

Model-Based Value Expansion for Efficient Model-Free Reinforcement Learning

Vladimir Feinberg¹ Alvin Wan¹ Ion Stoica¹ Michael I. Jordan¹ Joseph E. Gonzalez¹ Sergey Levine¹

Abstract

Recent model-free reinforcement learning algorithms have proposed incorporating learned dynamics models as a source of additional data with the intention of reducing sample complexity. Such methods hold the promise of incorporating imagined data coupled with a notion of model uncertainty to accelerate the learning of continuous control tasks. Unfortunately, they rely on heuristics that limit usage of the dynamics model. We present *model-based value expansion*, which controls for uncertainty in the model by only allowing imagination to fixed depth. By enabling wider use of learned dynamics models within a model-free reinforcement learning algorithm, we improve value estimation, which, in turn, reduces the sample complexity of learning.

1. Introduction

Recent progress in model-free (MF) reinforcement learning has demonstrated the capacity of rich value function approximators to master complex tasks. However, these model-free approaches require access to an impractically large number of training interactions for most real-world problems. In contrast, model-based (MB) methods can quickly arrive at near-optimal control with learned models under fairly restricted dynamics classes (Dean et al., 2017). In settings with nonlinear dynamics, fundamental issues arise with the MB approach: *complex dynamics demand high-capacity models, which in turn are prone to overfitting in precisely those low-data regimes where they are most needed*. Model inaccuracy is further exacerbated by the difficulty of long-term dynamics predictions (Fig. 1).

The MF and MB approaches have distinct strengths and weaknesses: expressive value estimation MF methods can

achieve good asymptotic performance but have poor sample complexity, while MB methods exhibit efficient learning but struggle on complex tasks. In this paper, we seek to *reduce sample complexity while supporting complex nonlinear dynamics by combining MB and MF learning techniques through disciplined model use for value estimation*.

We present model-based value expansion (MVE), a hybrid algorithm that uses a dynamics model to simulate the short-term horizon and Q-learning to estimate the long-term value beyond the simulation horizon. This improves Q-learning by providing higher-quality target values for training. Splitting value estimates into a near-future MB component and a distant-future MF component offers a model-based value estimate that (1) creates a decoupled interface between value estimation and model use and (2) does not require differentiable dynamics.

In this scheme, our trust in the model informs the selection of the horizon up to which we believe the model can make accurate estimates. This horizon is an interpretable metric applicable to many dynamics model classes. Prior approaches that combine model-based and model-free RL, such as model-based acceleration (Gu et al., 2016), incorporate data from the model directly into the model-free RL algorithm, which we show can lead to poor results. Alternatively, imagination-augmented agents (I2A) offloads all uncertainty estimation and model use into an implicit neural network training process, inheriting the inefficiency of model-free methods (Racanière et al., 2017).

By incorporating the model into Q-value target estimation, we only require the model to be able to make forward predictions. In contrast to stochastic value gradients (SVG), we make no differentiability assumptions on the underlying

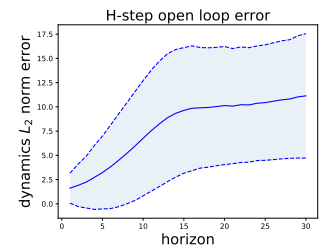


Figure 1. The mean L_2 error, with standard deviation, for open-loop dynamics prediction of a *cheetah* agent increases in the prediction horizon.

¹EECS Department, UC Berkeley, Berkeley, CA, USA. Correspondence to: Vladimir Feinberg <vladf@berkeley.edu>.

dynamics, which usually include non-differentiable phenomena such as contact interactions (Heess et al., 2015). Using an approximate, few-step simulation of a reward-dense environment, the improved value estimate provides enough signal for faster learning for an actor-critic method (Fig. 6). Our experimental results show that our method can outperform both fully model-free RL algorithms and prior approaches to combining real-world and model-based roll-outs for accelerated learning. The remainder of this paper is organized along the following contributions:

- A general method to reduce the value estimation error in algorithms with model-free critics (Sec. 3).
- An evaluation of the reduced sample complexity resulting from better value estimation (Sec. 4.1).
- A characterization of the error that arises from value-based methods applied to imaginary data (Sec. 4.2).

2. Background

A deterministic Markov decision process (MDP) is characterized by a set of possible actions \mathcal{A} and states \mathcal{S} . Its dynamics are deterministic and are described by a state transition function $f : \mathcal{S} \times \mathcal{A} \rightarrow \mathcal{S}$ and the bounded reward function $r : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$. In this work, we consider continuous state and action spaces. For a deterministic policy $\pi : \mathcal{S} \rightarrow \mathcal{A}$, the action-value function $Q^\pi(s_0, a_0) = \sum_{t=0}^{\infty} \gamma^t r_t$ gives the deterministic γ -discounted cumulative reward, where $s_{t+1} = f(s_t, \pi(s_t))$, $a_t = \pi(s_t)$, and $r_t = r(s_t, a_t)$. We will use \hat{Q} to refer to some estimate of Q^π . Finally, recall that we may express the value function $V^\pi(s) = Q^\pi(s, \pi(s))$. We will denote state-value estimates with \hat{V} . The policy π is implicitly parameterized by θ . We indicate this with subscripts π_θ when necessary.

The objective is to maximize $J_{d_0}(\pi) = \mathbb{E}_{d_0}[V^\pi(S)]$, where $S \sim d_0$ is sampled from the initial state distribution. With access to only off-policy states from an exploration policy, summarized by some empirical distribution ν , we consider the proxy J_ν instead, as is typical (Degris et al., 2012).

2.1. Deterministic Policy Gradients (DPG)

Silver et al. (2014) describe an off-policy actor-critic MF algorithm. For a deterministic, parameterized policy π_θ and parameterized critic \hat{Q} (we leave its parameters implicit), Silver et al. (2014) prove that a tractable form of the policy gradient $\nabla_\theta J_{d_0}(\theta)$ exists when f is continuously differentiable with respect to actions taken by the policy. This result, the deterministic policy gradient theorem, expresses the policy gradient as an expectation over on-policy data.

To encourage exploration, the data collected may be off-policy. To reconcile the use of off-policy data with an on-

policy estimator, Silver et al. (2014) briefly note an analogous off-policy policy improvement theorem in the case of finite state spaces (Degris et al., 2012).

2.2. Continuous DPG

In the continuous case, the conditions under which the off-policy policy improvement theorem holds are distinct from the discrete case. We emphasize these conditions here and provide a proof for the statement in Silver et al. (2014) inspired by the discrete case (Degris et al., 2012).

Theorem 2.1 (Off-Policy Deterministic Continuous Policy Improvement Theorem). *Let β be an off-policy distribution of states and set*

$$g_\theta(s) = D_{\pi(s)}(\theta)^\top \nabla_a Q^{\pi_\theta}(s, a)|_{a=\pi(s)}, \quad (1)$$

with $D_{\pi(s)}(\theta)$ the Jacobian of $\theta \mapsto \pi_\theta(s)$. Then $\mathbb{E}_\beta g_\theta(S)$ ascends $J_\beta(\theta)$ if the following conditions hold β -almost always: (1) Q^{π_θ} must be differentiable with respect to the action at $(s, \pi_\theta(s))$, (2) the Jacobian $D_{\pi(s)}(\theta)$ must exist, and (3) $g_\theta(s)$ is nonzero.

Proof. See the appendix (Sec. B).

Critically, we do not require DPG’s assumptions of continuous differentiability in the reward or dynamics functions. This will allow model-based value expansion to be both theoretically and practically compatible with arbitrary forward models, including discrete event simulators, and settings with non-differentiable physics.

Finally, a proxy \hat{Q} may be used as a replacement for the true Q^π term to approximate the ascent direction g_θ in Eq. (1), as described in deep deterministic policy gradients (DDPG) (Lillicrap et al., 2016).

3. Model-Based Value Expansion

MVE improves value estimates for a policy π by assuming we an approximate dynamical model $\hat{f} : \mathcal{S} \times \mathcal{A} \rightarrow \mathcal{S}$ and the true reward function r . Such an improved value estimate can be used in training a critic for faster task mastery in reward-dense environments (Fig. 6). We assume that the model is accurate to depth H ; that is, for a fixed policy π , we may use \hat{f} to evaluate (“imagine”) future transitions that occur when taking actions according to π with $\hat{f}^\pi = \hat{f}(\cdot, \pi(\cdot))$. We use these future transitions to estimate value.

Definition 3.1 (H -Step Model Value Expansion). *Using the imagined rewards reward $\hat{r}_t = r(\hat{s}_t, \pi(\hat{s}_t))$ obtained from our model $\hat{s}_t = \hat{f}^\pi(\hat{s}_{t-1})$ under π we define the H -step model value expansion (MVE) estimate for the value of a given state $V^\pi(s_0)$:*

$$\hat{V}_H(s_0) = \sum_{t=0}^{H-1} \gamma^t \hat{r}_t + \gamma^H \hat{V}(\hat{s}_H). \quad (2)$$

The H -step model value expansion in Definition 3.1 decomposes the state-value estimate at s_0 into the component predicted by learned dynamics $\sum_{t=0}^{H-1} \gamma^t \hat{r}_t$ and the tail, estimated by \hat{V} . This approach can be extended to stochastic policies and dynamics by integrating Eq. 2 with a Monte Carlo method, assuming a generative model for the stochastic dynamics and policy. Since \hat{r}_t is derived from actions $\hat{a}_t = \pi(\hat{s}_t)$, this is an on-policy use of the model and, even in the stochastic case, MVE would not require importance weights, as opposed to the case of using traces generated by off-policy trajectories.

While MVE is most useful in settings where the H step horizon is not sparse, even in sparse reward settings predicting the future state will improve critic accuracy. Finally, MVE may be applied to state-action estimates: in this case $\hat{a}_0 \triangleq a_0$ while $\hat{a}_t = \pi(\hat{s}_t)$ for $t > 0$ and $\hat{V}(\hat{s}_H) \triangleq \hat{Q}(\hat{s}_H, \hat{a}_H)$ and may be used for estimating $\hat{Q}_H(s_0, a_0)$ with a state-action critic \hat{Q} .

3.1. Value Estimation Error

In this section, we discuss the conditions under which the MVE estimate improves the mean-squared error (MSE):

$$\text{MSE}_\nu(V) = \mathbb{E}_{S \sim \nu} \left[(V(S) - V^\pi(S))^2 \right], \quad (3)$$

of the original critic \hat{V} with respect to ν , a distribution of states. We emphasize that, even assuming an ideal model, the mere combination of MB and MF does not guarantee improved estimates. Further, while our analysis will be conducted on state-value estimation, it may be naturally extended to state-action-value estimation.

Recall the H -depth model accuracy assumption: if we observe a state s_0 we may imagine $\hat{s}_t \approx s_t$, so $\hat{a}_t \approx a_t$ and $\hat{r}_t \approx r_t$ for $t \leq H$. If the modelling assumption holds for some fixed s_0 , we have

$$\hat{V}_H(s_0) - V^\pi(s_0) \approx \gamma^H \left(\hat{V}(s_H) - V^\pi(s_H) \right).$$

This translates to an expression for MVE MSE,

$$\text{MSE}_\nu(\hat{V}_H) \approx \gamma^{2H} \text{MSE}_{(f^\pi)^H \nu}(\hat{V}),$$

where $(f^\pi)^H \nu$ denotes the pushforward measure resulting from playing π H times starting from states in ν . This informal presentation demonstrates the relation of MVE MSE to the underlying critic MSE by assuming the model is nearly ideal. We verify that the informal reasoning above is sound in the presence of model error.

Theorem 3.1 (Model-Based Value Expansion Error). Define s_t, a_t, r_t to be the states, actions, and rewards resulting from following policy π using the true dynamics

f starting at $s_0 \sim \nu$ and analogously define $\hat{s}_t, \hat{a}_t, \hat{r}_t$ using the learned dynamics \hat{f} in place of f . Let the reward function r be L_r -Lipschitz and the value function V^π be L_V -Lipschitz. Let ϵ be an upper bound

$$\max_{t \in [H]} \mathbb{E} \left[\|\hat{s}_t - s_t\|^2 \right] \leq \epsilon^2,$$

on the model risk for an H -step rollout. Then

$$\text{MSE}_\nu(\hat{V}_H) \leq c_1^2 \epsilon^2 + (1 + c_2 \epsilon) \gamma^{2H} \text{MSE}_{(f^\pi)^H \nu}(\hat{V}),$$

where c_1, c_2 grow at most linearly in L_r, L_V and are independent of H for $\gamma < 1$. We assume $\text{MSE}_{(f^\pi)^H \nu}(\hat{V}) \geq \epsilon^2$ for simplicity of presentation, but an analogous result holds when the critic outperforms the model.

Proof. See the appendix (Sec. C).

Sufficient conditions for improving on the original critic $\text{MSE}_\nu(\hat{V})$ are then small $\epsilon, \gamma < 1$, and the critic being at least as accurate on imagined states as on those sampled from ν ,

$$\text{MSE}_\nu(\hat{V}) \geq \text{MSE}_{(f^\pi)^H \nu}(\hat{V}). \quad (4)$$

However, if ν is an arbitrary sampling distribution, such as the one generated by an exploratory policy, the inequality of Eq. (4) rarely holds. In particular, this naive choice results in poor performance overall (Fig. 3) and counteracts the benefit of model-based reward estimates, even assuming a perfect oracle dynamics model (Fig. 6). Thus, if \hat{V} is trained on Bellman error from ν , then the **distribution mismatch** between $(f^\pi)^H \nu$ and ν eclipses the benefit of γ^{2H} . In effect, any model-based approach evaluating the critic \hat{V} on imaginary states from $(f^\pi)^H \nu$ must be wary of only training its critic on the real distribution ν . We believe this insight can be incorporated into a variety of works similar to MVE, such as value prediction networks (Oh et al., 2017).

We propose a solution to the distribution-mismatch problem by observing that the issue disappears if $(f^\pi)^H \nu = \nu$, i.e., the training distribution ν is a fixed point of f^π . In practice, given an arbitrary off-policy distribution of state-action pairs β we set $\nu = \mathbb{E}[(f^\pi)^T \beta]$ as an approximation to the fixed point, where $T \sim \text{Uniform}\{0, \dots, H-1\}$. If we then sample a state $\hat{s}_T | T \sim (f^\pi)^T \beta$, our model accuracy assumption dictates that we may accurately simulate states $\{\hat{s}_{T+i}\}_{i=1}^{H-T}$ arising from playing π starting at \hat{s}_T . These simulated states can be used to construct k -step MVE targets $\hat{V}_k(\hat{s}_T)$ accurately while adhering to assumptions about the model by setting $k = H - T$. These targets can then be used to train \hat{V} on the entire support of ν , instead of just β .

Since we do not have access to the true MSE of \hat{V} , we minimize its Bellman error with respect to ν , using this error as a proxy. In this context, using a target $\hat{V}_k(\hat{s}_T)$ is equivalent

to training \hat{V} with imagined TD- k error. This **TD- k trick** enables us to skirt the distribution mismatch problem to the extent that ν is an approximate fixed point. We find that the TD- k trick greatly improves task performance relative to training the critic on β alone (Fig. 3).

3.2. Deep Reinforcement Learning Implementation

In the preceding section, we presented an analysis that motivates our model-based value expansion approach. In this section, we will present a practical implementation of this approach for high-dimensional continuous deep reinforcement learning. We **demonstrate how to apply MVE in a general actor-critic setting to improve the target Q -values, with the intention of achieving faster convergence**. Our implementation relies on a parameterized actor π_θ and critic Q_φ , but note that the separate parameterized actor may be removed if it is feasible to compute $\pi(s) = \operatorname{argmax}_a Q_\varphi(s, a)$.

We assume that the actor critic method supplies a differentiable actor loss ℓ_{actor} and a critic loss $\ell_{\text{critic}}^{\pi, Q}$. These losses are functions of θ, φ as well as transitions $\tau = (S, A, R, S')$ sampled from some distribution \mathcal{D} . For instance, in DDPG, the θ gradient of $\mathbb{E}_{\mathcal{D}} [\ell_{\text{actor}}(\theta, \varphi, \tau)] = \mathbb{E}_{\mathcal{D}} [Q_\varphi(S, \pi_\theta(S))]$ approximately ascends $J_{\mathcal{D}}$ per the deterministic continuous policy improvement theorem (Lillicrap et al., 2016). The DDPG critic loss depends on a target actor π and critic Q : $\ell_{\text{critic}}^{\pi, Q}(\varphi, \tau) = (Q_\varphi(S, A) - (R + \gamma Q(S', \pi(S'))))^2$.

MVE relies on our approximate fixed point construction from an empirical distribution of transitions β . Recall our approximation from the previous section, which relies on the current policy to imagine up to H steps ahead: $\nu(\theta', \hat{f}) = \frac{1}{H} \sum_{t=0}^{H-1} (\hat{f}^{\pi_{\theta'}})^t \beta$, where $\hat{f}^{\pi_{\theta'}}(\tau) = (S', A', r(S', A'), \hat{f}(S', A'))$ and $A' = \pi_{\theta'}(S')$. Thus, sampling transitions from ν is equivalent to sampling from any point up to H imagined steps into the future, when starting from a state sampled from β . The MVE-augmented method follows the usual actor-critic template, but critic training uses MVE targets and transitions sampled from ν (Alg. 1).

We imagine rollouts with the target actor, whose parameters are exponentially weighted averages of previous iterates, for parity with DDPG, which uses a target actor to compute target value estimates. Taking $H = 0$, $\nu(\theta', \hat{f}) = \beta$, we recover the original actor-critic algorithm. Our implementation uses multi-layer fully-connected neural networks to represent both the Q -function and the policy. We use the actor and critic losses described by DDPG (Lillicrap et al., 2016).

Importantly, we do not use an imagination buffer to save simulated states, and instead generate simulated states on-the-fly by sampling from $\nu(\theta', \hat{f})$. We perform a stratified sampling from $\nu(\theta', \hat{f})$, with H dependent samples at a time, for each $t \in \{0, \dots, H-1\}$ in Line 11. First, we

Algorithm 1 Use model-based value expansion to enhance critic target values in a generic actor-critic method abstracted by $\ell_{\text{actor}}, \ell_{\text{critic}}$. Parameterize π, Q with θ, φ , respectively. We assume \hat{f} is selected from some class of dynamics models and the space \mathcal{S} is equipped with a norm.

```

1: procedure MVE-AC(initial  $\theta, \varphi$ )
2:   Initialize targets  $\theta' = \theta, \varphi' = \varphi$ 
3:   Initialize the replay buffer  $\beta \leftarrow \emptyset$ 
4:   while not tired do
5:     Collect transitions from any exploratory policy
6:     Add observed transitions to  $\beta$ 
7:     Fit the dynamics

       $\hat{f} \leftarrow \operatorname{argmin}_f \mathbb{E}_{\beta} [\|f(S, A) - S'\|^2]$ 

8:     for a fixed number of iterations do
9:       sample  $\tau_0 \sim \beta$ 
10:      update  $\theta$  with  $\nabla_{\theta} \ell_{\text{actor}}(\pi_{\theta}, Q_{\varphi}, \tau_0)$ 
11:      imagine future transitions for  $t \in [H-1]$ 

           $\tau_t = \hat{f}^{\pi_{\theta'}}(\tau_{t-1})$ 

12:       $\forall k$  define  $\hat{Q}_k$  as the  $k$ -step MVE of  $Q_{\varphi'}$ 
13:      update  $\varphi$  with  $\nabla_{\varphi} \sum_t \ell_{\text{critic}}^{\pi_{\theta'}, \hat{Q}_k}(\varphi, \tau_t) / H$ 
14:      update targets  $\theta', \varphi'$  with some decay
15:    end for
16:  end while
17:  return  $\theta, \varphi$ 
18: end procedure

```

sample a real transition $\tau_0 = (s_{-1}, a_{-1}, r_{-1}, s_0)$ from β , the empirical distribution of transitions observed from interacting with the environment according to an exploratory policy. We use the learned dynamics model \hat{f} to generate \hat{s}_t and \hat{r}_t . Since $\pi_{\theta'}$ changes during the joint optimization of θ, φ , these simulated states are discarded immediately after the batch. We then take a stochastic ∇_{φ} step to minimize ν -based Bellman error of Q_{φ} ,

$$\frac{1}{H} \sum_{t=-1}^{H-1} \left(Q_{\varphi}(\hat{s}_t, \hat{a}_t) - \left(\sum_{k=t}^{H-1} \gamma^{k-t} \hat{r}_k + \gamma^H Q_{\varphi'}(\hat{s}_H, \hat{a}_H) \right) \right)^2,$$

where $Q_{\varphi'}$ and $\hat{a}_t = \pi_{\theta'}(\hat{s}_t)$ use target parameter values (Lines 11-13 of Alg. 1). As such, every observation of the Bellman error always relies on some real data.

For the dynamics \hat{f} , we use a neural network with 8 layers of 128 neurons each with a fixed 10^{-3} learning rate trained to predict the difference in real-vector-valued states, similar to previous work (Kurutach et al., 2018). While we expect more accurate and carefully tuned models to allow us to use large H , even a weak model with shared hyperparameters across all tasks from a flexible class suffices to

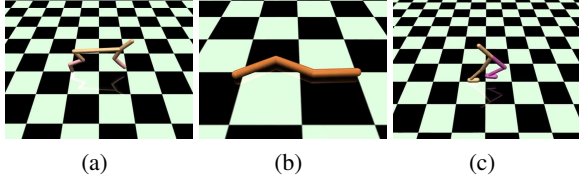


Figure 2. We use fully-observable analogues of typical Gym environments (Brockman et al., 2016). By default, the environments excerpt certain observed dimensions, such as the abscissa of (a), which are requisite for reward calculation. This is done as a supervised imposition of policy invariance to certain dimensions, which we remove for full observability. (a) shows *Fully Observable Half Cheetah* (*cheetah*); (b), *Fully Observable Swimmer* (*swimmer*); (c), *Fully Observable Walker, 2-D* (*walker*). Agents are rewarded for forward motion. We detail our changes in the Appendix (Sec. A).

demonstrate our point.

4. Results

We evaluate MVE on several continuous control environments. Experimental details are in Sec. A. We would like to verify the following:

- Does MVE improve estimates of Q^π ?
- Do the improved estimates result in faster mastery?
- Does the TD- k trick resolve distribution mismatch?

In all of our experiments, we tune the baseline, DDPG, and report its best performance. For exploration, we use parameter-space noise (Plappert et al., 2017). Every experiment and each setting uses the same adaptive parameter-space noise standard deviation target, so exploration is controlled to be the same in all trials. We then add the MVE extension (we do not tune DDPG parameters to MVE performance). To evaluate the effect of the TD- k , we evaluate against the naive approach of using H -step MVE \hat{Q} estimates for Bellman error on states sampled from β . This is equivalent to using only the first term for $t = 0$ in the sum of Line 13 of Alg. 1, i.e., updating critic parameters φ with the gradient $\nabla_{\varphi} \ell_{\text{critic}}^{\pi_{\theta'}, \hat{Q}^H}(\varphi, \tau_0)$. Without the TD- k trick the model is still used to simulate to depth H , but the opportunity to learn on a distribution of additional support is neglected.

We plot the mean learning curve, along with the standard deviation. We smooth each graph with a 20 point window (evaluation data is collected at intervals of at most 10^3 timesteps).

4.1. Performance

First, we evaluate that MVE-DDPG (with the TD- k trick) improves in terms of raw reward performance by comparing its learning curves to those of the original DDPG, MVE-DDPG without the TD- k trick, and imagination buffer (IB) approaches (Kalweit & Boedecker, 2017). We find that MVE-DDPG outperforms the alternatives (Fig. 3).¹ Our result shows that incorporating synthetic samples from a learned model can drastically improve the performance of model-free RL, greatly reducing the number of samples required to attain good performance. As illustrated by the comparison to the IB baseline, this improvement is obtained only when carefully incorporating this synthetic experience via a short horizon and the TD- k trick. As we will discuss in the next section, the specific design decisions here are critical for good results, which helps to explain the lack of success with learned neural network models observed with related methods in prior work (Gu et al., 2016).

MVE-DDPG improves on similar approaches, such as MA-DDPG, by its treatment of synthetic data obtained from the dynamics model. This alternative approach adds simulated data back into a separate imagination buffer, effectively modifying β from Alg. 1 into a mixture between β and simulated data (where the mixture is dependent on the relative number of samples taken from each buffer). This is problematic because the policy changes during training, so the data from this mixed distribution of real and fake data is stale relative to ν in terms of representing actions that would be taken by π . In our implementation of MA-DDPG, we do not reuse imagined states in this manner, but MVE-DDPG still outperforms MA-DDPG. We suspect this is due to two factors: (1) the staleness of the imaginary states in the IB approach, and (2) the delicate interaction between using more imaginary data and overtraining the actor. In order to use the additional synthetic data, an IB approach must take more gradient steps with imaginary batches. On the other hand, since MVE-DDPG uses a gradient averaged over both real and simulated data, the choice to make additional gradient steps becomes an independent consideration dependent on the stability of the actor-critic method being trained.

4.2. MVE as Critic Improvement

We make sure that MVE can make use of accurate models to improve the critic value estimate (Fig. 4). The improved critic performance results in faster training compared to the $H = 0$ DDPG baseline on *cheetah*. Also, we replicate the density plot from DDPG to analyze Q accuracy directly,

¹We run IB with an imaginary-to-real ratio of 4, as used for 2 of 3 environments in (Kalweit & Boedecker, 2017). We tested lower ratios on *cheetah* but found they performed worse. Note that for parity with DDPG we ran MVE-DDPG with only 4 gradient steps.

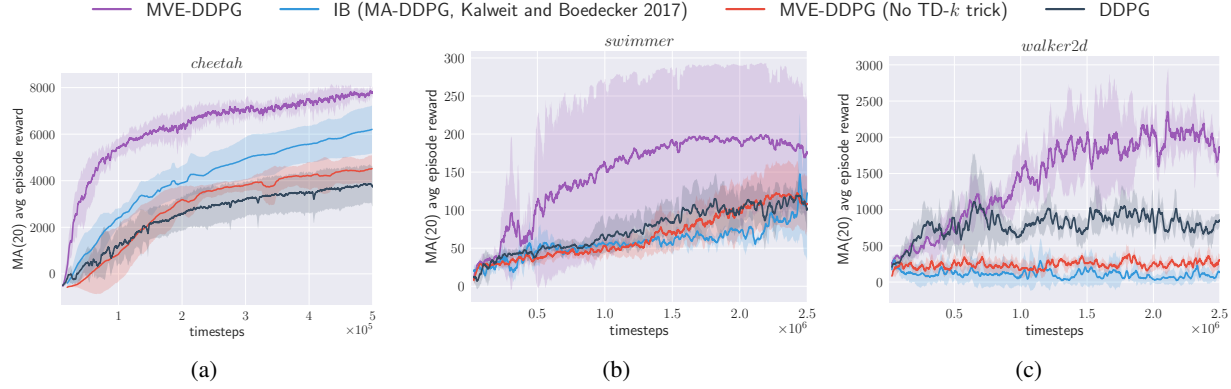


Figure 3. Learning curves comparing MVE with learned dynamics (purple), MVE without the TD- k trick (orange), IB (Kalweit & Boedecker, 2017) (blue), and DDPG (black) on (a) *cheetah*, (b) *swimmer*, and (c) *walker*. We used $H = 10$ for (a,b), but found the same dynamics class inadequate for walker, reducing *walker* experiments to $H = 3$ reduces the improvement MVE has to offer over DDPG, but it still exhibits greater robustness to the poor model fit than IB. Note that we use MA-DDPG, not MA-BDDPG in the IB approach. The bootstrap estimation in MA-BDDPG may reduce model use in some cases, so it is possible that MA-BDDPG would have improved performance in *walker*, where the learned dynamics are poor compared to the other environments.

from which it is clear that MVE improves the critic by providing better target values (Fig. 5).

In addition, we verify that the TD- k trick is essential to training \hat{Q} appropriately. To do so, we conduct an ablation analysis on the *cheetah* environment: we hold all parameters constant and substitute the learned dynamics model \hat{f} with the true dynamics model f , making model error zero. If $\text{MSE}_{\beta}(\hat{Q}) \approx \text{MSE}_{(f^{\pi})^H \beta}(\hat{Q})$, the MVE estimates must improve exponentially in H , even without the TD- k trick. However, this is not the case. With the TD- k trick, increasing H yields increasing but diminishing returns (Fig. 6). Without the adjustment for distribution mismatch, past a certain point, increasing H hurts performance. Because the dynamics model is ideal in these cases, the only difference is that the critic \hat{Q} is trained on the distribution of states $\nu = \frac{1}{H} \sum_{t=0}^{H-1} (f^{\pi})^t \beta$ instead of β , where β is the empirical distribution resulting from the replay buffer. Since the TD- k trick increases the support of the training data on which \hat{Q} is trained, the function class for the critic may need to have sufficient capacity to capture the new distribution, but we did not find this to be an issue in our experiments.

5. Related Work

A number of prior works have sought to incorporate dynamics models into model-free RL and value estimation. We observe three general approaches: (1) direct incorporation of dynamics into the value gradient, (2) use of imagination as additional training data, and (3) use of imagination as context for value estimation.

(1) Stochastic value gradients (SVG) uses its model for improved credit assignment from traces of real trajec-

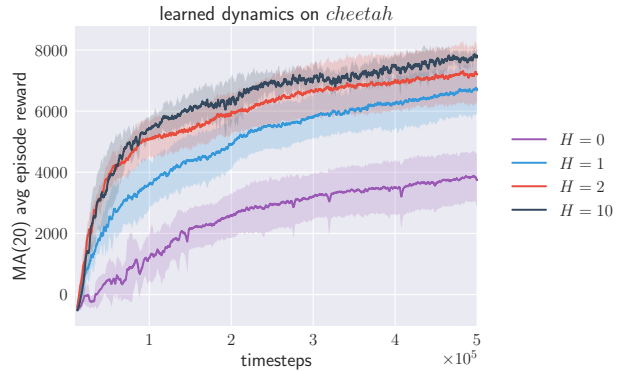


Figure 4. Learning curves from *cheetah* for MVE-DDPG with learned dynamics at different model horizons H .

ries (Heess et al., 2015). By applying a differentiable dynamics model to real data only, SVG avoids instability from planning with an overfitted dynamics model, since the model is used for its gradients rather than its forward predictions. A major limitation of this approach is that the dynamics model can now only be used to retrieve information that is already present in the observed data, albeit with lower variance, so the actual improvement in efficiency is relatively small. Applied to a horizon of length one, given off-policy data from β , SVG(1) estimates $J_{\beta}(\pi) \approx \mathbb{E} \left[w(A|S) \left(r(S, A) + \gamma \hat{V}(\hat{S}) \right) \right]$ with $S, A, r(S, A)$ observed from the transition taken according to some sampling distribution β and \hat{S} following the probabilistic dynamics model (the stochastic value gradient is achieved by using the reparameterization trick to differenti-

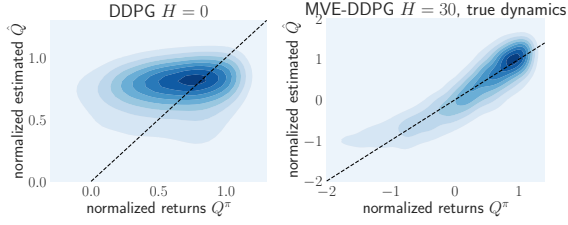


Figure 5. We plot the true observed cumulative discounted returns against those predicted by the critic for *cheetah* at the end of training (both values are normalized), reconstructing Fig. 3 of (Lillicrap et al., 2016). The dotted black line represents unity. An ideal critic concentrates over the line. We verify that with MVE at $H = 30$ with the true dynamics model trains a critic with improved Q values relative to the original DDPG algorithm. Both of the above runs use a reduced mini-batch size because oracle dynamics are expensive to compute.

ate the approximation). Heess et al. (2015) does not consider the case of SVG(n) for off-policy data and $n > 1$, likely due the use of the importance weight $w(a|s) = \frac{\pi(a|s)}{\beta(a|s)}$, which may vanish for model-based expansion of longer lengths. This is not a problem for the on-policy SVG(∞), but by the authors’ own account SVG(∞) is less sample-efficient than SVG(1).

(2) Alternatively, Dyna-like approaches use the learned model for imagination rollouts (Sutton, 1990). Providing imagined data to model-free value estimation algorithms is potentially a more robust use of a potentially erroneous view of a task’s dynamics compared to planning. A number of follow-up methods have expanded on this idea. For instance, in model-based acceleration (MBA), imagination rollouts are used as additional training data for a parameterized value network (Gu et al., 2016). The original MBA proposal adds a replay buffer with the imagined data. Imagined states may then be used as the starting point for further imagination at later iterations. This reuse violates the rule of trusting the model for only H steps of simulation. The authors find that this approach does not work well with neural network models, and settle for locally linear ones. Even with this modification, the model is deactivated in the middle of training heuristically to attain the best results. Our analysis and comparisons shed light on why this approach may be ineffective.

One approach to limit model use automatically is model-assisted bootstrapped DDPG (MA-BDDPG) (Kalweit & Boedecker, 2017). This has two modifications from MBA. The first is that the imaginary states are not used as the starting point for future imaginary rollouts (this change alone is referred to as MA-DDPG). The second change adds an estimate of approximate critic uncertainty via the bootstrap. The imagined states are then used for training

with frequency increasing in estimated critic uncertainty. Thus model use is limited except in scenarios where it is needed disambiguate uncertainty in the model-free critic estimate. Fig. 4 compares MVE to the IB approach of MA-DDPG. We do not use bootstrap to limit model use as MA-BDDPG does. The IB approach is in part representative of MBA as well, though MBA explicitly states that it is not compatible with neural network dynamics. MBA also uses the model for local planning in its imagination rollouts, and this is not done here.

Another prior approach is model-ensemble trust region policy optimization (ME-TRPO) (Kurutach et al., 2018). Here, real data is used to train the dynamics model only, and a TRPO algorithm learns from imagination rollouts alone. ME-TRPO limits over-training to the imagined data by using an ensemble metric for policy performance. In other words, ME-TRPO bootstraps the dynamics model prediction, as opposed to the critic as in MA-BDDPG. However, dynamics model predictions are treated equally regardless of imagination depth, and the lack of use of real data for policy training may become problematic if dynamics models fail to capture phenomena occurring in reality. In any case, ME-TRPO demonstrates the success of using the bootstrap for estimating the degree of uncertainty in models, and we believe that this notion can be incorporated in future work combined with MVE. **ME-TRPO is orthogonal and complementary to our method.**

(3) The use of imagination as context for value estimation is most similar to MVE, but existing literature does not address model accuracy. In particular, unlike the aforementioned works, which focus on quantification of model uncertainty to limit model use, an end-to-end approach, I2A, avoids explicit reasoning about model inaccuracy (Racanière et al., 2017). I2A supplies imagined rollouts as inputs to critic and actor networks, which are free to interpret the imaginary data in whatever way they learn to. A related approach is proposed by value prediction networks (VPN), which expand encoded state predictions and perform backups on a set of expanded imagined paths to compute an improved target estimate (Oh et al., 2017). Both I2A and VPN are tested on planning problems in discrete spaces. However, in continuous spaces, some degree of dynamics prediction error is unavoidable and may affect I2A stability or sample complexity as the network must learn to deal with uncertain rollouts. In addition, VPN crucially relies on the ability to perform stable backups on imagined paths with respect to current and future actions. These backups amount to optimizing over the action space to maximize the value function, which may be problematic in the continuous case. However, we believe that a stable model-based value expansion, which may be achieved with the TD- k trick, along with careful optimization, may make a VPN-like approach viable in continuous contexts.

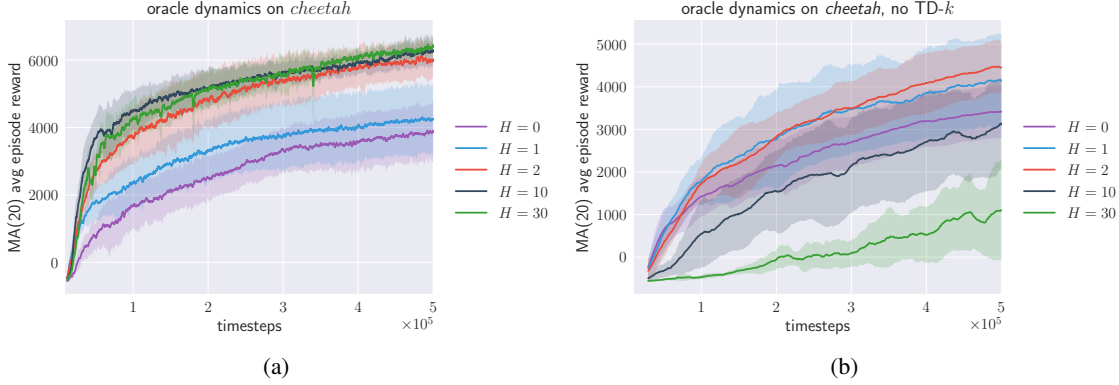


Figure 6. Learning curves for the *cheetah* environment for MVE-DDPG with an ideal, oracle dynamical model at different horizons H of model prediction. We examine performance (a) with and (b) without the TD- k trick. $H = 0$ implies no model use; this is the original DDPG. First, (a) exemplifies that improving value estimation with a model has a marked effect on performance in dense reward environments and offers an upper bound to the improvement that can result from learned-dynamics MVE. Note that as mentioned in Fig. 5 the batch size for oracle dynamics evaluations was reduced out of computational necessity, so these curves are not comparable to Fig. 4. The diminishing returns for increases in H that we observe further emphasize that model improvement is captured even with a short horizon. Second, (b) demonstrates the value of the TD- k trick: for small H , distribution mismatch is small so (b) still shows a performance improvement, but as H increases we lose the monotonic improvements observed in (a).

Finally, we note that MVE has some high-level similarity to n -step return methods. Both use short horizon rollouts to improve the value estimate, frequently used as a target value in computing Bellman error. Usually, n -step return methods are on-policy: they estimate target Q -values at some state-action pair as $\sum_{t=0}^{H-1} \gamma^t r_t + \gamma^H \hat{Q}(s_H, a_H)$ for a trace of rewards r_t in an observed H -step trajectory (Peng & Williams, 1994). The main difference of MVE from n -step return methods, explicit state prediction via dynamics modeling, is essential because it enables faster learning by use of off-policy data. Recent work, path consistency learning (PCL), relieves the on-policy requirements: PCL trains on a soft consistency error that is compatible with off-policy trajectories (Nachum et al., 2017). We believe that model-based extension of PCL with imagined rollouts may be possible following the algorithm design lessons MVE recommends. In particular, using an imagination buffer of rollouts to augment paths chosen for consistency learning may result in stale data and requires model use to be tempered by the stability of the algorithm to taking many gradient steps without gathering additional data. An MVE approach could augment paths with imagined branches that would be followed by the current policy, and the path consistency value would be the average over such branches. In other words, MVE may be complimentary to PCL, similar to how off-policy actor-critic methods can still be improved with MVE value estimation from imagined data, even though they are already off-policy.

6. Conclusion

In this paper, we introduce the model-based value expansion (MVE) method, an algorithm for incorporating predictive models of system dynamics into model-free value function estimation. Our approach provides for improved sample complexity on a range of continuous action benchmark tasks, and our analysis illuminates some of the design decisions that are involved in choosing how to combine model-based predictions with model-free value function learning.

Existing approaches following a general Dyna-like approach to using imagination rollouts for improvement of model-free value estimates either use stale data in an imagination buffer or use the model to imagine past horizons where the prediction is accurate. Multiple heuristics (Kurutach et al., 2018; Kalweit & Boedecker, 2017) have been proposed to reduce model usage to combat such problems, but these techniques generally involve a complex combination of uncertainty estimation and additional hyperparameters and may not always appropriately restrict model usage to reasonable horizon lengths. MVE offers a single, simple, and adjustable notion of model trust (H), and fully utilizes the model to that extent. MVE also demonstrates that state dynamics prediction enables on-policy imagination via the TD- k trick starting from off-policy data.

Our work justifies further exploration in model use for model-free sample complexity reduction. In particular, estimating uncertainty in the dynamics model explicitly would enable automatic selection of H . To deal with sparse reward signals, we also believe it is important to consider explo-

ration with the model, not just refinement of value estimates. Finally, MVE admits extensions into domains with probabilistic dynamics models and stochastic policies via Monte Carlo integration over imagined rollouts.

Acknowledgements

The authors are very thankful for the valuable feedback from Roberto Calandra, Gregory Kahn, Anusha Nagabandi, and Richard Liaw. This research is supported in part by DHS Award HSHQDC-16-3-00083, NSF CISE Expeditions Award CCF-1139158, and gifts from Alibaba, Amazon Web Services, Ant Financial, CapitalOne, Ericsson, GE, Google, Huawei, Intel, IBM, Microsoft, Scotiabank, Splunk and VMware.

References

- Brockman, Greg, Cheung, Vicki, Pettersson, Ludwig, Schneider, Jonas, Schulman, John, Tang, Jie, and Zaremba, Wojciech. OpenAI gym. *arXiv preprint arXiv:1606.01540*, 2016.
- Dean, Sarah, Mania, Horia, Matni, Nikolai, Recht, Benjamin, and Tu, Stephen. On the sample complexity of the linear quadratic regulator. *arXiv preprint arXiv:1710.01688*, 2017.
- Degrís, Thomas, White, Martha, and Sutton, Richard S. Off-policy actor-critic. *arXiv preprint arXiv:1205.4839*, 2012.
- Gu, Shixiang, Lillicrap, Timothy, Sutskever, Ilya, and Levine, Sergey. Continuous deep q-learning with model-based acceleration. In *International Conference on Machine Learning*, pp. 2829–2838, 2016.
- Heess, Nicolas, Wayne, Gregory, Silver, David, Lillicrap, Tim, Erez, Tom, and Tassa, Yuval. Learning continuous control policies by stochastic value gradients. In *Advances in Neural Information Processing Systems*, pp. 2944–2952, 2015.
- Kalweit, Gabriel and Boedecker, Joschka. Uncertainty-driven imagination for continuous deep reinforcement learning. In Levine, Sergey, Vanhoucke, Vincent, and Goldberg, Ken (eds.), *Proceedings of the 1st Annual Conference on Robot Learning*, volume 78 of *Proceedings of Machine Learning Research*, pp. 195–206. PMLR, 13–15 Nov 2017.
- Kurutach, Thanard, Clavera, Ignasi, Duan, Yan, Tamar, Aviv, and Abbeel, Pieter. Model-ensemble trust-region policy optimization. *International Conference on Learning Representations*, 2018.
- Lillicrap, Timothy P, Hunt, Jonathan J, Pritzel, Alexander, Heess, Nicolas, Erez, Tom, Tassa, Yuval, Silver, David, and Wierstra, Daan. Continuous control with deep reinforcement learning. *International Conference on Learning Representations*, 2016.
- Nachum, Ofir, Norouzi, Mohammad, Xu, Kelvin, and Schuurmans, Dale. Bridging the gap between value and policy based reinforcement learning. In *Advances in Neural Information Processing Systems*, pp. 2772–2782, 2017.
- Oh, Junhyuk, Singh, Satinder, and Lee, Honglak. Value prediction network. In *Advances in Neural Information Processing Systems*, pp. 6120–6130, 2017.
- Peng, Jing and Williams, Ronald J. Incremental multi-step Q-learning. In *Machine Learning Proceedings 1994*, pp. 226–232. Elsevier, 1994.
- Plappert, Matthias, Houthoofd, Rein, Dhariwal, Prafulla, Sidor, Szymon, Chen, Richard Y, Chen, Xi, Asfour, Tamim, Abbeel, Pieter, and Andrychowicz, Marcin. Parameter space noise for exploration. *NIPS Deep Reinforcement Learning Workshop*, 2017.
- Racanière, Sébastien, Weber, Théophane, Reichert, David, Buesing, Lars, Guez, Arthur, Rezende, Danilo Jimenez, Badia, Adrià Puigdomènech, Vinyals, Oriol, Heess, Nicolas, Li, Yujia, et al. Imagination-augmented agents for deep reinforcement learning. In *Advances in Neural Information Processing Systems*, pp. 5694–5705, 2017.
- Silver, David, Lever, Guy, Heess, Nicolas, Degrís, Thomas, Wierstra, Daan, and Riedmiller, Martin. Deterministic policy gradient algorithms. In *International Conference on Machine Learning*, 2014.
- Sutton, Richard S. Integrated architectures for learning, planning, and reacting based on approximating dynamic programming. In *International Conference on Machine Learning*, pp. 216–224. Morgan Kaufmann, 1990.
- Todorov, Emanuel, Erez, Tom, and Tassa, Yuval. 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.

A. Experiment Details

To simulate MVE behavior with an ideal, oracle dynamics model, we needed to use the true environment, the MuJoCo simulation, for dynamics predictions. To make experiments tractable we implemented our own `cython`-based interface to the physics engine, based off of `mujoco-py` (Todorov et al., 2012). However, we were able to use the usual `gym` interface when applying learned dynamics.²

All runs wait until $10 \cdot 10^3$ timesteps to be collected until DDPG training starts. We wait until $5 \cdot 10^3$ timesteps have been collected until model training begins. We perform 4 gradient steps on both the dynamics and DDPG networks per collected timestep, though more may be possible. Batch sizes were always 512, except when training on the oracle dynamics model, where they were reduced to 32 out of computational necessity. In all experiments with learned dynamics, we use 4 seeds. To make up for the increased variance in the runs, we ran all experiments with the true dynamics model with 6 seeds.

Table 1. Final tuned DDPG parameters for all the environments. LR refers to learning rate. The networks had 2 hidden layers of 64 units each. Adaptive parameter-space noise was fixed at 0.2. Decay refers to target decay

ENVIRONMENT	ACTOR LR	CRITIC LR	DECAY
<i>cheetah</i>	10^{-3}	10^{-3}	10^{-2}
<i>swimmer</i>	10^{-4}	10^{-3}	10^{-2}
<i>walker</i>	10^{-4}	10^{-4}	10^{-3}

B. Off-Policy Deterministic Continuous Policy Improvement Theorem

Definition. A direction $\mathbf{v} \in \mathbb{R}^n$ ascends $\Phi : \mathbb{R}^n \rightarrow \mathbb{R}$ at \mathbf{x} , or is an ascent direction for Φ at \mathbf{x} , if there exists an $\epsilon > 0$ such that for all $\epsilon' \in (0, \epsilon)$, $\Phi(\mathbf{x} + \epsilon' \mathbf{v}) > \Phi(\mathbf{x})$.

Remark. If $\nabla \Phi(\mathbf{x})$ exists and is nonzero then the ascent directions are exactly those directions having positive inner product with the gradient.

Let $\pi_\theta : \mathcal{S} \rightarrow \mathcal{A}$ be a deterministic policy for an MDP admitting a transition measure f (which may be a delta measure, corresponding to deterministic dynamics) from $\mathcal{S} \times \mathcal{A}$ to \mathcal{S} , which are all open subsets of a Euclidean space. For any distribution of states β , let the off-policy objective be $J_\beta(\theta) = \mathbb{E}_\beta[V^{\pi_\theta}(S)]$.

Theorem (Degris et al., 2012). Define:

$$g_\theta(s) = D_{\pi_{(\cdot)}(s)}(\theta)^\top \nabla_a Q^{\pi_\theta}(s, a)|_{a=\pi_\theta(s)},$$

²We based environments off of gym commit 4c460ba6c8959dd8e0a03b13a1ca817da6d4074f.

where $D_{\pi_{(\cdot)}(s)}(\theta)$ is the Jacobian of π with respect to θ for a fixed s . To improve legibility, we use shortened notation

$$\nabla_a Q^{\pi_\theta}(s, a)|_{a=\pi_\theta(s)} = \nabla_a Q,$$

keeping in mind that the latter is implicitly a function of θ, s . We require that $g_\theta(s)$ exists and is nonzero for β -almost every s . Then $\mathbb{E}_\beta[g_\theta(S)]$ ascends J_β at θ .

Proof. Fix s where $g_\theta(s) \neq \mathbf{0}$. Notice $g_\theta(s)$ is the gradient at $\varphi = \theta$ for the function

$$h(\varphi) = \langle \pi_\varphi(s), \nabla_a Q \rangle.$$

Then for ϵ sufficiently small if $\theta' = \theta + \epsilon g_\theta(s)$ then $h(\theta') > h(\theta)$. Then by a Taylor expansion of $Q^\pi(s, a)$ for fixed s around a at $a = \pi_\theta(s)$, for ϵ sufficiently small:

$$\begin{aligned} Q^{\pi_{\theta'}}(s, \pi_{\theta'}(s)) &= Q^{\pi_\theta}(s, \pi_\theta(s)) + \langle \nabla_a Q, \pi_{\theta'}(s) - \pi_\theta(s) \rangle + o(\epsilon) \\ &= Q^{\pi_\theta}(s, \pi_\theta(s)) + h(\theta') - h(\theta) + o(\epsilon) \end{aligned}$$

Now by the Taylor expansion of h around $\varphi = \theta$, swallowing the error term into the existing one:

$$\begin{aligned} Q^{\pi_{\theta'}}(s, \pi_{\theta'}(s)) &= Q^{\pi_\theta}(s, \pi_\theta(s)) + \langle g_\theta(s), \theta' - \theta \rangle + o(\epsilon) \\ &= Q^{\pi_\theta}(s, \pi_\theta(s)) + \epsilon \|g_\theta(s)\|^2 + o(\epsilon) \\ &> Q^{\pi_\theta}(s, \pi_\theta(s)) \end{aligned}$$

The last inequality holds only for ϵ sufficiently small. Now we may proceed in a manner nearly identical to (Degris et al., 2012).

$$\begin{aligned} J_\beta(\theta) &= \int d\beta(s) V^{\pi_\theta}(s) = \int d\beta(s) Q^{\pi_\theta}(s, \pi_\theta(s)) \\ &< \int d\beta(s) Q^{\pi_{\theta'}}(s, \pi_{\theta'}(s)) \\ &= \int d\beta(s) \left(r(s, \pi_{\theta'}(s)) + \gamma \int df(s'|s, \pi_{\theta'}(s)) V^{\pi_{\theta'}}(s') \right) \end{aligned}$$

Notice we may continue unrolling $V^{\pi_{\theta'}}(s') < r(s', a') + \gamma \int df(s''|s', a') V^{\pi_{\theta'}}(s'')$ where $a' = \pi_{\theta'}(s')$. Continuing in this manner ad infinitum, with $a_t = \pi_{\theta'}(s_t)$:

$$\begin{aligned} J_\beta(\theta) &< \int d\beta(s_0) r(s_0, a_0) + \gamma \int df(s_1|s_0, a_0) \\ &\quad \times \left(r(s_1, a_1) + \gamma \int df(s_2|s_1, a_1) \cdots \right) \end{aligned}$$

At this point, one may notice that the right-hand side is $J_\beta(\theta')$. \square

Remark. The requirement that $g_\theta(s)$ is almost surely nonzero is not an artificial result of the analysis: the DPG

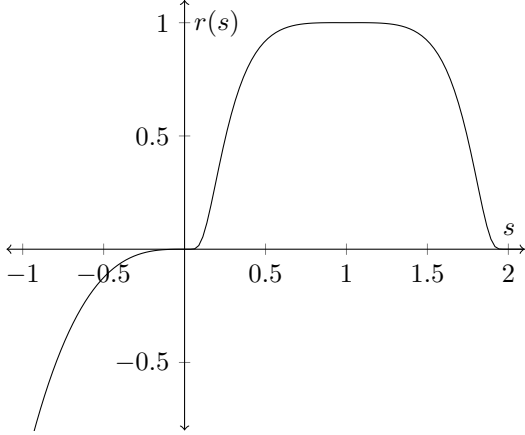
Table 2. Descriptions of the environments used for testing, all of which are based off of their corresponding gym environments with the usual unmodified rewards for forward motion.

ENVIRONMENT	gym BASE ENVIRONMENT	MODIFICATIONS	INPUT DIMENSION	OUTPUT DIMENSION
<i>cheetah</i>	HalfCheetah-v1	Add x coordinate	18	6
<i>swimmer</i>	Swimmer-v1	Add x, y coordinates	18	6
<i>walker</i>	Walker2d-v1	Add x coordinate	10	2

ascent direction requires local improvement everywhere for sufficiently small step size. Consider the following one-dimensional hill-climbing MDP with $\mathcal{S} = [-1, 2]$. An agent attempts to climb the reward function $r(s) = s^3 \mathbb{1}\{s < 0\} + \Phi(s - 1)$, where Φ is a quadratic bump function.

$$\Phi(s) = \begin{cases} \exp\left(\frac{-x^4}{1-x^2}\right) & x \in (-1, 1) \\ 0 & \text{otherwise} \end{cases}$$

We consider the linear policy class $\pi_\theta(s) = \theta$, where the action represents movement towards a state θ the agent finds desirable: the dynamics are given by agent movement from position s to position $s + \delta(\pi_\theta(s) - s)$.



Note that $r'(0), r''(0), r'''(0) = 0, 0, 6$ (r''' is the left derivative) while $r'(1 + \epsilon), r''(1 + \epsilon), r'''(1 + \epsilon) = -\Theta(\epsilon^3), -\Theta(\epsilon^2), -\Theta(\epsilon)$. Now, consider a pathological β with equal mass split between 0 and $1 + \epsilon$, for some small $\epsilon > 0$, and suppose the current $\theta = 0$. Because the agent has no local derivative information at $s = 0$, but a step to the left at $1 + \epsilon$ improves its situation, the direction $\mathbb{E}_\beta g_0(s)$ actually worsens performance: indeed, $g_0(0) = 0$ yet $g_0(1 + \epsilon) = -\Omega(\epsilon^3)$, so any step in the direction $\mathbb{E}_\beta g_0(s)$ results in $\theta < 0$. But because $|r'''(0)| > |r'''(1 + \epsilon)|$ for ϵ sufficiently small, any step in this direction reduces overall performance J_β !

C. Model-based Value Expansion Error Theorem

Theorem. Let $s_0 \sim \nu$ and define s_t, a_t, r_t to be the states, actions and rewards that come about from acting according to $\pi : \mathcal{S} \rightarrow \mathcal{A}$ in the true environment with dynamics f . Let $\hat{s}_t, \hat{a}_t, \hat{r}_t$ be constructed from starting at s_0 and using the learned dynamics \hat{f} to imagine an H -step roll-out. If the model has ϵ generalization error at depth H , i.e., $\max_{t \in [H]} \mathbb{E} \|\hat{s}_t - s_t\|^2 \leq \epsilon^2$, and r, V^π are L_r, L_V -Lipschitz functions of state, then

$$\text{MSE}_\nu(\hat{V}_H) \leq c_1^2 \epsilon^2 + (1 + c_2 \epsilon) \gamma^{2H} \text{MSE}_{(\hat{f}^\pi)^H \nu}(\hat{V}),$$

where c_1, c_2 grow at most linearly in L_r, L_V and are independent of H for $\gamma < 1$. We assume $\text{MSE}_{(\hat{f}^\pi)^H \nu}(\hat{V}) \geq \epsilon^2$ for simplicity of presentation, but an analogous result holds when the critic outperforms the model.

Recall we have the induced policy dynamics $f^\pi : \mathcal{S} \rightarrow \mathcal{S}$ from $f : \mathcal{S} \times \mathcal{A} \rightarrow \mathcal{S}$ where $f^\pi(\cdot) = f(\cdot, \pi(\cdot))$ (with analogous notation for \hat{f}^π). We denote the Lipschitz constants of $r(\cdot, \pi(\cdot))$ and V^π as L_r, L_V .

Proof. Fix s_0 . Recall Eq. 2:

$$\hat{V}_H(s_0) = \sum_{t=0}^{H-1} \gamma^t \hat{r}_t + \gamma^H \hat{V}(\hat{s}_H)$$

Then we may decompose $(\hat{V}_H(s_0) - V^\pi(s_0))^2$ as:

$$\left((\hat{M} - M) - \gamma^H (\hat{V}(\hat{s}_H) - V^\pi(s_H)) \right)^2,$$

where $M = \sum_{t=0}^{H-1} \gamma^t r_t$ denotes the model-based component, defined analogously for the imagined \hat{M} . We will then further rely on the decomposition $\hat{M} - M = \sum_{t=0}^{H-1} \gamma^t (\hat{r}_t - r_t)$.

We will require repeated appeals to the Cauchy-Schwarz inequality. For any L^2 random variables A, B , notice:

$$\begin{aligned} \mathbb{E}[(A - B)^2] &= \mathbb{E}A^2 - 2\mathbb{E}[AB] + \mathbb{E}B^2 \\ &\leq \mathbb{E}A^2 + 2\mathbb{E}^{1/2}A^2\mathbb{E}^{1/2}B^2 + \mathbb{E}B^2 \end{aligned}$$

Since each of the aforementioned decompositions results in a difference of squares, we can bound the MSE:

$$\begin{aligned} \text{MSE}_{\nu} \hat{V}_H &\leq \mathbb{E}(\hat{M} - M)^2 \\ &\quad + 2\gamma^H \sqrt{\mathbb{E}(\hat{M} - M)^2 \mathbb{E}(\hat{V}(\hat{s}_H) - V^\pi(s_H))^2} \\ &\quad + \gamma^{2H} \mathbb{E}(\hat{V}(\hat{s}_H) - V^\pi(s_H))^2 \end{aligned}$$

Then we bound all model-based terms, taking i, j over $[H] - 1$:

$$\mathbb{E}(\hat{M} - M)^2 \leq \sum_{i,j} \gamma^{2(i+j)} \sqrt{\mathbb{E}(\hat{r}_i - r_i)^2 \mathbb{E}(\hat{r}_j - r_j)^2}$$

Next, by smoothness, $\hat{r}_i - r_i \leq L_r \|\hat{s}_i - s_i\|$ almost surely for all i , so $\mathbb{E}(\hat{r}_i - r_i)^2 \leq L_r^2 \mathbb{E}\|\hat{s}_i - s_i\|^2$, so $\mathbb{E}(\hat{M} - M)^2 \leq \epsilon^2 c_1^2$, where the constant is bounded by $c_1^2 \triangleq \sum_{i,j} \gamma^{2(i+j)} \leq L_r^2 \min(H^2, (1 - \gamma^2)^{-2})$.

Then the cross term can be simplified by the assumption that the critic performs worse than the model (without this assumption, an additional $O(\epsilon)$ cross term remains).

$$\begin{aligned} 2\gamma^H \sqrt{\mathbb{E}(\hat{M} - M)^2 \mathbb{E}(\hat{V}(\hat{s}_H) - V^\pi(s_H))^2} &\leq \\ 2\gamma^H c_1 \epsilon \sqrt{\mathbb{E}(\hat{V}(\hat{s}_H) - V^\pi(s_H))^2} &\leq \\ 2\gamma^H c_1 \mathbb{E}(\hat{V}(\hat{s}_H) - V^\pi(s_H))^2 & \end{aligned}$$

At this point, we have for some c'_1 linear in L_r ,

$$\text{MSE}_{\nu} \hat{V}_H \leq c_1^2 \epsilon^2 + \gamma^{2H} (1 + c'_1) \mathbb{E}(\hat{V}(\hat{s}_H) - V^\pi(s_H))^2.$$

To bound the remaining term we use the same technique yet again, decomposing

$$\hat{V}(\hat{s}_H) - V^\pi(s_H) = (\hat{V}(\hat{s}_H) - V^\pi(\hat{s}_H)) - (V^\pi(s_H) - V^\pi(\hat{s}_H))$$

with Cauchy-Schwarz:

$$\begin{aligned} &\mathbb{E}(\hat{V}(\hat{s}_H) - V^\pi(s_H))^2 \\ &\leq \mathbb{E}(\hat{V}(\hat{s}_H) - V^\pi(\hat{s}_H))^2 \\ &\quad + 2\sqrt{\mathbb{E}(\hat{V}(\hat{s}_H) - V^\pi(\hat{s}_H))^2 \mathbb{E}(V^\pi(\hat{s}_H) - V^\pi(s_H))^2} \\ &\quad + \mathbb{E}(V^\pi(\hat{s}_H) - V^\pi(s_H))^2. \end{aligned}$$

Since $\mathbb{E}(\hat{V}(\hat{s}_H) - V^\pi(\hat{s}_H))^2 = \text{MSE}_{(\hat{f}^\pi)_{H\nu}} \hat{V}$ we finish by bounding $\mathbb{E}(V^\pi(\hat{s}_H) - V^\pi(s_H))^2 \leq L_V^2 \epsilon^2$ by relying on Lipschitz smoothness of V^π . \square