma\ 3) \\ &= -2r_{max}\frac{\gamma(\epsilon_m + 2\epsilon_\pi)}{(1-\gamma)^2} - 4r_{max}\frac{\epsilon_\pi}{1-\gamma}\end{aligned}$$

$$\Rightarrow \eta[\pi] \geq \hat{\eta}[\pi] - [\frac{2\gamma r_{max}(\epsilon_m + 2\epsilon_\pi)}{(1-\gamma)^2} + \frac{4r_{max}\epsilon_\pi}{1-\gamma}]$$

□

After we derive the above precise bound, we try to derive a more precise bound under branch rollout with branch length k and denote the model's error on the distribution of the current policy π by ϵ'_m. So, we can derive the following bound

$$\eta[\pi] \geq \eta^{branch}[\pi] - 2r_{max}[\frac{\gamma^{k+1}\epsilon_\pi}{(1-\gamma)^2} + \frac{\gamma^k\epsilon_\pi}{1-\gamma} + \frac{k\epsilon'_m}{1-\gamma}].$$

Proof. I prove this bound again and mark the place which is wrong in the original paper in red.

$$\eta[\pi] - \eta^{branch}[\pi] = \eta[\pi] - \eta^{\pi_D,\pi} + \eta^{\pi_D,\pi} - \eta^{branch}[\pi]$$

$$|\eta[\pi] - \eta^{\pi_D,\pi}| \leq 2r_{max}[\frac{\gamma^{k+1}\epsilon_\pi}{(1-\gamma)^2} + \frac{\gamma^k\epsilon_\pi}{1-\gamma}] \quad (by\ lemma\ 4)$$

$$|\eta^{\pi_D,\pi} - \eta^{branch}[\pi]| \leq 2r_{max}[\frac{k\epsilon'_m}{1-\gamma}]$$

$(the\ red\ part\ in\ the\ paper\ is\ |\eta[\pi] - \eta^{\pi_D,\pi}|\ and\ I\ change\ to\ |\eta^{\pi_D,\pi} - \eta^{branch}[\pi]|.)$

We can use the above two inequality to continue to prove the bound.

$$\begin{aligned}\eta[\pi] - \eta^{branch}[\pi] &= \eta[\pi] - \eta^{\pi_D,\pi} + \eta^{\pi_D,\pi} - \eta^{branch}[\pi] \\ &\geq -2r_{max}[\frac{\gamma^{k+1}\epsilon_\pi}{(1-\gamma)^2} + \frac{\gamma^k\epsilon_\pi}{1-\gamma}] - 2r_{max}[\frac{k\epsilon'_m}{1-\gamma}] \\ &\quad (by\ lemma\ 4) \\ &\geq -2r_{max}[\frac{\gamma^{k+1}\epsilon_\pi}{(1-\gamma)^2} + \frac{\gamma^k\epsilon_\pi}{1-\gamma} + \frac{k\epsilon'_m}{1-\gamma}]\end{aligned}$$

$$\Rightarrow \eta[\pi] \geq \eta^{branch}[\pi] - 2r_{max}[\frac{\gamma^{k+1}\epsilon_\pi}{(1-\gamma)^2} + \frac{\gamma^k\epsilon_\pi}{1-\gamma} + \frac{k\epsilon'_m}{1-\gamma}]$$

□

In addition, we can approximate ϵ'_m by $\epsilon_m + 2\epsilon_\pi$. So we can change the bound to

$$\eta[\pi] \geq \eta^{branch}[\pi] - 2r_{max}[\frac{\gamma^{k+1}\epsilon_\pi}{(1-\gamma)^2} + \frac{\gamma^k\epsilon_\pi}{1-\gamma} + \frac{k(\epsilon_m + 2\epsilon_\pi)}{1-\gamma}].$$

Proof.

$$\begin{aligned}
\eta[\pi] - \eta^{branch}[\pi] &= \eta[\pi] - \eta^{\pi_D,\pi} + \eta^{\pi_D,\pi} - \eta^{\pi_D,\hat{\pi}_D} + \eta^{\pi_D,\hat{\pi}_D} - \eta^{branch}[\pi] \\
&\geq -2r_{max}[\frac{k(\epsilon_m + \epsilon_\pi)}{1-\gamma} + \frac{\epsilon_\pi}{1-\gamma}] - 2r_{max}[\frac{k\epsilon_\pi}{1-\gamma} + \frac{\epsilon_\pi}{1-\gamma}] \\
&\quad - 2r_{max}[\frac{\gamma^{k+1}}{(1-\gamma)^2} + \frac{\gamma^k\epsilon_\pi}{1-\gamma}] \quad (by\ lemma\ 4) \\
&\geq -2r_{max}[\frac{k(\epsilon_m + 2\epsilon_\pi)}{1-\gamma} + \frac{\gamma^{k+1}\epsilon_\pi}{(1-\gamma)^2} + \frac{\gamma^k\epsilon_\pi}{1-\gamma}]
\end{aligned}$$

$$\Rightarrow \eta[\pi] \geq \eta^{branch}[\pi] - 2r_{max}[\frac{\gamma^{k+1}\epsilon_\pi}{(1-\gamma)^2} + \frac{\gamma^k\epsilon_\pi}{1-\gamma} + \frac{k(\epsilon_m + 2\epsilon_\pi)}{1-\gamma}]$$

□

According to the above bound, we know that we need limited use of truncated model rollout and the algorithm will guarantee monotonic improvement.

Now, I try to extend this result by reducing the number of hyperparameter. I try to find a proper and fixed value for rollout length k. At first, I try to set k to 1 or 15 because authors set k from 1 to 15 in their experiment. If I set k to 15, the bound will become

$$\eta[\pi] \geq \eta^{branch}[\pi] - 2r_{max}[\frac{\gamma^{16}\epsilon_\pi}{(1-\gamma)^2} + \frac{\gamma^{15}\epsilon_\pi}{1-\gamma} + \frac{15(\epsilon_m + 2\epsilon_\pi)}{1-\gamma}].$$

If the above inequality is still hold, the algorithm still guarantees monotonic improvement. However, 15 may be too large to find the policy to hold this inequality. In addition, 15 is too large and the model inaccuracy will be amplified. Now, I try to set k to 1 and the bound is changed to be

$$\eta[\pi] \geq \eta^{branch}[\pi] - 2r_{max}[\frac{\gamma^2\epsilon_\pi}{(1-\gamma)^2} + \frac{\gamma\epsilon_\pi}{1-\gamma} + \frac{(\epsilon_m + 2\epsilon_\pi)}{1-\gamma}].$$

Setting k to 1 is small enough and still consider the model's error in the current distribution. Although it is theoretically possible to directly set k to 1 and reduce the number of the hyperparameter, it still needs to be verified by experiments.

4 Conclusion

In this report, I try to set k to 1 and reduce the number of hyperparameter but it still needs to do the experiment to verify whether it is proper or not. In addition, the authors use SAC which is off-policy algorithm to be the policy optimization algorithm. Maybe we can try to change to use on-policy algorithm such as PPO in the future and see whether performance improves or not. In the problem formulation section, I mention that finding a more precise bound is an interesting problem because more precise bound can help us to design a more efficient model-based algorithm. I find there are some proof about this problem in the SLBO paper(Luo et al. [2018]). However, it is not suitable for improving MBPO because it does not use the concept of branched rollout.

References

Michael Janner, Justin Fu, Marvin Zhang, and Sergey Levine. When to trust your model: Model-based policy optimization. *Advances in Neural Information Processing Systems*, 32, 2019.

Yuping Luo, Huazhe Xu, Yuanzhi Li, Yuandong Tian, Trevor Darrell, and Tengyu Ma. Algorithmic framework for model-based deep reinforcement learning with theoretical guarantees. *arXiv preprint arXiv:1807.03858*, 2018.