ALGORITHMIC FRAMEWORK FOR MODEL-BASED DEEP REINFORCEMENT LEARNING WITH THEORETICAL GUARANTEES

Anonymous authors
Paper under double-blind review

ABSTRACT

Model-based reinforcement learning (RL) is considered to be a promising approach to reduce the sample complexity that hinders model-free RL. However, the theoretical understanding of such methods has been rather limited. This paper introduces a novel algorithmic framework for designing and analyzing model-based RL algorithms with theoretical guarantees. We design a meta-algorithm with a theoretical guarantee of monotone improvement to a local maximum of the expected reward. The meta-algorithm iteratively builds a lower bound of the expected reward based on the estimated dynamical model and sample trajectories, and then maximizes the lower bound jointly over the policy and the model. The framework extends the optimism-in-face-of-uncertainty principle to non-linear dynamical models in a way that requires *no explicit* uncertainty quantification. Instantiating our framework with simplification gives a variant of model-based RL algorithms Stochastic Lower Bounds Optimization (SLBO). Experiments demonstrate that SLBO achieves state-of-the-art performance when only one million or fewer samples are permitted on a range of continuous control benchmark tasks.

1 INTRODUCTION

In recent years deep reinforcement learning has achieved strong empirical success, including superhuman performances on Atari games and Go (Mnih et al., 2015; Silver et al., 2017) and learning locomotion and manipulation skills in robotics (Levine et al., 2016; Schulman et al., 2015b; Lillicrap et al., 2015). Many of these results are achieved by model-free RL algorithms that often require a massive number of samples, and therefore their applications are mostly limited to simulated environments. Model-based deep reinforcement learning, in contrast, exploits the information from state observations explicitly — by planning with an estimated dynamical model — and is considered to be a promising approach to reduce the sample complexity. Indeed, empirical results (Deisenroth & Rasmussen, 2011b; Deisenroth et al., 2013; Levine et al., 2016; Nagabandi et al., 2017; Kurutach et al., 2018; Pong et al., 2018a) have shown strong improvements in sample efficiency.

Despite promising empirical findings, many of *theoretical* properties of model-based deep reinforcement learning are not well-understood. For example, how does the error of the estimated model affect the estimation of the value function and the planning? Can model-based RL algorithms be guaranteed to improve the policy monotonically and converge to a local maximum of the value function? How do we quantify the uncertainty in the dynamical models?

It's challenging to address these questions theoretically in the context of deep RL with continuous state and action space and non-linear dynamical models. Due to the high-dimensionality, learning models from observations in one part of the state space and extrapolating to another part sometimes involves a leap of faith. The uncertainty quantification of the non-linear parameterized dynamical models is difficult — even without the RL components, it is an active but widely-open research area. Prior work in model-based RL mostly quantifies uncertainty with either heuristics or simpler models (Moldovan et al., 2015; Xie et al., 2016; Deisenroth & Rasmussen, 2011a).

Previous theoretical work on model-based RL mostly focuses on either the finite-state MDPs (Jaksch et al., 2010; Bartlett & Tewari, 2009; Fruit et al., 2018; Lakshmanan et al., 2015; Hinderer, 2005; Pirotta et al., 2015; 2013), or the linear parametrization of the dynamics, policy, or value func-

tion (Abbasi-Yadkori & Szepesvári, 2011; Simchowitz et al., 2018; Dean et al., 2017; Sutton et al., 2012; Tamar et al., 2012), but not much on non-linear models. Even with an oracle prediction intervals[1] or posterior estimation, to the best of our knowledge, there was no previous algorithm with convergence guarantees for model-based deep RL.

Towards addressing these challenges, the main contribution of this paper is to propose a novel algorithmic framework for model-based deep RL with theoretical guarantees. Our meta-algorithm (Algorithm 1) extends the optimism-in-face-of-uncertainty principle to non-linear dynamical models in a way that requires *no explicit* uncertainty quantification of the dynamical models.

Let V^π be the value function V^π of a policy π on the true environment, and let $\widehat{V}^\pi$ be the value function of the policy π on the estimated model $\widehat{M}$. We design provable upper bounds, denoted by $D^{\pi,\widehat{M}}$, on how much the error can compound and divert the expected value $\widehat{V}^\pi$ of the imaginary rollouts from their real value V^π, in a neighborhood of some reference policy. Such upper bounds capture the intrinsic difference between the estimated and real dynamical model with respect to the particular reward function under consideration.

The discrepancy bounds $D^{\pi,\widehat{M}}$ naturally leads to a lower bound for the true value function:

$$V^\pi \geq \widehat{V}^\pi - D^{\pi,\widehat{M}}\,. \tag{1.1}$$

Our algorithm iteratively collects batches of samples from the interactions with environments, builds the lower bound above, and then maximizes it over both the dynamical model $\widehat{M}$ and the policy π. We can use any RL algorithms to optimize the lower bounds, because it will be designed to only depend on the sample trajectories from a fixed reference policy (as opposed to requiring new interactions with the policy iterate.)

We show that the performance of the policy is guaranteed to monotonically increase, assuming the optimization within each iteration succeeds (see Theorem 3.1.) To the best of our knowledge, this is the first theoretical guarantee of monotone improvement for model-based deep RL.

Readers may have realized that optimizing a robust lower bound is reminiscent of robust control and robust optimization. The distinction is that we optimistically and iteratively maximize the RHS of (1.1) jointly over the model and the policy. The iterative approach allows the algorithms to collect higher quality trajectory adaptively, and the optimism in model optimization encourages explorations of the parts of space that are not covered by the current discrepancy bounds.

To instantiate the meta-algorithm, we design a few valid discrepancy bounds in Section 4. In Section 4.1, we recover the norm-based model loss by imposing the additional assumption of a Lipschitz value function. The result suggests a norm is preferred compared to the square of the norm. Indeed in Section 6.2, we show that experimentally learning with ℓ_2 loss significantly outperforms the mean-squared error loss (ℓ_2^2).

In Section 4.2, we design a discrepancy bound that is *invariant* to the representation of the state space. Here we measure the loss of the model by the difference between the value of the predicted next state and the value of the true next state. Such a loss function is shown to be invariant to one-to-one transformation of the state space. Thus we argue that the loss is an intrinsic measure for the model error without any information beyond observing the rewards. We also refine our bounds in Section A by utilizing some mathematical tools of measuring the difference between policies in χ^2-divergence (instead of KL divergence or TV distance).

Our analysis also sheds light on the comparison between model-based RL and on-policy model-free RL algorithms such as policy gradient or TRPO (Schulman et al., 2015a). The RHS of equation (1.1) is likely to be a good approximator of V^π in a larger neighborhood than the linear approximation of V^π used in policy gradient is (see Remark 4.5.)

Finally, inspired by our framework and analysis, we design a variant of model-based RL algorithms Stochastic Lower Bounds Optimization (SLBO). Experiments demonstrate that SLBO achieves state-of-the-art performance when only 1M samples are permitted on a range of continuous control benchmark tasks.

[1]We note that the confidence interval of parameters are likely meaningless for over-parameterized neural networks models.

2 NOTATIONS AND PRELIMINARIES

We denote the state space by $\mathcal{S}$, the action space by $\mathcal{A}$. A policy $\pi(\cdot|s)$ specifies the conditional distribution over the action space given a state s. A dynamical model $M(\cdot|s,a)$ specifies the conditional distribution of the next state given the current state s and action a. We will use $M^\star$ globally to denote the unknown true dynamical model. Our target applications are problems with the continuous state and action space, although the results apply to discrete state or action space as well. When the model is deterministic, $M(\cdot|s,a)$ is a dirac measure. In this case, we use $M(s,a)$ to denote the unique value of s' and view M as a function from $\mathcal{S}\times\mathcal{A}$ to $\mathcal{S}$. Let $\mathcal{M}$ denote a (parameterized) family of models that we are interested in, and Π denote a (parameterized) family of policies.

Unless otherwise stated, for random variable X, we will use p_X to denote its density function.

Let S_0 be the random variable for the initial state. Let $S_t^{\pi,M}$ to denote the random variable of the states at steps t when we execute the policy π on the dynamic model M stating with S_0. Note that $S_0^{\pi,M}=S_0$ unless otherwise stated. We will omit the subscript when it's clear from the context. We use A_t to denote the actions at step t similarly. We often use τ to denote the random variable for the trajectory $(S_0,A_1,\dots,S_t,A_t,\dots)$. Let $R(s,a)$ be the reward function at each step. We assume R is *known* throughout the paper, although R can be also considered as part of the model if unknown. Let γ be the discount factor.

Let $V^{\pi,M}$ be the value function on the model M and policy π defined as:

$$V^{\pi,M}(s)=\mathop{\mathbb{E}}_{\substack{\forall t\geq 0, A_t\sim\pi(\cdot|S_t)\\ S_{t+1}\sim M(\cdot|S_t,A_t)}}\left[\sum_{t=0}^{\infty}\gamma^t R(S_t,A_t)\mid S_0=s\right] \tag{2.1}$$

We define $V^{\pi,M}=\mathbb{E}\left[V^{\pi,M}(S_0)\right]$ as the expected reward-to-go at Step 0 (averaged over the random initial states). Our goal is to maximize the reward-to-go on the true dynamical model, that is, $V^{\pi,M^\star}$, over the policy π. For simplicity, throughout the paper, we set $\kappa=\gamma(1-\gamma)^{-1}$ since it occurs frequently in our equations. Every policy π induces a distribution of states visited by policy π:

Definition 2.1. For a policy π, define $\rho^{\pi,M}$ as the discounted distribution of the states visited by π on M. Let ρ^π be a shorthand for $\rho^{\pi,M^\star}$ and we omit the superscript $M^\star$ throughout the paper. Concretely,we have $\rho^\pi=(1-\gamma)\sum_{t=0}^{\infty}\gamma^t\cdot p_{S_t^\pi}$

3 ALGORITHMIC FRAMEWORK

As mentioned in the introduction, towards optimizing $V^{\pi,M^\star}$,[2] our plan is to build a lower bound for $V^{\pi,M^\star}$ of the following type and optimize it iteratively:

$$V^{\pi,M^\star}\geq V^{\pi,\widehat{M}}-D(\widehat{M},\pi) \tag{3.1}$$

where $D(\widehat{M},\pi)\in\mathbb{R}_{\geq 0}$ bounds from above the discrepancy between $V^{\pi,\widehat{M}}$ and $V^{\pi,M^\star}$. Building such an optimizable discrepancy bound globally that holds for all $\widehat{M}$ and π turns out to be rather difficult, if not impossible. Instead, we shoot for establishing such a bound over the neighborhood of a reference policy π_{ref}.

$$V^{\pi,M^\star}\geq V^{\pi,\widehat{M}}-D_{\pi_{\text{ref}},\delta}(\widehat{M},\pi),\qquad \forall\pi \text{ s.t. } d(\pi,\pi_{\text{ref}})\leq\delta \tag{R1}$$

Here $d(\cdot,\cdot)$ is a function that measures the closeness of two policies, which will be chosen later in alignment with the choice of D. We will mostly omit the subscript δ in D for simplicity in the rest of the paper. We will require our discrepancy bound to vanish when $\widehat{M}$ is an accurate model:

$$\widehat{M}=M^\star\Longrightarrow D_{\pi_{\text{ref}}}(\widehat{M},\pi)=0,\quad\forall\pi,\pi_{\text{ref}} \tag{R2}$$

The third requirement for the discrepancy bound D is that it can be estimated and optimized in the sense that

$$D_{\pi_{\text{ref}}}(\widehat{M},\pi)\text{ is of the form }\mathop{\mathbb{E}}_{\tau\sim\pi_{\text{ref}},M^\star}[f(\widehat{M},\pi,\tau)] \tag{R3}$$

[2]Note that in the introduction we used V^π for simplicity, and in the rest of the paper we will make the dependency on $M^\star$ explicit.

where f is a known differentiable function. We can estimate such discrepancy bounds for *every* π in the neighborhood of π_{ref} by sampling empirical trajectories $\tau^{(1)}, \ldots, \tau^{(n)}$ from executing policy π_{ref} on the real environment $M^\star$ and compute the average of $f(\widehat{M}, \pi, \tau^{(i)})$'s. We would have to insist that the expectation cannot be over the randomness of trajectories from π on $M^\star$, because then we would have to re-sample trajectories for every possible π encountered.

For example, assuming the dynamical models are all deterministic, one of the valid discrepancy bounds (under some strong assumptions) that will prove in Section 4 is a multiple of the error of the prediction of $\widehat{M}$ on the trajectories from π_{ref}:

$$D_{\pi_{\text{ref}}}(\widehat{M}, \pi) = L \cdot \mathop{\mathbb{E}}_{S_0,\ldots,S_t,\sim\pi_{\text{ref}},M^\star} \left[\|\widehat{M}(S_t) - S_{t+1}\| \right] \tag{3.2}$$

Suppose we can establish such an discrepancy bound D (and the distance function d) with properties (R1), (R2), and (R3), — which will be the main focus of Section 4 —, then we can devise the following meta-algorithm (Algorithm 1). We iteratively optimize the lower bound over the policy π_{k+1} and the model M_{k+1}, subject to the constraint that the policy is not very far from the reference policy π_k obtained in the previous iteration. For simplicity, we only state the population version with the exact computation of $D_{\pi_{\text{ref}}}(\widehat{M}, \pi)$, though empirically it is estimated by sampling trajectories.

Algorithm 1 Meta-Algorithm for Model-based RL

Inputs: Initial policy π_0. Discrepancy bound D and distance function d that satisfy equation (R1) and (R2).
For $k = 0$ to T:

$$\pi_{k+1}, M_{k+1} = \mathop{\text{argmax}}_{\pi\in\Pi,\ M\in\mathcal{M}} \quad V^{\pi,M} - D_{\pi_k,\delta}(M, \pi) \tag{3.3}$$

$$\text{s.t. } d(\pi, \pi_k) \le \delta \tag{3.4}$$

We first remark that the discrepancy bound $D_{\pi_k}(M, \pi)$ in the objective plays the role of learning the dynamical model by ensuring the model to fit to the sampled trajectories. For example, using the discrepancy bound in the form of equation (3.2), we roughly recover the standard objective for model learning, with the caveat that we only have the norm instead of the square of the norm in MSE. Such distinction turns out to be empirically important for better performance (see Section 6.2).

Second, our algorithm can be viewed as an extension of the optimism-in-face-of-uncertainty (OFU) principle to non-linear parameterized setting: jointly optimizing M and π encourages the algorithm to choose the most optimistic model among those that can be used to accurately estimate the value function. (See (Jaksch et al., 2010; Bartlett & Tewari, 2009; Fruit et al., 2018; Lakshmanan et al., 2015; Pirotta et al., 2015; 2013) and references therein for the OFU principle in finite-state MDPs.) The main novelty here is to optimize the lower bound directly, without explicitly building any confidence intervals, which turns out to be challenging in deep learning. In other words, the uncertainty is measured straightforwardly by how the error would affect the estimation of the value function.

Thirdly, the maximization of $V^{\pi,M}$, when M is fixed, can be solved by any model-free RL algorithms with M as the environment without querying any real samples. Optimizing $V^{\pi,M}$ jointly over π, M can be also viewed as another RL problem with an extended actions space using the known "extended MDP technique". See (Jaksch et al., 2010, section 3.1) for details.

Our main theorem shows formally that the policy performance in the real environment is non-decreasing under the assumption that the real dynamics belongs to our parameterized family $\mathcal{M}$.[3]

Theorem 3.1. *Suppose that $M^\star \in \mathcal{M}$, that D and d satisfy equation* (R1) *and* (R2)*, and the optimization problem in equation* (3.3) *is solvable at each iteration. Then, Algorithm 1 produces a sequence of policies $\pi_0, \ldots, \pi_T$ with monotonically increasing values:*

$$V^{\pi_0,M^\star} \le V^{\pi_1,M^\star} \le \cdots \le V^{\pi_T,M^\star} \tag{3.5}$$

[3]We note that such an assumption, though restricted, may not be very far from reality: optimistically speaking, we only need to approximate the dynamical model accurately on the trajectories of the optimal policy. This might be much easier than approximating the dynamical model globally.

Moreover, as $k \to \infty$, the value $V^{\pi_k, M^\star}$ converges to some $V^{\bar{\pi}, M^\star}$, where $\bar{\pi}$ is a local maximum of $V^{\pi, M^\star}$ in domain Π.

The theorem above can also be extended to a finite sample complexity result with standard concentration inequalities. We show in Theorem G.2 that we can obtain an approximate local maximum in $O(1/\varepsilon)$ iterations with sample complexity (in the number of trajectories) that is polynomial in dimension and accuracy ε and is logarithmic in certain smoothness parameters.

Proof of Theorem 3.1. Since D and d satisfy equation (R1), we have that

$$V^{\pi_{k+1}, M^\star} \geq V^{\pi_{k+1}, M_{k+1}} - D_{\pi_k}(M_{k+1}, \pi_{k+1})$$

By the definition that π_{k+1} and M_{k+1} are the optimizers of equation (3.3), we have that

$$V^{\pi_{k+1}, M_{k+1}} - D_{\pi_k}(M_{k+1}, \pi_{k+1}) \geq V^{\pi_k, M^\star} - D_{\pi_k}(M^\star, \pi_k) = V^{\pi_k, M^\star} \qquad \text{(by equation R2)}$$

Combing the two equations above we complete the proof of equation (3.5).

For the second part of the theorem, by compactness, we have that a subsequence of π_k converges to some $\bar{\pi}$. By the monotonicity we have $V^{\pi_k, M^\star} \leq V^{\bar{\pi}, M^\star}$ for every $k \geq 0$. For the sake of contradiction, we assume $\bar{\pi}$ is a not a local maximum, then in the neighborhood of $\bar{\pi}$ there exists π' such that $V^{\pi', M^\star} > V^{\bar{\pi}, M^\star}$ and $d(\bar{\pi}, \pi') < \delta/2$. Let t be such that π_t is in the $\delta/2$-neighborhood of $\bar{\pi}$. Then we see that $(\pi', M^\star)$ is a better solution than (π_{t+1}, M_{t+1}) for the optimization problem (3.3) in iteration t because $V^{\pi', M^\star} > V^{\bar{\pi}, M^\star} \geq V^{\pi_{t+1}, M^\star} \geq V^{\pi_{t+1}, M_{t+1}} - D_{\pi_t}(M_{t+1}, \pi_{t+1})$. (Here the last inequality uses equation (R1) with π_t as π_{ref}.) The fact $(\pi', M^\star)$ is a strictly better solution than (π_{t+1}, M_{t+1}) contradicts the fact that (π_{t+1}, M_{t+1}) is defined to be the optimal solution of (3.3) . Therefore $\bar{\pi}$ is a local maximum and we complete the proof.

□

4 Discrepancy Bounds Design

In this section, we design discrepancy bounds that can provably satisfy the requirements (R1), (R2), and (R3). We design increasingly stronger discrepancy bounds from Section 4.1 to Section A.

4.1 Norm-based prediction error bounds

In this subsection, we assume the dynamical model $M^\star$ is deterministic and we also learn with a deterministic model $\widehat{M}$. Under assumptions defined below, we derive a discrepancy bound D of the form $\|\widehat{M}(S, A) - M^\star(S, A)\|$ averaged over the observed state-action pair (S, A) on the dynamical model $\widehat{M}$. This suggests that the norm is a better metric than the mean-squared error for learning the model, which is empirically shown in Section 6.2. Through the derivation, we will also introduce a telescoping lemma, which serves as the main building block towards other finer discrepancy bounds.

We make the (strong) assumption that the value function $V^{\pi, \widehat{M}}$ on the estimated dynamical model is L-Lipschitz w.r.t to some norm $\|\cdot\|$ in the sense that

$$\forall s, s' \in \mathcal{S}, \left|V^{\pi, \widehat{M}}(s) - V^{\pi, \widehat{M}}(s')\right| \leq L \cdot \|s - s'\| \tag{4.1}$$

In other words, nearby starting points should give reward-to-go under the same policy π. We note that not every real environment $M^\star$ has this property, let alone the estimated dynamical models. However, once the real dynamical model induces a Lipschitz value function, we may penalize the Lipschitz-ness of the value function of the estimated model during the training.

We start off with a lemma showing that the expected prediction error is an upper bound of the discrepancy between the real and imaginary values.

Lemma 4.1. *Suppose $V^{\pi, \widehat{M}}$ is L-Lipschitz (in the sense of* (4.1)*). Recall $\kappa = \gamma(1-\gamma)^{-1}$.*

$$\left|V^{\pi, \widehat{M}} - V^{\pi, M^\star}\right| \leq \kappa L \mathop{\mathbb{E}}_{\substack{S \sim \rho^\pi \\ A \sim \pi(\cdot|S)}} \left[\|\widehat{M}(S, A) - M^\star(S, A)\|\right] \tag{4.2}$$

However, in RHS in equation 4.2 cannot serve as a discrepancy bound because it does not satisfy the requirement (R3) — to optimize it over π we need to collect samples from ρ^π for every iterate π — the state distribution of the policy π on the *real* model $M^\star$. The main proposition of this subsection stated next shows that for every π in the neighborhood of a reference policy π_{ref}, we can replace the distribution ρ^π be a fixed distribution $\rho^{\pi_{\text{ref}}}$ with incurring only a higher order approximation. We use the expected KL divergence between two π and π_{ref} to define the neighborhood:

$$d^{\text{KL}}(\pi, \pi_{\text{ref}}) = \mathop{\mathbb{E}}_{S\sim\rho^\pi}\left[KL(\pi(\cdot|S), \pi_{\text{ref}}(\cdot|S))^{1/2}\right] \tag{4.3}$$

Proposition 4.2. *In the same setting of Lemma 4.1, assume in addition that π is close to a reference policy π_{ref} in the sense that $d^{\text{KL}}(\pi, \pi_{\text{ref}}) \le \delta$, and that the states in $\mathcal{S}$ are uniformly bounded in the sense that $\|s\| \le B, \forall s \in \mathcal{S}$. Then,*

$$\left|V^{\pi,\widehat{M}} - V^{\pi,M^\star}\right| \le \kappa L \mathop{\mathbb{E}}_{\substack{S\sim\rho^{\pi_{\text{ref}}}\\ A\sim\pi(\cdot|S)}}\left[\|\widehat{M}(S,A) - M^\star(S,A)\|\right] + 2\kappa^2\delta B \tag{4.4}$$

In a benign scenario, the second term in the RHS of equation (4.4) should be dominated by the first term when the neighborhood size δ is sufficiently small. Moreover, the term B can also be replaced by $\max_{S,A}\|\widehat{M}(S,A) - M^\star(S,A)\|$ (see the proof that is deferred to Section C.). The dependency on κ may not be tight for real-life instances, but we note that most analysis of similar nature loses the additional κ factor Schulman et al. (2015a); Achiam et al. (2017), and it's inevitable in the worst-case.

A telescoping lemma. Towards proving Propositions 4.2 and deriving stronger discrepancy bound, we define the following quantity that captures the discrepancy between $\widehat{M}$ and $M^\star$ on a single state-action pair (s, a).

$$G^{\pi,\widehat{M}}(s,a) = \mathop{\mathbb{E}}_{\hat{s}'\sim\widehat{M}(\cdot|s,a)} V^{\pi,\widehat{M}}(\hat{s}') - \mathop{\mathbb{E}}_{s'\sim M^\star(\cdot|s,a)} V^{\pi,\widehat{M}}(s') \tag{4.5}$$

Note that if $M, \widehat{M}$ are deterministic, then $G^{\pi,\widehat{M}}(s,a) = V^{\pi,\widehat{M}}(\widehat{M}(s,a)) - V^{\pi,\widehat{M}}(M^\star(s,a))$. We give a telescoping lemma that decompose the discrepancy between $V^{\pi,M}$ and $V^{\pi,M^\star}$ into the expected single-step discrepancy G.

Lemma 4.3. *[Telescoping Lemma] Recall that $\kappa := \gamma(1-\gamma)^{-1}$. For any policy π and dynamical models $M, \widehat{M}$, we have that*

$$V^{\pi,\widehat{M}} - V^{\pi,M} = \kappa \mathop{\mathbb{E}}_{\substack{S\sim\rho^{\pi,M}\\ A\sim\pi(\cdot|S)}}\left[G^{\pi,\widehat{M}}(S,A)\right] \tag{4.6}$$

The proof is reminiscent of the telescoping expansion in Kakade & Langford (2002) (c.f. Schulman et al. (2015a)) for characterizing the value difference of two policies, but we apply it to deal with the discrepancy between models. The detail is deferred to Section B. With the telescoping Lemma 4.3, Proposition 4.1 follows straightforwardly from Lipschitzness of the imaginary value function. Proposition 4.2 follows from that ρ^π and $\rho^{\pi_{\text{ref}}}$ are close. We defer the proof to Appendix C.

4.2 Representation-invariant Discrepancy Bounds

The main limitation of the norm-based discrepancy bounds in previous subsection is that it depends on the state representation. Let $\mathcal{T}$ be a one-to-one map from the state space $\mathcal{S}$ to some other space $\mathcal{S}'$, and for simplicity of this discussion let's assume a model M is deterministic. Then if we represent every state s by its transformed representation $\mathcal{T}s$, then the transformed model $M^{\mathcal{T}}$ defined as $M^{\mathcal{T}}(s,a) \triangleq \mathcal{T}M(\mathcal{T}^{-1}s, a)$ together with the transformed reward $R^{\mathcal{T}}(s,a) \triangleq R(\mathcal{T}^{-1}s, a)$ and transformed policy $\pi^{\mathcal{T}}(s) \triangleq \pi(\mathcal{T}^{-1}s)$ is equivalent to the original set of the model, reward, and policy in terms of the performance (Lemma C.1). Thus such transformation $\mathcal{T}$ is not identifiable from only observing the reward. However, the norm in the state space is a notion that depends on the hidden choice of the transformation $\mathcal{T}$. [4]

[4] That said, in many cases the reward function itself is known, and the states have physical meanings, and therefore we may be able to use the domain knowledge to figure out the best norm.

Another limitation is that the loss for the model learning should also depend on the state itself instead of only on the difference $\widehat{M}(S,A) - M^\star(S,A)$. It is possible that when S is at a critical position, the prediction error needs to be highly accurate so that the model $\widehat{M}$ can be useful for planning. On the other hand, at other states, the dynamical model is allowed to make bigger mistakes because they are not essential to the reward.

We propose the following discrepancy bound towards addressing the limitations above. Recall the definition of $G^{\pi,\widehat{M}}(s,a) = V^{\pi,\widehat{M}}(\widehat{M}(s,a)) - V^{\pi,\widehat{M}}(M^\star(s,a))$ which measures the difference between $\widehat{M}(s,a))$ and $M^\star(s,a)$ according to their imaginary rewards. We construct a discrepancy bound using the absolute value of G. Let's define ε_1 and $\varepsilon_{\max}$ as the average of $|G^{\pi,\hat{M}}|$ and its maximum: $\varepsilon_1 = \mathbb{E}_{S\sim\rho^{\pi_{\text{ref}}}}\left[\left|G^{\pi,\widehat{M}}(S,A)\right|\right]$ and $\varepsilon_{\max} = \max_S \left|G^{\pi,\widehat{M}}(S)\right|$ where $G^{\pi,\widehat{M}}(S) = \mathbb{E}_{A\sim\pi}\left[G^{\pi,\widehat{M}}(S,A)\right]$. We will show that the following discrepancy bound $D^G_{\pi_{\text{ref}}}(\widehat{M},\pi)$ satisfies the property (R1), (R2).

$$D^G_{\pi_{\text{ref}}}(\widehat{M},\pi) = \kappa\cdot\varepsilon_1 + \kappa^2\delta\varepsilon_{\max} \tag{4.7}$$

Proposition 4.4. *Let d^{KL} and D^G be defined as in equation* (4.3) *and* (4.7)*. Then the choice $d = d^{\text{KL}}$ and $D = D^G$ satisfies the basic requirements (equation* (R1) *and* (R2)*). Moreover, G is invariant w.r.t any one-to-one transformation of the state space (in the sense of equation C.1 in the proof).*

The proof follows from the telescoping lemma (Lemma 4.3) and is deferred to Section C. We remark that the first term $\kappa\varepsilon_1$ can in principle be estimated and optimized approximately: the expectation be replaced by empirical samples from $\rho^{\pi_{\text{ref}}}$, and $G^{\pi,\hat{M}}$ is an analytical function of π and $\widehat{M}$ when they are both deterministic, and therefore can be optimized by back-propagation through time (BPTT). (When π and $\widehat{M}$ and are stochastic with a re-parameterizable noise such as Gaussian distribution Kingma & Welling (2013), we can also use back-propagation to estimate the gradient.) The second term in equation (4.7) is difficult to optimize because it involves the maximum. However, it can be in theory considered as a second-order term because δ can be chosen to be a fairly small number. (In the refined bound in Section A, the dependency on δ is even milder.)

Remark 4.5. Proposition 4.4 intuitively suggests a technical reason of why model-based approach can be more sample-efficient than policy gradient based algorithms such as TRPO or PPO (Schulman et al., 2015a; 2017). The approximation error of $V^{\pi,\widehat{M}}$ in model-based approach decreases as the model error $\varepsilon_1, \varepsilon_{\max}$ decrease or the neighborhood size δ decreases, whereas the approximation error in policy gradient only linearly depends on the the neighborhood size Schulman et al. (2015a). In other words, model-based algorithms can trade model accuracy for a larger neighborhood size, and therefore the convergence can be faster (in terms of outer iterations.) This is consistent with our empirical observation that the model can be accurate in a descent neighborhood of the current policy so that the constraint (3.4) can be empirically dropped. We also refine our bonds in Section A, where the discrepancy bounds is proved to decay faster in δ.

5 Additional Related work

Model-based reinforcement learning is expected to require fewer samples than model-free algorithms (Deisenroth et al., 2013) and has been successfully applied to robotics in both simulation and in the real world (Deisenroth & Rasmussen, 2011b; Morimoto & Atkeson, 2003; Deisenroth et al., 2011) using dynamical models ranging from Gaussian process (Deisenroth & Rasmussen, 2011b; Ko & Fox, 2009), time-varying linear models (Levine & Koltun, 2013; Lioutikov et al., 2014; Levine & Abbeel, 2014; Yip & Camarillo, 2014), mixture of Gaussians (Khansari-Zadeh & Billard, 2011), to neural networks (Hunt et al., 1992; Nagabandi et al., 2017; Kurutach et al., 2018; Tangkaratt et al., 2014; Sanchez-Gonzalez et al., 2018; Pascanu et al., 2017). In particular, the work of Kurutach et al. (2018) uses an ensemble of neural networks to learn the dynamical model, and significantly reduces the sample complexity compared to model-free approaches. The work of Chua et al. (2018) makes further improvement by using a probabilistic model ensemble. Clavera et al. (Clavera et al., 2018) extended this method with meta-policy optimization and improve the robustness to model error. In contrast, we focus on theoretical understanding of model-based RL and the design of new algorithms, and our experiments use a *single* neural network to estimate the dynamical model.

Our discrepancy bound in Section 4 is closely related to the work (Farahmand et al., 2017) on the value-aware model loss. Our approach differs from it in three details: a) we use the absolute value of the value difference instead of the squared difference; b) we use the imaginary value function from the estimated dynamical model to define the loss, which makes the loss purely a function of the estimated model and the policy; c) we show that the iterative algorithm, using the loss function as a building block, can converge to a local maximum, partly by cause of the particular choices made in a) and b). Asadi et al. (2018) also study the discrepancy bounds under Lipschitz condition of the MDP.

Prior work explores a variety of ways of combining model-free and model-based ideas to achieve the best of the two methods (Sutton, 1991; 1990; Racanière et al., 2017; Mordatch et al., 2016; Sun et al., 2018). For example, estimated models (Levine & Koltun, 2013; Gu et al., 2016; Kalweit & Boedecker, 2017) are used to enrich the replay buffer in the model-free off-policy RL. Pong et al. (2018b) proposes goal-conditioned value functions trained by model-free algorithms and uses it for model-based controls. Feinberg et al. (2018); Buckman et al. (2018) use dynamical models to improve the estimation of the value functions in the model-free algorithms.

On the control theory side, Dean et al. (2018; 2017) provide strong finite sample complexity bounds for solving linear quadratic regulator using model-based approach. Boczar et al. (2018) provide finite-data guarantees for the "coarse-ID control" pipeline, which is composed of a system identification step followed by a robust controller synthesis procedure. Our method is inspired by the general idea of maximizing a low bound of the reward in (Dean et al., 2017). By contrast, our work applies to non-linear dynamical systems. Our algorithms also estimate the models iteratively based on trajectory samples from the learned policies.

Strong model-based and model-free sample complexity bounds have been achieved in the tabular case (finite state space). We refer the readers to (Kakade et al., 2018; Dann et al., 2017; Szita & Szepesvári, 2010; Kearns & Singh, 2002; Jaksch et al., 2010; Agrawal & Jia, 2017) and the reference therein. Our work focus on continuous and high-dimensional state space (though the results also apply to tabular case).

Another line of work of model-based reinforcement learning is to learn a dynamic model in a hidden representation space, which is especially necessary for pixel state spaces (Kakade et al., 2018; Dann et al., 2017; Szita & Szepesvári, 2010; Kearns & Singh, 2002; Jaksch et al., 2010). Srinivas et al. (2018) shows the possibility to learn an abstract transition model to imitate expert policy. Oh et al. (2017) learns the hidden state of a dynamical model to predict the value of the future states and applies RL or planning on top of it. Serban et al. (2018); Ha & Schmidhuber (2018) learns a bottleneck representation of the states. Our framework can be potentially combined with this line of research.

6 Practical Implementation and Experiments

6.1 Practical implementation

We design with simplification of our framework a variant of model-based RL algorithms, Stochastic Lower Bound Optimization (SLBO). First, we removed the constraints (3.4). Second, we stop the gradient w.r.t M (but not π) from the occurrence of M in $V^{\pi,M}$ in equation (3.3) (and thus our practical implementation is not optimism-driven.)

Extending the discrepancy bound in Section 4.1, we use a multi-step prediction loss for learning the models with ℓ_2 norm. For a state s_t and action sequence $a_{t:t+h}$, we define the h-step prediction $\hat{s}_{t+h}$ as $\hat{s}_t = s_t,$ and for $h \geq 0, \hat{s}_{t+h+1} = \widehat{M}_\phi(\hat{s}_{t+h}, a_{t+h})$, The *$H$-step loss* is then defined as

$$\mathcal{L}_\phi^{(H)}((s_{t:t+h}, a_{t:t+h}); \phi) = \frac{1}{H} \sum_{i=1}^{H} \|\hat{s}_{t+i} - s_{t+i}\|_2. \tag{6.1}$$

A similar loss is also used in Nagabandi et al. (2017) for validation. We note that motivation by the theory in Section 4.1, we use ℓ_2-norm instead of the square of ℓ_2 norm. The loss function we attempt to optimize at iteration k is thus[5]

$$\max_{\phi,\theta} \; V^{\pi_\theta, \mathrm{sg}(\widehat{M}_\phi)} - \lambda \mathop{\mathbb{E}}_{(s_{t:t+h}, a_{t:t+h}) \sim \pi_k, M^\star} \left[\mathcal{L}_\phi^{(H)}((s_{t:t+h}, a_{t:t+h}); \phi) \right] \tag{6.2}$$

[5]This is technically not a well-defined mathematical objective. The sg operation means identity when the function is evaluated, whereas when computing the update, $\mathrm{sg}(M_\phi)$ is considered fixed.

where λ is a tunable parameter and sg denotes the stop gradient operation.

We note that the term $V^{\pi_\theta,\mathrm{sg}(\widehat{M}_\phi)}$ depends on both the parameter θ and the parameter ϕ but there is no gradient passed through ϕ, whereas $\mathcal{L}_\phi^{(H)}$ only depends on the ϕ. We optimize equation (6.2) by alternatively maximizing $V^{\pi_\theta,\mathrm{sg}(\widehat{M}_\phi)}$ and minimizing $\mathcal{L}_\phi^{(H)}$: for the former, we use TRPO with samples from the estimated dynamical model $\widehat{M}_\phi$ (by treating $\widehat{M}_\phi$ as a fixed simulator), and for the latter we use standard stochastic gradient methods. Algorithm 2 gives a pseudo-code for the algorithm. The n_{model} and n_{policy} iterations are used to balance the number of steps of TRPO and Adam updates within the loop indexed by n_{inner}.[6]

Algorithm 2 Stochastic Lower Bound Optimization (SLBO)

1: Initialize model network parameters ϕ and policy network parameters θ
2: Initialize dataset $\mathcal{D} \leftarrow \emptyset$
3: **for** n_{outer} iterations **do**
4: $\quad\mathcal{D} \leftarrow \mathcal{D} \cup \{$ collect n_{collect} samples from real environment using π_θ with noises $\}$
5: $\quad$**for** n_{inner} iterations **do** $\quad\triangleright$ optimize (6.2) with stochastic alternating updates
6: $\quad\quad$**for** n_{model} iterations **do**
7: $\quad\quad\quad$optimize (6.1) over ϕ with sampled data from $\mathcal{D}$ by one step of Adam
8: $\quad\quad$**for** n_{policy} iterations **do**
9: $\quad\quad\quad\mathcal{D}' \leftarrow \{$ collect n_{trpo} samples using $\widehat{M}_\phi$ as dynamics $\}$
10: $\quad\quad\quad$optimize π_θ by running TRPO on $\mathcal{D}'$

Power of stochasticity and connection to standard MB RL: We identify the main advantage of our algorithms over standard model-based RL algorithms is that we alternate the updates of the model and the policy within an outer iteration. By contrast, most of the existing model-based RL methods only optimize the models once (for a lot of steps) after collecting a batch of samples (see Algorithm 3 for an example). The stochasticity introduced from the alternation with stochastic samples seems to dramatically reduce the overfitting (of the policy to the estimated dynamical model) in a way similar to that SGD regularizes ordinary supervised training. [7] Another way to view the algorithm is that the model obtained from line 7 of Algorithm 2 at different inner iteration serves as an ensemble of models. We do believe that a cleaner and easier instantiation of our framework (with optimism) exists, and the current version, though performing very well, is not necessarily the best implementation.

Entropy regularization: An additional component we apply to SLBO is the commonly-adopted entropy regularization in policy gradient method (Williams & Peng, 1991; Mnih et al., 2016), which was found to significantly boost the performance in our experiments (ablation study in Appendix F.5). Specifically, an additional entropy term is added to the objective function in TRPO. We hypothesize that entropy bonus helps exploration, diversifies the collected data, and thus prevents overfitting.

6.2 Experimental Results

We evaluate our algorithm SLBO (Algorithm 2) on five continuous control tasks from rllab (Duan et al., 2016), including Swimmer, Half Cheetah, Humanoid, Ant, Walker. All environments that we test have a maximum horizon of 500, which is longer than most of the existing model-based RL work (Nagabandi et al., 2017; Kurutach et al., 2018). (Environments with longer horizons are commonly harder to train.) More details can be found in Appendix F.1.

Baselines. We compare our algorithm with 3 other algorithms including: (1) Soft Actor-Critic (SAC) (Haarnoja et al., 2018), the state-of-the-art model-free off-policy algorithm in sample efficiency; (2) Trust-Region Policy Optimization (TRPO) (Schulman et al., 2015a), a policy-gradient based

[6] In principle, to balance the number of steps, it suffices to take one of n_{model} and n_{policy} to be 1. However, empirically we found the optimal balance is achieved with larger n_{model} and n_{policy}, possibly due to complicated interactions between the two optimization problem.

[7] Similar stochasticity can potentially be obtained by an extreme hyperparameter choice of the standard MB RL algorithm: in each outer iteration of Algorithm 3, we only sample a very small number of trajectories and take a few model updates and policy updates. We argue our interpretation of stochastic optimization of the lower bound (6.2) is more natural in that it reveals the regularization from stochastic optimization.

algorithm; and (3) Model-Based TRPO, a standard model-based algorithm described in Algorithm 3. Details of these algorithms can be found in Appendix F.4.[8]

The result is shown in Figure 1. In Fig 1, our algorithm shows superior convergence rate (in number of samples) than all the baseline algorithms while achieving better final performance with 1M samples. Specifically, we mark model-free TRPO performance after 8 million steps by the dotted line in Fig 1 and find out that our algorithm can achieve comparable or better final performance in one million steps. For ablation study, we also add the performance of SLBO-MSE, which corresponds to running SLBO with squared ℓ_2 model loss instead of ℓ_2. SLBO-MSE performs significantly worse than SLBO on four environments, which is consistent with our derived model loss in Section 4.1. We also study the performance of SLBO and baselines with 4 million training samples in F.5. Ablation study of multi-step model training can be found in Appendix F.5.[9]

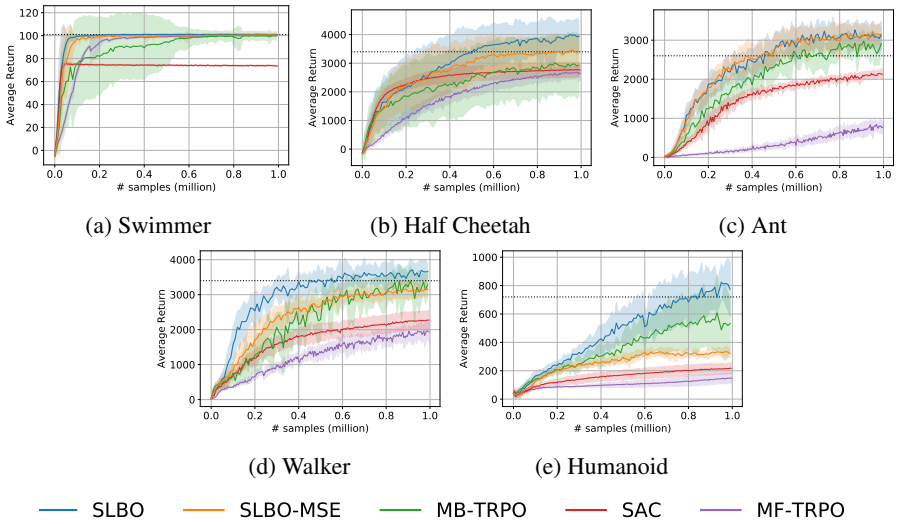


Figure 1: Comparison between SLBO (ours), SLBO with squared ℓ^2 model loss (SLBO-MSE), vanilla model-based TRPO (MB-TRPO), model-free TRPO (MF-TRPO), and Soft Actor-Critic (SAC). We average the results over 10 different random seeds, where the solid lines indicate the mean and shaded areas indicate one standard deviation. The dotted reference lines are the total rewards of MF-TRPO after 8 million steps.

7 Conclusions

We devise a novel algorithmic framework for designing and analyzing model-based RL algorithms with the guarantee to convergence monotonically to a local maximum of the reward. Experimental results show that our proposed algorithm (SLBO) achieves new state-of-the-art performance on several mujoco benchmark tasks when one million or fewer samples are permitted.

A compelling (but obvious) empirical open question then given rise to is whether model-based RL can achieve near-optimal reward on other more complicated tasks or real-world robotic tasks with fewer samples. We believe that understanding the trade-off between optimism and robustness is essential to design more sample-efficient algorithms. Currently, we observed empirically that the optimism-driven part of our proposed meta-algorithm (optimizing $V^{\pi,\widehat{M}}$ over $\widehat{M}$) may lead to instability in the optimization, and therefore don't in general help the performance. It's left for future work to find practical implementation of the optimism-driven approach.

In our theory, we assume that the parameterized model class contains the true dynamical model. Removing this assumption is also another interesting open question. It would be also very interesting if the theoretical analysis can be applied other settings involving model-based approaches (e.g., model-based imitation learning).

[8]We did not have the chance to implement the competitive random search algorithms in (Mania et al., 2018) yet, although our test performance with 500 episode length is higher than theirs with 1000 episode on Half Cheetach (3950 by ours vs 2345 by theirs) and Walker (3650 by ours vs 894 by theirs).

[9]Videos demonstrations are available at https://sites.google.com/view/algombrl/home. A link to the codes will also be available at the same address upon approval.

REFERENCES

Yasin Abbasi-Yadkori and Csaba Szepesvári. Regret bounds for the adaptive control of linear quadratic systems. In *Proceedings of the 24th Annual Conference on Learning Theory*, pp. 1–26, 2011.

Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. *arXiv preprint arXiv:1705.10528*, 2017.

Shipra Agrawal and Randy Jia. Optimistic posterior sampling for reinforcement learning: worst-case regret bounds. In *Advances in Neural Information Processing Systems*, pp. 1184–1194, 2017.

Kavosh Asadi, Dipendra Misra, and Michael L Littman. Lipschitz continuity in model-based reinforcement learning. *arXiv preprint arXiv:1804.07193*, 2018.

Peter L Bartlett and Ambuj Tewari. Regal: A regularization based algorithm for reinforcement learning in weakly communicating mdps. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pp. 35–42. AUAI Press, 2009.

Ross Boczar, Nikolai Matni, and Benjamin Recht. Finite-data performance guarantees for the output-feedback control of an unknown system. *arXiv preprint arXiv:1803.09186*, 2018.

J. Buckman, D. Hafner, G. Tucker, E. Brevdo, and H. Lee. Sample-Efficient Reinforcement Learning with Stochastic Ensemble Value Expansion. *ArXive-prints*, July 2018.

Kurtland Chua, Roberto Calandra, Rowan McAllister, and Sergey Levine. Deep reinforcement learning in a handful of trials using probabilistic dynamics models. *arXiv preprint arXiv:1805.12114*, 2018.

Ignasi Clavera, Jonas Rothfuss, John Schulman, Yasuhiro Fujita, Tamim Asfour, and Pieter Abbeel. Model-based reinforcement learning via meta-policy optimization. *arXiv preprint arXiv:1809.05214*, 2018.

Thomas M Cover and Joy A Thomas. *Elements of information theory*. John Wiley & Sons, 2012.

Christoph Dann, Tor Lattimore, and Emma Brunskill. Unifying pac and regret: Uniform pac bounds for episodic reinforcement learning. In *Advances in Neural Information Processing Systems*, pp. 5713–5723, 2017.

Sarah Dean, Horia Mania, Nikolai Matni, Benjamin Recht, and Stephen Tu. On the sample complexity of the linear quadratic regulator. *arXiv preprint arXiv:1710.01688*, 2017.

Sarah Dean, Horia Mania, Nikolai Matni, Benjamin Recht, and Stephen Tu. Regret bounds for robust adaptive control of the linear quadratic regulator. *arXiv preprint arXiv:1805.09388*, 2018.

Marc Deisenroth and Carl E Rasmussen. Pilco: A model-based and data-efficient approach to policy search. In *Proceedings of the 28th International Conference on machine learning (ICML-11)*, pp. 465–472, 2011a.

Marc Deisenroth and Carl E Rasmussen. Pilco: A model-based and data-efficient approach to policy search. In *Proceedings of the 28th International Conference on machine learning (ICML-11)*, pp. 465–472, 2011b.

Marc Peter Deisenroth, Carl Edward Rasmussen, and Dieter Fox. Learning to control a low-cost manipulator using data-efficient reinforcement learning. 2011.

Marc Peter Deisenroth, Gerhard Neumann, Jan Peters, et al. A survey on policy search for robotics. *Foundations and Trends® in Robotics*, 2(1–2):1–142, 2013.

Yan Duan, Xi Chen, Rein Houthooft, John Schulman, and Pieter Abbeel. Benchmarking deep reinforcement learning for continuous control. In *International Conference on Machine Learning*, pp. 1329–1338, 2016.

Amir-massoud Farahmand, Andre Barreto, and Daniel Nikovski. Value-aware loss function for model-based reinforcement learning. In *Artificial Intelligence and Statistics*, pp. 1486–1494, 2017.

Vladimir Feinberg, Alvin Wan, Ion Stoica, Michael I Jordan, Joseph E Gonzalez, and Sergey Levine. Model-based value estimation for efficient model-free reinforcement learning. *arXiv preprint arXiv:1803.00101*, 2018.

Ronan Fruit, Matteo Pirotta, Alessandro Lazaric, and Ronald Ortner. Efficient bias-span-constrained exploration-exploitation in reinforcement learning. *arXiv preprint arXiv:1802.04020*, 2018.

Shixiang Gu, Timothy Lillicrap, Ilya Sutskever, and Sergey Levine. Continuous deep q-learning with model-based acceleration. In *International Conference on Machine Learning*, pp. 2829–2838, 2016.

David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. *arXiv preprint arXiv:1801.01290*, 2018.

Karl Hinderer. Lipschitz continuity of value functions in markovian decision processes. *Mathematical Methods of Operations Research*, 62(1):3–22, 2005.

K Jetal Hunt, D Sbarbaro, R Żbikowski, and Peter J Gawthrop. Neural networks for control systems—a survey. *Automatica*, 28(6):1083–1112, 1992.

Thomas Jaksch, Ronald Ortner, and Peter Auer. Near-optimal regret bounds for reinforcement learning. *Journal of Machine Learning Research*, 11(Apr):1563–1600, 2010.

S. Kakade, M. Wang, and L. F. Yang. Variance Reduction Methods for Sublinear Reinforcement Learning. *ArXiv e-prints*, February 2018.

Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *ICML*, volume 2, pp. 267–274, 2002.

Gabriel Kalweit and Joschka Boedecker. Uncertainty-driven imagination for continuous deep reinforcement learning. In *Conference on Robot Learning*, pp. 195–206, 2017.

Michael Kearns and Satinder Singh. Near-optimal reinforcement learning in polynomial time. *Machine learning*, 49(2-3):209–232, 2002.

S Mohammad Khansari-Zadeh and Aude Billard. Learning stable nonlinear dynamical systems with gaussian mixture models. *IEEE Transactions on Robotics*, 27(5):943–957, 2011.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Jonathan Ko and Dieter Fox. Gp-bayesfilters: Bayesian filtering using gaussian process prediction and observation models. *Autonomous Robots*, 27(1):75–90, 2009.

Thanard Kurutach, Ignasi Clavera, Yan Duan, Aviv Tamar, and Pieter Abbeel. Model-ensemble trust-region policy optimization. *arXiv preprint arXiv:1802.10592*, 2018.

Kailasam Lakshmanan, Ronald Ortner, and Daniil Ryabko. Improved regret bounds for undiscounted continuous reinforcement learning. In *International Conference on Machine Learning*, pp. 524–532, 2015.

Sergey Levine and Pieter Abbeel. Learning neural network policies with guided policy search under unknown dynamics. In *Advances in Neural Information Processing Systems*, pp. 1071–1079, 2014.

Sergey Levine and Vladlen Koltun. Guided policy search. In *International Conference on Machine Learning*, pp. 1–9, 2013.

Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. End-to-end training of deep visuomotor policies. *The Journal of Machine Learning Research*, 17(1):1334–1373, 2016.

Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

Rudolf Lioutikov, Alexandros Paraschos, Jan Peters, and Gerhard Neumann. Sample-based informationl-theoretic stochastic optimal control. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pp. 3896–3902. IEEE, 2014.

Horia Mania, Aurelia Guy, and Benjamin Recht. Simple random search provides a competitive approach to reinforcement learning. *arXiv preprint arXiv:1803.07055*, 2018.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529, 2015.

Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*, pp. 1928–1937, 2016.

Teodor Mihai Moldovan, Sergey Levine, Michael I Jordan, and Pieter Abbeel. Optimism-driven exploration for nonlinear systems. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pp. 3239–3246. IEEE, 2015.

Igor Mordatch, Nikhil Mishra, Clemens Eppner, and Pieter Abbeel. Combining model-based policy search with online model learning for control of physical humanoids. In *RoboticsandAutomation(ICRA),2016IEEEInternationalConferenceon*, pp. 242–248. IEEE, 2016.

Jun Morimoto and Christopher G Atkeson. Minimax differential dynamic programming: An application to robust biped walking. In *Advances in neural information processing systems*, pp. 1563–1570, 2003.

Anusha Nagabandi, Gregory Kahn, Ronald S Fearing, and Sergey Levine. Neural network dynamics for model-based deep reinforcement learning with model-free fine-tuning. *arXiv preprint arXiv:1708.02596*, 2017.

Frank Nielsen and Richard Nock. On the chi square and higher-order chi distances for approximating f-divergences. *IEEE Signal Processing Letters*, 21(1):10–13, 2014.

Junhyuk Oh, Satinder Singh, and Honglak Lee. Value prediction network. In *Advances in NeuralInformationProcessingSystems*, pp. 6118–6128, 2017.

Razvan Pascanu, Yujia Li, Oriol Vinyals, Nicolas Heess, Lars Buesing, Sebastien Racanière, David Reichert, Théophane Weber, Daan Wierstra, and Peter Battaglia. Learning model-based planning from scratch. *arXiv preprint arXiv:1707.06170*, 2017.

Deepak Pathak, Parsa Mahmoudieh, Guanghao Luo, Pulkit Agrawal, Dian Chen, Yide Shentu, Evan Shelhamer, Jitendra Malik, Alexei A Efros, and Trevor Darrell. Zero-shot visual imitation. In *International Conference on Learning Representations*, 2018.

Matteo Pirotta, Marcello Restelli, and Luca Bascetta. Adaptive step-size for policy gradient methods. In *Advances in Neural Information Processing Systems*, pp. 1394–1402, 2013.

Matteo Pirotta, Marcello Restelli, and Luca Bascetta. Policy gradient in lipschitz markov decision processes. *Machine Learning*, 100(2-3):255–283, 2015.

Vitchyr Pong, Shixiang Gu, Murtaza Dalal, and Sergey Levine. Temporal difference models: Model-free deep rl for model-based control. *arXiv preprint arXiv:1802.09081*, 2018a.

Vitchyr Pong, Shixiang Gu, Murtaza Dalal, and Sergey Levine. Temporal difference models: Model-free deep rl for model-based control. *International Conference on Learning Representations*, 2018b.

Sébastien Racanière, Théophane Weber, David Reichert, Lars Buesing, Arthur Guez, Danilo Jimenez Rezende, Adrià Puigdomènech Badia, Oriol Vinyals, Nicolas Heess, Yujia Li, et al. Imagination-augmented agents for deep reinforcement learning. In *AdvancesinNeuralInformationProcessingSystems*, pp. 5690–5701, 2017.

Alvaro Sanchez-Gonzalez, Nicolas Heess, Jost Tobias Springenberg, Josh Merel, Martin Riedmiller, Raia Hadsell, and Peter Battaglia. Graph networks as learnable physics engines for inference and control. *arXiv preprint arXiv:1806.01242*, 2018.

Igal Sason and Sergio Verdú. f-divergence inequalities. *IEEE Transactions on Information Theory*, 62(11): 5973–6006, 2016.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International Conference on Machine Learning*, pp. 1889–1897, 2015a.

John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*, 2015b.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Iulian Vlad Serban, Chinnadhurai Sankar, Michael Pieper, Joelle Pineau, and Yoshua Bengio. The bottleneck simulator: A model-based deep reinforcement learning approach. *arXiv preprint arXiv:1807.04723*, 2018.

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *Nature*, 550(7676):354, 2017.

Max Simchowitz, Horia Mania, Stephen Tu, Michael I Jordan, and Benjamin Recht. Learning without mixing: Towards a sharp analysis of linear system identification. *arXiv preprint arXiv:1802.08334*, 2018.

Aravind Srinivas, Allan Jabri, Pieter Abbeel, Sergey Levine, and Chelsea Finn. Universal planning networks. *arXiv preprint arXiv:1804.00645*, 2018.

Wen Sun, Geoffrey J Gordon, Byron Boots, and J Andrew Bagnell. Dual policy iteration. *arXiv preprint arXiv:1805.10755*, 2018.

Richard S Sutton. Integrated architectures for learning, planning, and reacting based on approximating dynamic programming. In *Machine Learning Proceedings 1990*, pp. 216–224. Elsevier, 1990.

Richard S Sutton. Dyna, an integrated architecture for learning, planning, and reacting. *ACM SIGART Bulletin*, 2(4):160–163, 1991.

Richard S Sutton, Csaba Szepesvári, Alborz Geramifard, and Michael P Bowling. Dyna-style planning with linear function approximation and prioritized sweeping. *arXiv preprint arXiv:1206.3285*, 2012.

István Szita and Csaba Szepesvári. Model-based reinforcement learning with nearly tight exploration complexity bounds. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pp. 1031–1038, 2010.

Aviv Tamar, Dotan Di Castro, and Ron Meir. Integrating a partial model into model free reinforcement learning. *Journal of Machine Learning Research*, 13(Jun):1927–1966, 2012.

Voot Tangkaratt, Syogo Mori, Tingting Zhao, Jun Morimoto, and Masashi Sugiyama. Model-based policy gradients with parameter-based exploration by least-squares conditional density estimation. *Neural networks*, 57:128–140, 2014.

Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pp. 5026–5033. IEEE, 2012.

Ronald J Williams and Jing Peng. Function optimization using connectionist reinforcement learning algorithms. *Connection Science*, 3(3):241–268, 1991.

Chris Xie, Sachin Patil, Teodor Moldovan, Sergey Levine, and Pieter Abbeel. Model-based reinforcement learning with parametrized physical models and optimism-driven exploration. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 504–511. IEEE, 2016.

Michael C Yip and David B Camarillo. Model-less feedback control of continuum manipulators in constrained environments. *IEEE Transactions on Robotics*, 30(4):880–889, 2014.

A REFINED BOUNDS

The theoretical limitation of the discrepancy bound $D^G(\widehat{M}, \pi)$ is that the second term involving $\varepsilon_{\max}$ is not rigorously optimizable by stochastic samples. In the worst case, there seem to exist situations where such infinity norm of $G^{\pi,\widehat{M}}$ is inevitable. In this section we tighten the discrepancy bounds with a different closeness measure d, χ^2-divergence, in the policy space, and the dependency on the $\varepsilon_{\max}$ is smaller (though not entirely removed.) We note that χ^2-divergence has the same second order approximation as KL-divergence around the local neighborhood the reference policy and thus locally affects the optimization much.

We start by defining a re-weighted version β^π of the distribution ρ^π where examples in later step are slightly weighted up. We can effectively sample from β^π by importance sampling from ρ^π

Definition A.1. For a policy π, define β^π as the *re-weighted* version of discounted distribution of the states visited by π on $M^\star$. Recall that $p_{S_t^\pi}$ is the distribution of the state at step t, we define $\beta^\pi = (1-\gamma)^2 \sum_{t=1}^\infty t\gamma^{t-1} p_{S_t^\pi}$.

Then we are ready to state our discrepancy bound. Let

$$d^{\chi^2}(\pi, \pi_{\text{ref}}) = \max\{\mathop{\mathbb{E}}_{S\sim\rho^{\pi_{\text{ref}}}}\left[\chi^2(\pi(\cdot|S), \pi_{\text{ref}}(\cdot|S))\right], \mathop{\mathbb{E}}_{S\sim\beta^{\pi_{\text{ref}}}}\left[\chi^2(\pi(\cdot|S), \pi_{\text{ref}}(\cdot|S))\right]\} \tag{A.1}$$

$$D^{\chi^2}_{\pi_{\text{ref}}}(\widehat{M}, \pi) = \kappa\varepsilon_1 + \kappa\delta\varepsilon_2 + \kappa^{3/2}\delta_{3/2}\varepsilon_{\max} \tag{A.2}$$

where $\varepsilon_2 = \mathbb{E}_{S\sim\beta^{\pi_{\text{ref}}}}\left[f^2\right]$ and $\varepsilon_1, \varepsilon_{\max}$ are defined in equation (4.2).

Proposition A.2. *The discrepancy bound D^{χ^2} and closeness measure d^{χ^2} satisfies requirements* (R1) *and* (R2).

We defer the proof to Section C so that we can group relevant proofs with similar tools together. Some of these tools may be of independent interests and used for better analysis of model-free reinforcement learning algorithms such as TRPO Schulman et al. (2015a), PPO Schulman et al. (2017) and CPO Achiam et al. (2017).

B PROOF OF LEMMA 4.3

Proof of Lemma 4.3. Let W_j be the cumulative reward when we use dynamical model M for j steps and then $\widehat{M}$ for the rest of the steps, that is,

$$W_j = \mathop{\mathbb{E}}_{\substack{\forall t\geq 0, A_t\sim\pi(\cdot|S_t)\\ \forall j>t\geq 0, S_{t+1}\sim M(\cdot|S_t,A_t)\\ \forall t\geq j, S_{t+1}\sim\widehat{M}(\cdot|S_t,A_t)}} \left[\sum_{t=0}^{\infty}\gamma^t R(S_t,A_t) \mid S_0 = s\right]$$

By definition, we have that $W_\infty = V^{\pi,M}(s)$ and $W_0 = V^{\pi,\widehat{M}}(s)$. Then, we decompose the target into a telescoping sum,

$$V^{\pi,M}(s) - V^{\pi,\widehat{M}}(s) = \sum_{j=0}^{\infty}(W_{j+1} - W_j) \tag{B.1}$$

Now we re-write each of the summands $W_{j+1} - W_j$. Comparing the trajectory distribution in the definition of W_{j+1} and W_j, we see that they only differ in the dynamical model applied in j-th step. Concretely, W_j and W_{j+1} can be rewritten as $W_j = R + \mathbb{E}_{S_j,A_j\sim\pi,M}\left[\mathbb{E}_{\hat{S}_{j+1}\sim\widehat{M}(\cdot|S_j,A_j)}\left[\gamma^{j+1}V^{\pi,\widehat{M}}(\hat{S}_{j+1})\right]\right]$ and $W_{j+1} = R + \mathbb{E}_{S_j,A_j\sim\pi,M^\star}\left[\mathbb{E}_{S_{j+1}\sim M(\cdot|S_j,A_j)}\left[\gamma^{j+1}V^{\pi,\widehat{M}}(S_{j+1})\right]\right]$ where R denotes the reward from the first j steps from policy π and model $M^\star$. Canceling the shared term in the two equations above, we get

$$W_{j+1} - W_j = \gamma^{j+1}\mathop{\mathbb{E}}_{S_j,A_j\sim\pi,M}\left[\mathop{\mathbb{E}}_{\substack{\hat{S}_{j+1}\sim\widehat{M}(\cdot|S_j,A_j)\\ S_{j+1}\sim M(\cdot|S_j,A_j)}}\left[V^{\pi,\widehat{M}}(S_{j+1}) - V^{\pi,\widehat{M}}(\widehat{S}_{j+1})\right]\right]$$

Combining the equation above with equation (B.1) concludes that

$$V^{\pi,M} - V^{\pi,\widehat{M}} = \frac{\gamma}{1-\gamma}\mathop{\mathbb{E}}_{S\sim\rho^\pi, A\sim\pi(S)}\left[\mathop{\mathbb{E}}_{S'\sim M^\star(\cdot|S,A)}V^{\pi,\widehat{M}}(S') - \mathop{\mathbb{E}}_{\hat{S}'\sim\widehat{M}(\cdot|S,A)}V^{\pi,\widehat{M}}(\hat{S}')\right]$$

□

C MISSING PROOFS IN SECTION 4

Lemma C.1. *Suppose for simplicity the model and the policy are both deterministic. For any one-to-one transformation from $\mathcal{S}$ to $\mathcal{S}'$, let $M^{\mathcal{T}}(s,a) \triangleq \mathcal{T}M(\mathcal{T}^{-1}s,a)$, $R^{\mathcal{T}}(s,a) \triangleq R(\mathcal{T}^{-1}s,a)$, and $\pi^{\mathcal{T}}(s) \triangleq \pi(\mathcal{T}^{-1}s)$ be a set of transformed model, reward and policy. Then we have that (M,π,R) is equivalent to $(M^{\mathcal{T}},\pi^{\mathcal{T}},R^{\mathcal{T}})$ in the sense that*

$$V^{\pi^{\mathcal{T}},M^{\mathcal{T}}}(\mathcal{T}s) = V^{\pi,M}(s)$$

where the value function $V^{\pi^{\mathcal{T}},M^{\mathcal{T}}}$ is defined with respect to $R^{\mathcal{T}}$.

Proof. Let $s_0^{\mathcal{T}} = \mathcal{T}s,\ldots,$ be the sequence of states visited by policy $\pi^{\mathcal{T}}$ on model $M^{\mathcal{T}}$ starting from s. We have that $s_0^{\mathcal{T}} = \mathcal{T}s = \mathcal{T}s_0$. We prove by induction that $s_t^{\mathcal{T}} = \mathcal{T}s_t$. Assume this is true for some value t, then we prove that $s_{t+1}\mathcal{T} = \mathcal{T}s_{t+1}$ holds:

$$\begin{aligned}
s_{t+1}^{\mathcal{T}} &= M^{\mathcal{T}}(s_t^{\mathcal{T}},\pi^{\mathcal{T}}(s_t^{\mathcal{T}})) = M^{\mathcal{T}}(\mathcal{T}s_t,\pi^{\mathcal{T}}(\mathcal{T}s_t)) && \text{(by inductive hypothesis)}\\
&= \mathcal{T}M(s_t,\pi(s_t)) && \text{(by defintion of } M^{\mathcal{T}},\pi^{\mathcal{T}})\\
&= \mathcal{T}s_{t+1}
\end{aligned}$$

Thus we have $R^{\mathcal{T}}(s_t^{\mathcal{T}},a_t^{\mathcal{T}}) = R(s_t,a_t)$. Therefore $V^{\pi^{\mathcal{T}},M^{\mathcal{T}}}(\mathcal{T}s) = V^{\pi,M}(s)$. □

Proof of Proposition 4.4. We first show the invariant of G under deterministic models and policies. The same result applies to stochastic policies with slight modification. Let $s^{\mathcal{T}} = \mathcal{T}s$. We consider the transformation applied to M and $M^\star$ and the resulting G function

$$G^{\mathcal{T}}(s^{\mathcal{T}}, a) \triangleq |V^{\pi^{\mathcal{T}}, M^{\mathcal{T}}}(M^{\mathcal{T}}(s^{\mathcal{T}}, a)) - V^{\pi^{\mathcal{T}}, M^{\mathcal{T}}}(M^{\star,\mathcal{T}}(s^{\mathcal{T}}, a))|$$

Note that by Lemma C.1, we have that $V^{\pi^{\mathcal{T}}, M^{\mathcal{T}}}(M^{\mathcal{T}}(s^{\mathcal{T}}, a)) = V^{\pi, M}(\mathcal{T}^{-1}M^{\mathcal{T}}(s^{\mathcal{T}}, a)) = V^{\pi, M}(M(s, a))$. Similarly, $V^{\pi^{\mathcal{T}}, M^{\mathcal{T}}}(M^{\star,\mathcal{T}}(s^{\mathcal{T}}, a)) = V^{\pi, M}(M^\star(s, a))$. Therefore we obtain that

$$G^{\mathcal{T}}(s^{\mathcal{T}}, a) = G(s, a) \tag{C.1}$$

By Lemma 4.3 and triangle inequality, we have that

$$\begin{aligned}
\frac{1-\gamma}{\gamma}\left|V^{\pi,M} - V^{\pi,\widehat{M}}\right| &\leq \mathop{\mathbb{E}}_{S\sim\rho^\pi}\left[\left|G^{\pi,\widehat{M}}(S)\right|\right] && \text{(triangle inequality)}\\
&\leq \mathop{\mathbb{E}}_{S\sim\rho^{\pi_{\text{ref}}}}\left[\left|G^{\pi,\widehat{M}}(S)\right|\right] + |\rho^\pi - \rho^{\pi_{\text{ref}}}|_1 \cdot \max_S \left|G^{\pi,\widehat{M}}(S)\right| \\
& && \text{(Holder inequality)}
\end{aligned}$$

By Corollary E.7 we have that $|\rho^\pi - \rho^{\pi_{\text{ref}}}|_1 \leq \frac{\gamma}{1-\gamma}\mathbb{E}_{S\sim\rho^{\pi_{\text{ref}}}}\left[KL(\pi(S), \pi_{\text{ref}}(S))^{1/2}|S\right] = \frac{\delta\gamma}{1-\gamma}$. Combining this with the equation above, we complete the proof. □

Proof of Proposition A.2. Let μ be the distribution of the initial state S_0, and let P' and P be the state-to-state transition kernel under policy π and π_{ref}. Let $\bar{G} = (1-\gamma)\sum_{k=0}^\infty \gamma^k P^k$ and $\bar{G}' = (1-\gamma)\sum_{k=0}^\infty \gamma^k P'^k$. Under these notations, we can re-write $\rho^{\pi_{\text{ref}}} = \bar{G}\mu$ and $\rho^\pi = \bar{G}'\mu$. Moreover, we observe that $\beta^{\pi_{\text{ref}}} = \bar{G}P\bar{G}\mu$.

Let $\delta_1 = (1-\gamma)\chi^2_{\bar{G}\mu}(P', P)^{1/2}$ and $\delta_2 = (1-\gamma)\chi^2_{\bar{G}P\bar{G}\mu}(P', P)^{1/2}$ by the χ^2 divergence between P' and P, measured with respect to distributions $\bar{G}\mu = \rho^{\pi_{\text{ref}}}$ and $\bar{G}P\bar{G}\mu = \beta^{\pi_{\text{ref}}}$. By Lemma D.1, we have that the χ^2-divergence between the states can be bounded by the χ^2-divergence between the actions in the sense that:

$$\begin{aligned}
\chi^2_{\bar{G}\mu}(P', P)^{1/2} = \chi^2_{\rho^{\pi_{\text{ref}}}}(P', P)^{1/2} &\leq \mathop{\mathbb{E}}_{S\sim\rho^{\pi_{\text{ref}}}}\left[\chi^2(\pi(\cdot|S), \pi_{\text{ref}}(\cdot|S))\right]\\
\chi^2_{\bar{G}P\bar{G}\mu}(P', P)^{1/2} = \chi^2_{\beta^{\pi_{\text{ref}}}}(P', P)^{1/2} &\leq \mathop{\mathbb{E}}_{S\sim\beta^{\pi_{\text{ref}}}}\left[\chi^2(\pi(\cdot|S), \pi_{\text{ref}}(\cdot|S))\right]
\end{aligned}$$

Therefore we obtain that $\delta_1 \leq (1-\gamma)\delta$, $\delta_2 \leq (1-\gamma)\delta$. Let $f(s) = G^{\pi,\widehat{M}}(s)$. We can control the difference between $\langle\rho^{\pi_{\text{ref}}}, f\rangle$ and $\langle\rho^\pi, f\rangle$ by

$$\begin{aligned}
\left|\mathop{\mathbb{E}}_{S\sim\rho^{\pi_{\text{ref}}}}\left[G^{\pi,\widehat{M}}(S) - \mathop{\mathbb{E}}_{S\sim\rho^\pi}\left[G^{\pi,\widehat{M}}(S)\right]\right]\right| &= |\langle\rho^{\pi_{\text{ref}}}, f\rangle - \langle\rho^\pi, f\rangle|\\
&\leq (1-\gamma)^{-1}\left(\delta_1\langle\bar{G}P\bar{G}\mu, f^2\rangle^{1/2} + \delta_1\delta_2^{1/2}\|f\|_\infty\right)\\
&= (1-\gamma)^{-1}\left(\delta_1\varepsilon_2 + \delta_1\delta_2^{1/2}\varepsilon_{\max}\right)\\
&\leq \delta\varepsilon_2 + (1-\gamma)^{-1/2}\delta_1\delta_2^{1/2}\varepsilon_{\max}
\end{aligned}$$

It follows that

$$\begin{aligned}
\left|V^{\pi,\widehat{M}} - V^{\pi,M}\right| &\leq \gamma(1-\gamma)^{-1}\left|\mathop{\mathbb{E}}_{S\sim\rho^\pi}\left[G^{\pi,\widehat{M}}(S)\right]\right| && \text{(by Lemma 4.3)}\\
&\leq \gamma(1-\gamma)^{-1}\left(\left|\mathop{\mathbb{E}}_{S\sim\rho^{\pi_{\text{ref}}}}\left[G^{\pi,\widehat{M}}(S)\right]\right| + \left|\mathop{\mathbb{E}}_{S\sim\rho^{\pi_{\text{ref}}}}\left[G^{\pi,\widehat{M}}(S) - \mathop{\mathbb{E}}_{S\sim\rho^\pi}\left[G^{\pi,\widehat{M}}(S)\right]\right]\right|\right)\\
&\leq \gamma(1-\gamma)^{-1}\left|\mathop{\mathbb{E}}_{S\sim\rho^{\pi_{\text{ref}}}}\left[G^{\pi,\widehat{M}}(S)\right]\right| + \gamma(1-\gamma)^{-1}\delta\varepsilon_2 + \gamma(1-\gamma)^{-3/2}\delta_1\delta_2^{1/2}\varepsilon_{\max}
\end{aligned}$$

□

Proof of Proposition 4.1 and 4.2 . By definition of G and the Lipschitzness of $V^{\pi,\widehat{M}}$, we have that $|G^{\pi,\widehat{M}}(s,a)| \le L|\widehat{M}(s,a) - M^\star(s,a)|$. Then, by Lemma 4.3 and triangle inequality, we have that

$$\begin{aligned}\left|V^{\pi,\widehat{M}} - V^{\pi,M^\star}\right| &= \kappa \cdot \Big| \mathop{\mathbb{E}}_{\substack{S\sim\rho^{\pi,M}\\ A\sim\pi(\cdot|S)}} \left[G^{\pi,\widehat{M}}(S,A)\right] \Big| \le \kappa \mathop{\mathbb{E}}_{\substack{S\sim\rho^{\pi,M}\\ A\sim\pi(\cdot|S)}} \left[\left|G^{\pi,\widehat{M}}(S,A)\right|\right] \\ &\le \kappa \mathop{\mathbb{E}}_{\substack{S\sim\rho^{\pi}\\ A\sim\pi(\cdot|S)}} \left[\|\widehat{M}(S,A) - M^\star(S,A)\|\right] . \end{aligned} \tag{C.2}$$

Next we prove the main part of the proposition. Thus we proved Proposition 4.1. Note that for any distribution ρ and ρ' and function f, we have $\mathbb{E}_{S\sim\rho} f(S) = \mathbb{E}_{S\sim\rho'} f(S) + \langle \rho - \rho', f\rangle \le \mathbb{E}_{S\sim\rho'} f(S) + \|\rho-\rho'\|_1 \|f\|_\infty$. Thus applying this inequality with $f(S) = \mathbb{E}_{A\sim\pi(\cdot|S)}\left[\|\widehat{M}(S,A) - M^\star(S,A)\|\right]$, we obtain that

$$\begin{aligned}\mathop{\mathbb{E}}_{\substack{S\sim\rho^{\pi}\\ A\sim\pi(\cdot|S)}} \left[\|\widehat{M}(S,A) - M^\star(S,A)\|\right] &\le \mathop{\mathbb{E}}_{\substack{S\sim\rho^{\pi_{\text{ref}}}\\ A\sim\pi(\cdot|S)}} \left[\|\widehat{M}(S,A) - M^\star(S,A)\|\right] \\ &\quad + \|\rho^{\pi_{\text{ref}}} - \rho\|_1 \max_S \mathop{\mathbb{E}}_{A\sim\pi(\cdot|S)} \left[\|\widehat{M}(S,A) - M^\star(S,A)\|\right] \\ &\le \mathop{\mathbb{E}}_{\substack{S\sim\rho^{\pi_{\text{ref}}}\\ A\sim\pi(\cdot|S)}} \left[\|\widehat{M}(S,A) - M^\star(S,A)\|\right] + 2\delta\kappa B \end{aligned} \tag{C.3}$$

where the last inequality uses the inequalities (see Corollary E.7) that $\|\rho^\pi - \rho^{\pi_{\text{ref}}}\|_1 \le \frac{\gamma}{1-\gamma}\mathbb{E}_{S\sim\rho^{\pi_{\text{ref}}}}\left[KL(\pi(S),\pi_{\text{ref}}(S))^{1/2}|S\right] = \delta\kappa$ and that $\|\widehat{M}(S,A) - M^\star(S,A)\| \le 2B$. Combining (C.3) and (C.2) we complete the proof of Proposition 4.2. □

D χ^2-DIVERGENCE BASED INEQUALITIES

Lemma D.1. *Let S be a random variable over the domain $\mathcal{S}$. Let π and π' be two policies and and $A \sim \pi(\cdot \mid S)$ and $A' \sim \pi'(\cdot \mid S)$. Let $Y \sim M(\cdot \mid S, A)$ and $Y' \sim M(\cdot \mid S, A')$ be the random variables for the next states under two policies. Then,*

$$\mathbb{E}\left[\chi^2(Y|S, Y'|S)\right] \le \mathbb{E}\left[\chi^2(A|S, A'|S)\right]$$

Proof. By definition, we have that $Y|S = s, A = a$ has the same density as $Y'|S = s, A' = a$ for any a and s. Therefore by Theorem E.4 (setting X, X', Y, Y' in Theorem E.4 by $A|S = s$, $A'|S = s$, $Y|S = s$, $Y'|S = s$ respectively), we have

$$\chi^2(Y|S = s, Y'|S = s) \le \chi^2(A|S = s, A'|S = s)$$

Taking expectation over the randomness of S we complete the proof.

□

D.1 PROPERTIES OF MARKOV PROCESSES

In this subsection, we consider bounded the difference of the distributions induced by two markov process starting from the same initial distributions μ. Let P, P' be two transition kernels. Let $G = \sum_{k=0}^\infty \gamma^k P^k$ and $\bar{G} = (1-\gamma)G$. Define G' and $\bar{G}'$ similarly. Therefore we have that $\bar{G}\mu$ is the discounted distribution of states visited by the markov process starting from distribution μ. In other words, if μ is the distribution of S_0, and P is the transition kernel induced by some policy π, then $\bar{G}\mu = \rho^\pi$.

First of all, let $\Delta = \gamma(P' - P)$ and we note that with simple algebraic manipulation,

$$\bar{G}' - \bar{G} = (1-\gamma)^{-1}\bar{G}'\Delta\bar{G} \tag{D.1}$$

Let f be some function. We will mostly interested in the difference between $\mathbb{E}_{S\sim\bar{G}\mu}[f]$ and $\mathbb{E}_{S\sim\bar{G}'\mu}[f]$, which can be rewritten as $\langle(\bar{G}' - G)\mu, f\rangle$. We will bound this quantity from above by some divergence measure between P' and P.

We start off with a simple lemma that controls the form $\langle p-q, f\rangle$ by the χ^2 divergence between p and q. With this lemma we can reduce our problem of bounding $\langle(\bar{G}'-G)\mu, f\rangle$ to characterizing the χ^2 divergence between $\bar{G}'\mu$ and $\bar{G}\mu$.

Lemma D.2. *Let p and q be probability distributions. Then we have*

$$\langle q-p, f\rangle^2 \leq \chi^2(q,p)\cdot\langle p, f^2\rangle$$

Proof. By Cauchy-Schwartz inequality, we have

$$\langle q-p, f\rangle^2 \leq \left(\int \frac{(q(x)-p(x))^2}{p(x)}dx\right)\left(\int p(x)f(x)^2\right) = \chi^2(q,p)\cdot\langle p, f^2\rangle$$

□

The following Lemma is a refinement of the lemma above. It deals with the distributions p and q with the special structure $p = WP'\mu$ and $q = WP\mu$.

Lemma D.3. *Let W, P', P be transition kernels and μ be a distribution. Then,*

$$\langle W(P'-P)\mu, f\rangle^2 \leq \chi^2_\mu(P',P)\langle WP\mu, f^2\rangle$$

where $\chi^2_\mu(P',P)$ is a divergence between transitions defined in Definition E.3.

Proof. By Lemma D.2 with $p = WP\mu$ and $q = WP'\mu$, we conclude that

$$\langle W(P'-P)\mu, f\rangle^2 \leq \chi^2(q,p)\cdot\langle p, f^2\rangle \leq \chi^2(WP'\mu, WP\mu)\langle WP\mu, f^2\rangle$$

By Theorem E.4 and Theorem E.5 we have that $\chi^2(WP'\mu, WP\mu) \leq \chi^2(P'\mu, P\mu) \leq \chi^2_\mu(P',P)$, plugging this into the equation above we complete the proof. □

Now we are ready to state the main result of this subsection.

Lemma D.4. *Let $\bar{G}, \bar{G}', P', P, f$ as defined in the beginning of this section. Let $\delta_1 = (1-\gamma)\chi^2_{\bar{G}\mu}(P',P)^{1/2}$ and $\delta_2 = (1-\gamma)\chi^2_{\bar{G}P\bar{G}\mu}(P',P)^{1/2}$. Then,*

$$(1-\gamma)\left|\langle\bar{G}'\mu, f\rangle - \langle\bar{G}\mu, f\rangle\right| \leq \delta_1\|f\|_\infty$$
$$(1-\gamma)\left|\langle\bar{G}'\mu, f\rangle - \langle\bar{G}\mu, f\rangle\right| \leq \delta_1\langle\bar{G}P\bar{G}\mu, f^2\rangle^{1/2} + \delta_1\delta_2^{1/2}\|f\|_\infty$$

Proof. Recall by equation (D.1), we have $(\bar{G}'-\bar{G})\mu = \langle\bar{G}'\Delta\bar{G}\mu, f\rangle$. By Lemma D.3,

$$\langle\bar{G}'\Delta\bar{G}\mu, f\rangle^2 \leq \chi^2_{\bar{G}\mu}(P',P)\langle\bar{G}'P\bar{G}\mu, f^2\rangle \tag{D.2}$$

Using equation (D.1) again, we have

$$\langle\bar{G}'P\bar{G}\mu, f^2\rangle = \langle\bar{G}P\bar{G}\mu, f^2\rangle + \frac{1}{1-\gamma}\langle\bar{G}'\Delta\bar{G}P\bar{G}\mu, f^2\rangle \tag{D.3}$$

By Lemma D.3 again, we have that

$$\langle\bar{G}'\Delta\bar{G}P\bar{G}, f^2\rangle^2 \leq \chi^2_{\bar{G}P\bar{G}\mu}(P',P)\langle\bar{G}'P\bar{G}P\bar{G}\mu, f^4\rangle \tag{D.4}$$

By Holder inequality and the fact that $\|\bar{G}\|_{1\to1} = 1$, $\|\bar{G}'\|_{1\to1} = 1$ and $\|P\|_{1\to1} = 1$, we have

$$\langle\bar{G}'P\bar{G}P\bar{G}\mu, f^4\rangle \leq \|\bar{G}'P\bar{G}P\bar{G}\mu\|_1\|f^4\|_\infty \leq \|f\|_\infty^4 \tag{D.5}$$

Then, combining equation (D.2), (D.3), (D.5), we have

$$\begin{aligned}
(1-\gamma)|\langle\bar{G}'\Delta\bar{G}\mu, f\rangle| &\leq \delta_1\langle\bar{G}'P\bar{G}\mu, f^2\rangle^{1/2} && \text{(by equation (D.2))}\\
&\leq \delta_1\langle\bar{G}P\bar{G}\mu, f^2\rangle^{1/2} + \delta_1\langle\bar{G}'\Delta\bar{G}P\bar{G}\mu, f^2\rangle^{1/2} && \text{(by equation (D.3) and AM-GM)}\\
&\leq \delta_1\langle\bar{G}P\bar{G}\mu, f^2\rangle^{1/2} + \delta_1\delta_2^{1/2}\|f\|_\infty && \text{(by equation (D.4) and (D.5))}
\end{aligned}$$

□

The following Lemma is a stronger extension of Lemma D.4, which can be used to future improve Proposition A.2, and may be of other potential independent interests. We state it for completeness.

Lemma D.5. *Let $\bar{G}, \bar{G}', P', P, f$ as defined in the beginning of this section. Let $d_k = (\bar{G}P)^k \bar{G}\mu$ and $\delta_k = (1-\gamma)\chi^2_{d_{k-1}}(P', P)^{1/2}$, then we have that for any K,*

$$(1-\gamma)\left|\langle \bar{G}'\mu, f\rangle - \langle \bar{G}\mu, f\rangle\right| \leq \delta_1 \langle d_1, f^2\rangle^{-1/2} + \delta_1\delta_2^{-1/2}\langle d_2, f^4\rangle^{-1/4} + \delta_1 \ldots \delta_{K-1}^{2^{-K}}\langle d_k, f^{2^k}\rangle^{2^{-k}} + \delta_1 \ldots \delta_{K-1}^{2^{-K}}\|f\|_\infty$$

Proof. We first use induction to prove that:

$$(1-\gamma)|\langle (\bar{G}' - \bar{G})\mu, f\rangle| \leq \sum_{k=1}^{K}\left(\prod_{0\leq s\leq k-1}\delta_{s+1}^{2^{-s}}\right)\langle d_k, f^{2^k}\rangle^{2^{-k}} + \left(\prod_{0\leq s\leq K-1}\delta_{s+1}^{2^{-s}}\right)\langle \bar{G}'\Delta(\bar{G}P)^K\bar{G}\mu, f^{2^K}\rangle^{2^{-K}} \quad \text{(D.6)}$$

By Lemma D.4, we got the case for $K = 1$. Assuming we have prove the case for K, then applying

$$\begin{aligned}\langle \bar{G}'\Delta(\bar{G}P)^K\bar{G}\mu, f^{2^K}\rangle &= \langle \bar{G}\Delta(\bar{G}P)^K\bar{G}\mu, f^{2^K}\rangle + (1-\gamma)^{-1}\langle \bar{G}\Delta(\bar{G}P)^{K+1}\bar{G}\mu, f^{2^K}\rangle \quad \text{(D.7)}\\ &\leq \langle \bar{G}\Delta(\bar{G}P)^K\bar{G}\mu, f^{2^K}\rangle \\ &\quad + (1-\gamma)^{-1}\chi^2_{d_K}(P', P)^{1/2}\langle \bar{G}'\Delta(\bar{G}P)^{K+1}\bar{G}\mu, f^{2^{K+1}}\rangle^{1/2} \\ &\leq \langle \bar{G}\Delta(\bar{G}P)^K\bar{G}\mu, f^{2^K}\rangle + \delta_{K+1}\langle \bar{G}'\Delta(\bar{G}P)^{K+1}\bar{G}\mu, f^{2^{K+1}}\rangle^{1/2}\end{aligned}$$

By Cauchy-Schwartz inequality, we obtain that

$$\langle \bar{G}'\Delta(\bar{G}P)^K\bar{G}\mu, f^{2^K}\rangle^{2^{-K}} \leq \langle \bar{G}\Delta(\bar{G}P)^K\bar{G}\mu, f^{2^K}\rangle^{2^{-K}} + \delta_{K+1}\langle \bar{G}'\Delta(\bar{G}P)^{K+1}\bar{G}\mu, f^{2^{K+1}}\rangle^{2^{-K-1}}$$

Plugging the equation above into equation (D.6), we provide the induction hypothesis for the case with $K + 1$.

Now applying $\langle \bar{G}'\Delta(\bar{G}P)^K\bar{G}\mu, f^{2^K}\rangle^{2^{-K}} \leq \|f\|_\infty$ with equation (D.6) we complete the proof.

□

E TOOLBOX

Definition E.1 (χ^2 distance, c.f. Nielsen & Nock (2014); Cover & Thomas (2012))**.** The Neyman χ^2 distance between two distributions p and q is defined as

$$\chi^2(p, q) \triangleq \int \frac{(p(x) - q(x))^2}{q(x)}dx = \int \frac{p(x)^2}{q(x)}dx - 1$$

For notational simplicity, suppose two random variables X and Y has distributions p_X and p_Y, we often write $\chi^2(X, Y)$ as a simplification for $\chi^2(p_X, p_Y)$.

Theorem E.2 (Sason & Verdú (2016))**.** *The Kullback-Leibler (KL) divergence between two distributions p, q is bounded from above by the χ^2 distance:*

$$KL(p, q) \leq \chi^2(p, q)$$

Proof. Since $\log$ is a concave function, by Jensen inequality we have

$$\begin{aligned}KL(p, q) &= \int p(x)\log\frac{p(x)}{q(x)}dx \leq \log\int p(x)\cdot\frac{p(x)}{q(x)}dx \\ &= \log(\chi^2(p, q) + 1) \leq \chi^2(p, q)\end{aligned}$$

□

Definition E.3 (χ^2 distance between transitions)**.** Given two transition kernels P, P'. For any distribution μ, we define $\chi^2_\mu(P', P)$ as:

$$\chi^2_\mu(P', P) \triangleq \int \mu(x)\chi^2(P'(\cdot|X = x), P(\cdot|X = x))dx$$

Theorem E.4. *Suppose random variables (X, Y) and (X', Y') satisfy that $p_{Y|X} = p_{Y'|X'}$. Then*

$$\chi^2(Y, Y') \leq \chi^2(X, X')$$

Or equivalently, for any transition kernel P and distribution μ, μ', we have

$$\chi^2(P\mu, P\mu') \leq \chi^2(\mu, \mu')$$

Proof. Denote $p_{Y|X}(y \mid x) = p_{Y'|X'}(y \mid x)$ by $p(y \mid x)$, and we rewrite p_X as p and $p_{X'}$ as p'. By Cauchy-Schwarz inequality, we have:

$$\begin{aligned} p_Y(y)^2 = \left(\int p(y|x)p(x)dx\right)^2 &\leq \left(\int p(y|x)p'(x)dx\right)\left(\int p(y|x)\frac{p(x)^2}{p'(x)}dx\right) \\ &= p_{Y'}(y)\left(\int p(y|x)\frac{p(x)^2}{p'(x)}dx\right) \end{aligned} \tag{E.1}$$

It follows that

$$\chi^2(Y, Y') = \int \frac{p_Y(y)^2}{p_{Y'}(y)}dy - 1 \leq \int dy \int p(y|x)\frac{p(x)^2}{p'(x)}dx - 1 = \chi^2(X, X')$$

□

Theorem E.5. *Let X, Y, Y' are three random variables. Then,*

$$\chi^2(Y, Y') \leq \mathbb{E}\left[\chi^2(Y|X, Y'|X)\right]$$

We note that the expectation on the right hand side is over the randomness of X.[10] *As a direct corollary, we have for transition kernel P' and P and distribution μ,*

$$\chi^2(P'\mu, P\mu) \leq \chi^2_\mu(P', P)$$

Proof. We denote $p_{Y'|X}(y|x)$ by $p'(y \mid x)$ and $p_{Y|X}(y|x)$ by $p(y|x)$, and let $p(x)$ be a simplification for $p_X(x)$. We have by Cauchy-Schwarz,

$$\frac{p_Y(y)^2}{p_{Y'}(y)} = \frac{\left(\int p(y|x)p(x)dx\right)^2}{\int p'(y \mid x)p(x)dx} \leq \int \frac{p(y|x)^2}{p'(y|x)}p(x)dx$$

It follows that

$$\chi^2(Y, Y') = \int \frac{p_Y(y)^2}{p_{Y'}(y)}dy - 1 \leq \int \frac{p(y|x)^2}{p'(y|x)}p(x)dxdy - 1 = \mathbb{E}\left[\chi^2(Y|X, Y'|X) \mid X\right]$$

□

Claim E.6. *Let μ be a distribution over the state space $\mathcal{S}$. Let P and P' be two transition kernels. $G = \sum_{k=0}^{\infty}(\gamma P)^k = (\mathrm{Id} - \gamma P)^{-1}$ and $G' = \sum_{k=0}^{\infty}(\gamma P')^k = (\mathrm{Id} - \gamma P')^{-1}$. Let $d = (1-\gamma)G\mu$ and $d' = (1-\gamma)G'\mu$ be the discounted distribution starting from μ induced by the transition kernels G and G'. Then,*

$$|d - d'|_1 \leq \frac{1}{1-\gamma}|\Delta d|_1$$

Moreover, let $\gamma(P' - P) = \Delta$. Then, we have

$$G' - G = \sum_{k=1}^{\infty}(G\Delta)^k G$$

[10] Observe $\chi^2(Y|X, Y'|X)$ deterministically depends on X.

Proof. With algebraic manipulation, we obtain,

$$\begin{aligned} G' - G &= (\mathrm{Id} - \gamma P')^{-1}((\mathrm{Id} - \gamma P) - (\mathrm{Id} - \gamma P')(\mathrm{Id} - \gamma P)^{-1} \\ &= G' \Delta G \end{aligned} \tag{E.2}$$

It follows that

$$\begin{aligned} |d - d'|_1 &= (1-\gamma)|G'\Delta G\mu|_1 \leq |\Delta G \mu|_1 \qquad \text{(since } (1-\gamma)|G'|_{1\to 1} \leq 1) \\ &= \frac{1}{1-\gamma}|\Delta d|_1 \end{aligned}$$

Replacing G' in the RHS of the equation (E.2) by $G' = G + G'\Delta G$, and doing this recursively gives

$$G' - G = \sum_{k=1}^{\infty} (G\Delta)^k G$$

□

Corollary E.7. *Let π and π' be two policies and let ρ^π be defined as in Definition 2.1. Then,*

$$|\rho^\pi - \rho'|_1 \leq \frac{\gamma}{1-\gamma} \mathop{\mathbb{E}}_{S\sim\rho^\pi} \left[KL(\pi(S), \pi'(S))^{1/2} \mid S \right]$$

Proof. Let P and P' be the state-state transition matrix under policy π and π' and $\Delta = \gamma(P' - P)$ By Claim E.6, we have that

$$\begin{aligned} |\rho^\pi - \rho^{\pi'}|_1 \leq \frac{1}{1-\gamma}|\Delta\rho^\pi|_1 &= \frac{\gamma}{1-\gamma} \mathop{\mathbb{E}}_{S\sim\rho^\pi} \left[|p_{M^\star(S,\pi(S))|S} - p_{M^\star(S,\pi'(S))|S}|_1 \right] \\ &\leq \frac{\gamma}{1-\gamma} \mathop{\mathbb{E}}_{S\sim\rho^\pi} \left[|p_{\pi(S)|S} - p_{\pi'(S)|S}|_1 \right] \\ &\leq \frac{\gamma}{1-\gamma} \mathop{\mathbb{E}}_{S\sim\rho^\pi} \left[KL(\pi(S), \pi'(S))^{1/2} \mid S \right] \quad \text{(by Pinkser's inequality)} \end{aligned}$$

□

F Implementation Details

F.1 Environment Setup

We benchmark our algorithm on six tasks based on physics simulator Mujoco (Todorov et al., 2012). We use rllab's implementation (Duan et al., 2016) [11] to interact with Mujoco. All the environments we use have a maximum horizon of 500 steps. We remove all contact information from observation. To compute reward from states, we put the velocity of center of mass into the states.

F.2 Network Architecture and Model Learning

We use the same reward function as in rllab, except that all the coefficients C_{contact} in front of the contact force s are set to 0 in our case. We refer the readers to (Duan et al., 2016) Supp Material 1.2 for more details. All actions are projected to the action space by clipping. We normalize all observations by $s' = \frac{s-\mu}{\sigma}$ where $\mu, \sigma \in \mathbb{R}^{d_{\text{observation}}}$ are computed from all observations we collect from the real environment. Note that μ, σ may change as we collect new data. Our policy will always produce an action a in $[-1, 1]^{d_{\text{action}}}$ and the action a', which is fed into the environment, is scaled linearly by $a' = \frac{1-a}{2}a_{\text{min}} + \frac{1+a}{2}a_{\text{max}}$, where $a_{\text{min}}, a_{\text{max}}$ are the min or max values allowed at each entry.

[11] commit b3a2899 in `https://github.com/rll/rllab/`

Table 1: TRPO Hyperparameters.

Hyperparameters	**Values**
batch size	4000
max KL divergence	0.01
discount γ	0.99
GAE λ	0.95
CG iterations	10
CG damping	0.1

F.3 SLBO DETAILS

The dynamical model is represented by a feed-forward neural network with two hidden layers, each of which contains 500 hidden units. The activation function at each layer is ReLU. We use Adam to optimize the loss function with learning rate 10^{-3} and L_2 regularization 10^{-5}. The network does not predict the next state directly; instead, it predicts the normalized difference of $s_{t+1} - s_t$. The normalization scheme and statistics are the same as those of observations: We maintain μ, σ from collected data in the real environment and may change them as we collect more, and the normalized difference is $\frac{s_{t+1}-s_t-\sigma}{\mu}$.

The policy network is a feed-forward network with two hidden layers, each of which contains 32 hidden units. The policy network uses $\tanh$ as activation function and outputs a Gaussian distribution $\mathcal{N}(\mu(s), \sigma^2)$ where σ a state-independent trainable vector.

During our evaluation, we use $H = 2$ for multi-step model training and the batch size is given by $\frac{256}{H} = 128$, i.e., we enforce the model to see 256 transitions at each batch.

We run our algorithm $n_{\text{outer}} = 100$ iterations. We collect $n_{\text{train}} = 10000$ steps of real samples from the environment at the start of each iteration using current policy with Ornstein-Uhlunbeck noise (with parameter $\theta = 0.15, \sigma = 0.3$) for better exploration. At each iteration, we optimize dynamics model and policy alternatively for $n_{\text{inner}} = 20$ times. At each iteration, we optimize dynamics model for $n_{\text{model}} = 100$ times and optimize policy for $n_{\text{policy}} = 40$ times.

F.4 BASELINES

TRPO. TRPO hyperparameters are listed at Table 1, which are the same as OpenAI Baselines' implementation. These hyperparameters are fixed for all experiments where TRPO is used, including ours, MB-TRPO and MF-TRPO. We do not tune these hyperparameters. We also normalize observations as our algorithm and OpenAI Baselines do.

We use a neural network as the value function to reduce variance, which has 2 hidden layers of units 64 and uses $\tanh$ as activation functions. We use Generalized Advantage Estimator (GAE) Schulman et al. (2015b) to estimate advantages. Both TRPO used in our algorithm and that in model-free algorithm share the same set of hyperparameters.

SAC. For fair comparison, we do not use a large policy network (2 hidden layers, one of which has 256 hidden units) as the authors suggest, but use exactly the same policy network as ours. All other hyperparameters are kept the same as the authors'. Note that Q network and value network have 256 hidden units at each hidden layers, which is more than TRPO's. We refer the readers to Haarnoja et al. (2018) Appendix D for more details.

MB-TRPO. Model-Based TRPO (MB-TRPO) is similar to our algorithm SLBO but does not optimize model and policy alternatively during one iteration. We do not tune the hyperparameter n_{model} since any number beyond a certain threshold would bring similar results. For n_{policy} we try $\{100, 200, 400, 800\}$ on Ant and find $n_{\text{policy}} = 200$ works best in Ant so we use it for all other environments. Note that when Algo 2 uses 800 Adam updates (per outer iteration), it has the same amount of updates (per outer iteration) as in Algo 3. As suggested by Section 6.1, we use 0.005 as the coefficient of entropy bonus for all experiments.

Algorithm 3 Model-Based Trust Region Policy Optimization (MB-TRPO)

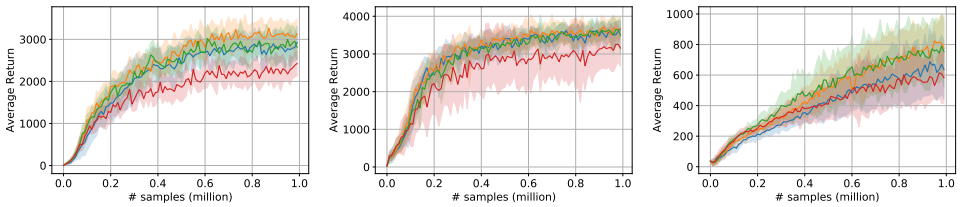
initialize model network parameters ϕ and policy network parameters θ
initialize dataset $\mathcal{D} \leftarrow \emptyset$
for n_{outer} iterations **do**
  $\mathcal{D} \leftarrow \mathcal{D} \cup \{$ collect n_{collect} samples from real environment using π_θ with noises $\}$
  for n_{model} iterations **do**
    optimize (6.1) over ϕ with sampled data from $\mathcal{D}$ by one step of Adam
  for n_{policy} iterations **do**
    $\mathcal{D}' \leftarrow \{$ collect n_{trpo} samples using $\widehat{M}_\phi$ as dynamics $\}$
    optimize π_θ by running TRPO on $\mathcal{D}'$

SLBO. We tune multi-step model training parameter $H \in \{1, 2, 4, 8\}$, entropy regularization coefficient $\lambda \in \{0, 0.001, 0.003, 0.005\}$ and $n_{\text{policy}} \in \{10, 20, 40\}$ on Ant and find $H = 2, \lambda = 0.005, n_{\text{policy}} = 40$ work best, then we fix them in all environments, though environment-specific hyperparameters may work better. The other hyperparameters, including $n_{\text{inner}}, n_{\text{model}}$ and network architecture, are never tuned. We observe that at the first several iterations, the policy overfits to the learnt model so a reduction of n_{policy} at the beginning can further speed up convergence but we omit this for simplicity.

The most important hyperparameters we found are n_{policy} and the coefficient in front of the entropy regularizer λ. It seems that once n_{model} is large enough we don't see any significant changes. We did have a held-out set for model prediction (with the same distribution as the training set) and found out the model doesn't overfit much. As mentioned in F.3, we also found out normalizing the state helped a lot since the raw entries in the state have different magnitudes; if we don't normalize them, the loss will be dominated by the loss of some large entries.

F.5 Ablation Study

Multi-step model training. We compare multi-step model training with single-step model training and the results are shown on Figure 2. Note that $H = 1$ means we use single-step model training. We observe that small H (e.g., 2 or 4) can be beneficial, but larger H (e.g., 8) can hurt. We hypothesize that smaller H can help the model learn the uncertainty in the input and address the error-propagation issue to some extent. Pathak et al. (2018) uses an auto-regressive recurrent model to predict a multi-step loss on a trajectory, which is closely related to ours. However, theirs differs from ours in the sense that they do not use the predicted output x_{t+1} as the input for the prediction of x_{t+2}, and so on and so forth.



(a) Ant (b) Walker (c) Humanoid

Figure 2: Ablation study on multi-step model training. All the experiments are average over 10 random seeds. The x-axis shows the total amount of real samples from the environment. The y-axis shows the averaged return from execution of our learned policy. The solid line is the mean of the total rewards from each seed. The shaded area is one-standard deviation.

Entropy regularization. Figure 3 shows that entropy reguarization can improve both sample efficiency and final performance. More entropy regularization leads to better sample efficiency and higher total rewards. We observe that in the late iterations of training, entropy regularization may hurt the performance thus we stop using entropy regularization in the second half of training.

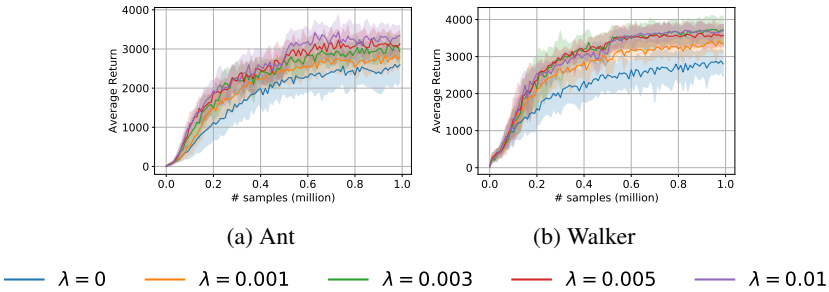


Figure 3: Ablation study on entropy regularization. λ is the coefficient of entropy regularization in the TRPO's objective. All the experiments are averaged over 10 random seeds. The x-axis shows the total amount of real samples from the environment. The y-axis shows the averaged return from execution of our learned policy. The solid line is the mean of the total rewards from each seed. The shaded area is one-standard deviation.

SLBO with 4M training steps. Figure 4 shows that SLBO is superior to SAC and MF-TRPO in Swimmer, Half Cheetah, Walker and Humanoid when 4 million samples or fewer samples are allowed. For Ant environment , although SLBO with less than one million samples reaches the performance of MF-TRPO with 8 million samples, SAC's performance surpasses SLBO after 2 million steps of training. Since model-free TRPO almost stops improving after 8M steps and our algorithms uses TRPO for optimizing the estimated environment, we don't expect SLBO can significantly outperform the reward of TRPO at 8M steps. The result shows that SLBO is also satisfactory in terms of asymptotic convergence (compared to TRPO.) It also indicates a better planner or policy optimizer instead of TRPO might be necessary to further improve the performance.

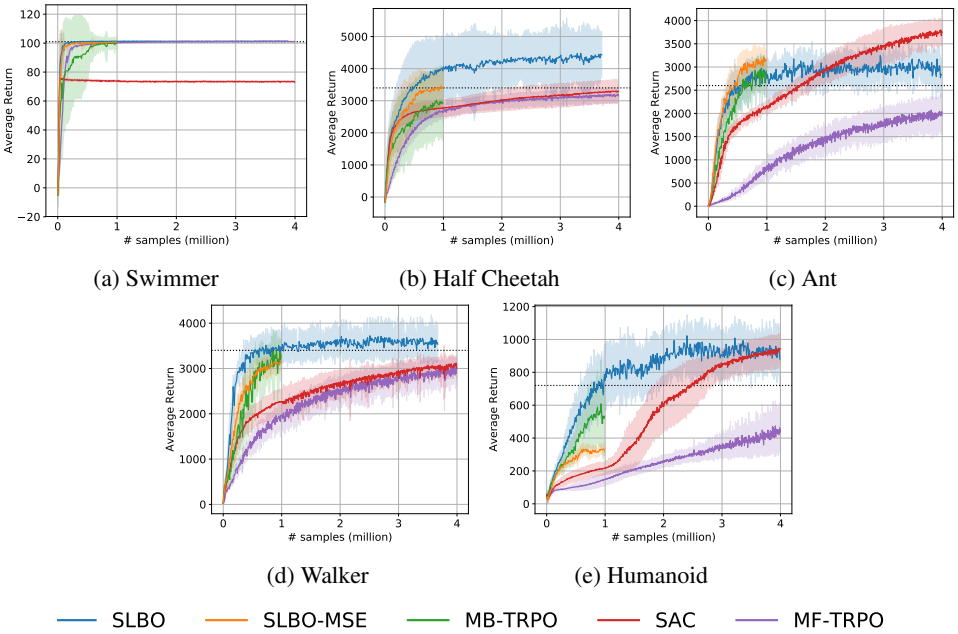


Figure 4: Comparison among SLBO (ours), SLBO with squared ℓ^2 model loss (SLBO-MSE), vanilla model-based TRPO (MB-TRPO), model-free TRPO (MF-TRPO), and Soft Actor-Critic (SAC) with more samples than in Figure 1. SLBO, SAC, MF-TRPO are trained with 4 million real samples. We average the results over 10 different random seeds, where the solid lines indicate the mean and shaded areas indicate one standard deviation. The dotted reference lines are the total rewards of MF-TRPO after 8 million steps.

G SAMPLE COMPLEXITY BOUNDS

In this section, we extend Theorem 3.1 to a final sample complexity result. For simplicity, let $L^{\pi,M}_{\pi_{\text{ref}},\delta} = V^{\pi,M} - D_{\pi_k,\delta}(M,\pi)$ be the lower bound of $V^{\pi,M^\star}$. We omit the subscript δ when it's clear from contexts. When D satisfies (R1), we have that,

$$V^{\pi,M^\star} \geq L^{\pi,M}_{\pi_{\text{ref}},\delta} \qquad \forall \pi \text{ s.t. } d(\pi,\pi_{\text{ref}}) \leq \delta \tag{G.1}$$

When D satisfies (R3), we use $\widehat{L}^{\pi,M}_{\pi_{\text{ref}},\delta}$ to denotes its empirical estimates. Namely, we replace the expectation in equation (R3) by empirical samples $\tau^{(1)},\ldots,\tau^{(n)}$. In other words, we optimize

$$\pi_{k+1}, M_{k+1} = \operatorname*{argmax}_{\pi\in\Pi,\ M\in\mathcal{M}} \widehat{L}^{\pi,M}_{\pi_{\text{ref}},\delta} = \ V^{\pi,M} - \frac{1}{n}\sum_{i=1}^{n} f(\widehat{M},\pi,\tau^{(i)}) \tag{G.2}$$

instead of equation (3.3).

Let p be the total number of parameters in the policy and model parameterization. We assume that we have a discrepancy bound $D_{\pi_{\text{ref}}}(\pi,M)$ satisfying (R3) with a function f that is bounded with $[-B_f,B_f]$ and that is L_f-Lipschitz in the parameters of π and M. That is, suppose π is parameterized by θ and M is parameterized by ϕ, then we require $|f(M_\phi,\phi_\theta,\tau) - f(M_{\phi'},\phi_{\theta'},\tau)| \leq L_f(\|\phi-\phi'\|_2^2 + \|\theta-\theta'\|^2)$ for all τ, $\theta,\theta',\phi,\phi'$. We note that L_f is likely to be exponential in dimension due to the recursive nature of the problem, but our bounds only depends on its logarithm. We also restrict our attention to parameters in an Euclidean ball $\{\theta : \|\theta\|_2 \leq B\}$ and $\{\phi : \|\phi\|_2 \leq B\}$. Our bounds will be logarithmic in B.

We need the following definition of approximate local maximum since with sampling error we cannot hope to converge to the exact local maximum.

Definition G.1. We say π is a (δ,ε)-local maximum of $V^{\pi,M^\star}$ with respect to the constraint set Π and metric d, if for any $\pi'\in\Pi$ with $d(\pi,\pi')\leq\delta$, we have $V^{\pi,M^\star} \geq V^{\pi',M^\star} - \varepsilon$.

We show a sample complexity bounds that scales linearly in p and logarithmically in L_f, B and B_f.

Theorem G.2. *Let $\varepsilon > 0$. In the setting of Theorem 3.1, under the additional assumptions above, suppose we use $n = O(B_f p\log(BL_f/\varepsilon)/\varepsilon^2)$ trajectories to estimate the discrepancy bound in Algorithm 1. Then, for any t, if π_t is not a (δ,ε)-local maximum, then the total reward will increase in the next step: with high probability,*

$$V^{\pi_{t+1},M^\star} \geq V^{\pi_t,M^\star} + \varepsilon/2 \tag{G.3}$$

As a direct consequence, suppose the maximum possible total reward is B_R and the initial total reward is 0, then for some $T = O(B_R/\varepsilon)$, we have that π_T is a (δ,ε)-local maximum of the $V^{\pi,M^\star}$.

Proof. By Hoeffding's inequality, we have for fix π and $\widehat{M}$, with probability $1-n^{O(1)}$ over the randomness of $\tau^{(1)},\ldots,\tau^{(n)}$,

$$\left|\frac{1}{n}\sum_{i=1}^{n} f(\widehat{M},\pi,\tau^{(i)}) - \mathop{\mathbb{E}}_{\tau\sim\pi_{\text{ref}},M^\star}[f(\widehat{M},\pi,\tau)]\right| \leq 4\sqrt{\frac{B_f\log n}{n}}. \tag{G.4}$$

In more succinct notations, we have $|\widehat{D}_{\pi_k,\delta}(M,\pi) - D_{\pi_k,\delta}(M,\pi)| \leq 4\sqrt{\frac{B_f\log n}{n}}$, and therefore

$$|\widehat{L}^{\pi,M} - L^{\pi,M}| \leq 4\sqrt{\frac{B_f\log n}{n}}. \tag{G.5}$$

By a standard ε-cover + union bound argument, we can prove the uniform convergence: with high probability (at least $1-n^{O(1)}$) over the choice of $\tau^{(1)},\ldots,\tau^{(n)}$, for all policy and model, for all policy π and dynamics M,

$$|\widehat{L}^{\pi,M} - L^{\pi,M}| \leq 4\sqrt{\frac{B_f p\log(nBL_f)}{n}} = \varepsilon/4. \tag{G.6}$$

Suppose at iteration t, we are at policy π_t which is not a (δ, ε)-local maximum of $V^{\pi, M^\star}$. Then, there exists π' such that $d(\pi', \pi_t) \leq \delta$ and

$$V^{\pi', M^\star} \geq V^{\pi_t, M^\star} + \varepsilon. \tag{G.7}$$

Then, we have that

$$\begin{aligned}
V^{\pi_{t+1}, M^\star} &\geq L_{\pi_t}^{\pi_{t+1}, M_{t+1}} && \text{(by equation (G.1))} \\
&\geq \widehat{L}_{\pi_t}^{\pi_{t+1}, M_{t+1}} - \varepsilon/4 && \text{(by uniform convergence, equation (G.6))} \\
&\geq \widehat{L}_{\pi_t}^{\pi', M^\star} - \varepsilon/4 && \text{(by the definition of } \pi_{t+1}, M_{t+1}\text{)} \\
&\geq L_{\pi_t}^{\pi', M^\star} - \varepsilon/2 && \text{(by uniform convergence, equation (G.6))} \\
&= V^{\pi', M^\star} - \varepsilon/2 && \text{(by equation (R2))} \\
&= V^{\pi_t, M^\star} + \varepsilon/2 && \text{(by equation (G.7))}
\end{aligned}$$

Note that the total reward can only improve by $\varepsilon/2$ for at most $O(B_R/\varepsilon)$ steps. Therefore, in the first $O(B_R/\varepsilon)$ iterations, we must have hit a solution that is a (δ, ε)-local maximum. This completes the proof.

□