VARIANCE REDUCTION FOR POLICY GRADIENT METHODS WITH ACTION-DEPENDENT BASELINES

Anonymous authors
Paper under double-blind review

ABSTRACT

Policy gradient methods have enjoyed success in deep reinforcement learning but suffer from high variance of gradient estimates. The high variance problem is particularly exasperated in problems with long horizons or high-dimensional action spaces. To mitigate this issue, we derive an action-dependent baseline for variance reduction which fully exploits the structural form of the stochastic policy itself, and does not make any additional assumptions about the MDP. We demonstrate and quantify the benefit of the action-dependent baseline through both theoretical analysis as well as numerical results. Our experimental results indicate that action-dependent baselines allow for faster learning on standard reinforcement learning benchmarks as well as on high dimensional manipulation and multi-agent communication tasks.

1 INTRODUCTION

Deep reinforcement learning has achieved impressive results in recent years in domains such as video games from raw visual inputs (Mnih et al., 2015), board games (Silver et al., 2016), simulated control tasks (Schulman et al., 2016; Lillicrap et al., 2016; Rajeswaran & V. Kumar, 2017), and robotics (Levine et al., 2016). An important class of methods behind many of these success stories are policy gradient methods (Williams, 1992; Sutton et al., 2000; Kakade, 2002; Schulman et al., 2015; Mnih et al., 2016), which directly optimize parameters of a stochastic policy through local gradient information obtained by interacting with the environment using the current policy. Policy gradient methods operate by increasing the log probability of actions proportional to the future rewards influenced by these actions. On average, actions which perform better will acquire higher probability, and the policy's expected performance improves.

A critical challenge of policy gradient methods is the high variance of the gradient estimator. This high variance is caused in part due to difficulty in credit assignment to the actions which affected the future rewards. Such issues are further exacerbated in long horizon problems, where assigning credits properly becomes even more challenging. To reduce variance, a "baseline" is often employed, which allows us to increase or decrease the log probability of actions based on whether they perform better or worse than the average performance when starting from the same state. This is particularly useful in long horizon problems, since the baseline helps with temporal credit assignment by removing the influence of future actions from the total reward. A better baseline, which predicts the average performance more accurately, will lead to lower variance of the gradient estimator.

The key insight of this paper is that when the individual actions produced by the policy can be decomposed into multiple factors, we can incorporate this additional information into the baseline to further reduce variance. In particular, when these factors are conditionally independent given the current state, we can compute a separate baseline for each factor, whose value can depend on all quantities of interest except that factor. This serves to further help credit assignment by removing the influence of other factors on the rewards, thereby reducing variance. In other words, information about the other factors can provide a better evaluation of how well a specific factor performs. Such factorized policies are very common, with some examples listed below.

- In continuous control and robotics tasks, multivariate Gaussian policies with a diagonal covariance matrix are often used. In such cases, each action coordinate can be considered a factor. Similarly, factorized categorical policies are used in game domains like board games and Atari.

- In multi-agent and distributed systems, each agent deploys its own policy, and thus the actions of each agent can be considered a factor of the union of all actions (by all agents). This is particularly useful in the recent emerging paradigm of centralized learning and decentralized execution (Foerster et al., 2017; Lowe et al., 2017). In contrast to the previous example, where factorized policies are a common design choice, in these problems they are dictated by the problem setting.

We demonstrate that action-dependent baselines consistently improve the performance compared to baselines that use only state information. The relative performance gain is task-specific, but in certain tasks, we observe significant speed-up in the learning process. We evaluate our proposed method on standard benchmark continuous control tasks, as well as on a high dimensional door opening task with a five-fingered hand, and on a blind peg insertion POMDP task. We believe that our method will facilitate further applications of reinforcement learning methods in domains with extremely high-dimensional actions, including multi-agent systems. Videos and additional results of the paper are available at `https://sites.google.com/view/ad-baselines`.

2 Related works

Three main classes of methods for reinforcement learning include value-based methods (Watkins & Dayan, 1992), policy-based methods (Williams, 1992; Kakade, 2002; Schulman et al., 2015), and actor-critic methods (Konda & Tsitsiklis, 2000; Peters & Schaal, 2008; Mnih et al., 2016). Value-based and actor-critic methods usually compute a gradient of the objective through the use of critics, which are often biased, unless strict compatibility conditions are met (Sutton et al., 2000; Konda & Tsitsiklis, 2000). Such conditions are rarely satisfied in practice due to the use of stochastic gradient methods and powerful function approximators. In comparison, policy gradient methods are able to compute an unbiased gradient, but suffer from high variance. These methods are therefore usually less sample efficient, but can be more stable than critic-based methods (Duan et al., 2016).

A large body of work has investigated variance reduction techniques for policy gradient methods. One effective method to reduce variance without introducing bias is through using a baseline, which has been widely studied (Sutton & Barto, 1998; Weaver & Tao, 2001; Greensmith et al., 2004; Schulman et al., 2016). this factorization has not been studied in detail. A recently proposed algorithm, Q-Prop (Gu et al., 2017), makes use of an action-dependent control variate, a technique commonly used in Monte Carlo methods and recently adopted for RL. Since Q-Prop utilizes off-policy data, it has the potential to be more sample efficient than pure on-policy methods. However, Q-prop is significantly more computationally expensive, since it needs to perform a large number of gradient updates on the critic using the off-policy data, thus not suitable with fast simulators. In contrast, our formulation of action-dependent baselines has little computational overhead, and improves the sample efficiency compared to on-policy methods with state-only baseline.

The idea of using additional information in the baseline or critic has also been studied in other contexts. Methods such as Guided Policy Search (Levine & Koltun, 2013; Mordatch et al., 2015) and variants train policies that act on high dimensional observations like images, but use a more low dimensional encoding of the problem like joint positions during the training process. Recent efforts in multi-agent systems (Foerster et al., 2017; Lowe et al., 2017) also use additional information in the centralized training phase to speed-up learning. However, using the structure in the policy parameterization itself to enhance the learning speed, as we do in this work, has not been explored.

3 Preliminaries

In this section, we establish the notations used throughout this paper, as well as basic results for policy gradient methods, and variance reduction via baselines.

3.1 Notation

This paper assumes a discrete-time Markov decision process (MDP), defined by $(\mathcal{S}, \mathcal{A}, \mathcal{P}, r, \rho_0, \gamma)$, in which $\mathcal{S} \subseteq \mathbb{R}^n$ is an n-dimensional state space, $\mathcal{A} \subseteq \mathbb{R}^m$ an m-dimensional action space, $\mathcal{P} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}_+$ a transition probability function, $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ a bounded reward function,

$\rho_0 : \mathcal{S} \to \mathbb{R}_+$ an initial state distribution, and $\gamma \in (0, 1]$ a discount factor. The presented models are based on the optimization of a stochastic policy $\pi_\theta : \mathcal{S} \times \mathcal{A} \to \mathbb{R}_+$ parameterized by θ. Let $\eta(\pi_\theta)$ denote its expected return: $\eta(\pi_\theta) = \mathbb{E}_\tau[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)]$, where $\tau = (s_0, a_0, \ldots)$ denotes the whole trajectory, $s_0 \sim \rho_0(s_0)$, $a_t \sim \pi_\theta(a_t|s_t)$, and $s_{t+1} \sim \mathcal{P}(s_{t+1}|s_t, a_t)$ for all t. Our goal is to find the optimal policy $\arg\max_\theta \eta(\pi_\theta)$.

For a partially observable Markov decision process (POMDP), two more components are required, namely Ω, a set of observations, and $\mathcal{O} : \mathcal{S} \times \Omega \to \mathbb{R}_{\geq 0}$, the observation probability distribution. In the fully observable case, $\Omega \equiv \mathcal{S}$. Though the analysis in this article is written for policies over states, the same analysis can be done for policies over observations.

3.2 The Score Function (SF) Estimator

An important technique used in the derivation of the policy gradient is known as the score function (SF) estimator (Williams, 1992), which also comes up in the justification of baselines. Suppose that we want to estimate $\nabla_\theta \mathbb{E}_x[f(x)]$ where $x \sim p_\theta(x)$, and the family of distributions $\{p_\theta(x) : \theta \in \Theta\}$ has common support. Further suppose that $\log p_\theta(x)$ is continuous in θ. In this case we have

$$\begin{aligned}\nabla_\theta \mathbb{E}_x[f(x)] &= \nabla_\theta \int p_\theta(x) f(x) dx = \int p_\theta(x) \frac{\nabla_\theta p_\theta(x)}{p_\theta(x)} f(x) dx \\ &= \int p_\theta(x) \nabla_\theta \log p_\theta(x) f(x) dx = \mathbb{E}_x \left[\nabla_\theta \log p_\theta(x) f(x)\right]. \end{aligned} \tag{1}$$

3.3 Policy Gradient

The Policy Gradient Theorem (Sutton et al., 2000) states that

$$\nabla_\theta \eta(\pi_\theta) = \mathbb{E}_\tau \left[\sum_{t=0}^{\infty} \nabla_\theta \log \pi_\theta(a_t|s_t) \sum_{t'=t}^{\infty} \gamma^{t'-t} r_{t'}\right]. \tag{2}$$

For convenience, define $\rho_\pi(s) = (1-\gamma)\sum_{t=0}^{\infty} \gamma^t p(s_t = s)$ as the normalized state visitation frequency, and $\hat{Q}(s_t, a_t) = \sum_{t'=t}^{\infty} \gamma^{t'-t} r_{t'}$. We can rewrite the above equation as

$$\nabla_\theta \eta(\pi_\theta) = \mathbb{E}_{\rho_\pi, \pi}\left[\nabla_\theta \log \pi_\theta(a_t|s_t) \hat{Q}(s_t, a_t)\right]. \tag{3}$$

It is further shown that we can reduce the variance of this gradient estimator without introducing bias by subtracting off a quantity dependent on s_t from $\hat{Q}(s_t, a_t)$ (Williams, 1992; Greensmith et al., 2004).

$$\nabla_\theta \eta(\pi_\theta) = \mathbb{E}_{\rho_\pi, \pi}\left[\nabla_\theta \log \pi_\theta(a_t|s_t) \left(\hat{Q}(s_t, a_t) - b(s_t)\right)\right] \tag{4}$$

This is valid because, applying the SF estimator in the opposite direction, we have

$$\mathbb{E}_{a_t}\left[\nabla_\theta \log \pi_\theta(a_t|s_t) b(s_t)\right] = \nabla_\theta \mathbb{E}_{a_t}\left[b(s_t)\right] = 0 \tag{5}$$

4 Action-dependent baselines for factorized policies

In practice there can be rich internal structure in the policy parameterization. For example, for continuous control tasks, a very common parameterization is to make $\pi_\theta(a_t|s_t)$ a multivariate Gaussian with diagonal variance, in which case each dimension a_t^i of the action a_t is conditionally independent of other dimensions, given the current state s_t. Another example is when the policy outputs a tuple of discrete actions with factorized categorical distributions. In the following subsections, we show that such structure can be exploited to further reduce the variance of the gradient estimator without introducing bias by changing the form of the baseline. Then, we derive the optimal action-dependent baseline for a class of problems and analyze the suboptimality of non-optimal baselines in terms of variance reduction. We then propose several practical baselines for implementation purposes. Even if this conditional independence does not hold (say for Gaussians with general covariance structure), as long as we can decompose the action into multiple factors, our analysis still holds, despite yielding a different baseline. Finally, we give an exposition on how action-dependent baselines can be combined with the Generalized Advantage Estimator (GAE) (Schulman et al., 2016) to smoothly interpolate the bias-variance trade-off curve.

4.1 BASELINES FOR CONDITIONALLY INDEPENDENT ACTIONS

First, we start with the conditionally independent case. Assuming an m-dimensional action space, we have $\pi_\theta(a_t|s_t) = \prod_{i=1}^m \pi_\theta(a_t^i|s_t)$. Hence

$$\nabla_\theta \eta(\pi_\theta) = \mathbb{E}_{\rho_\pi,\pi}\left[\nabla_\theta \log \pi_\theta(a_t|s_t)\hat{Q}(s_t,a_t)\right] = \mathbb{E}_{\rho_\pi,\pi}\left[\sum_{i=1}^m \nabla_\theta \log \pi_\theta(a_t^i|s_t)\hat{Q}(s_t,a_t)\right] \tag{6}$$

In this case, we can set b_i, the baseline for the ith factor, to depend on all other actions in addition to the state. Let a_t^{-i} denote all dimensions other than i in a_t and denote the ith baseline by $b_i(s_t, a_t^{-i})$. Due to conditional independence, we have

$$\mathbb{E}_{a_t}\left[\nabla_\theta \log \pi_\theta(a_t^i|s_t) b(s_t, a_t^{-i})\right] = \mathbb{E}_{a_t^{-i}}\left[\nabla_\theta \mathbb{E}_{a_t^i}\left[b_i(s_t, a_t^{-i})\right]\right] = 0 \tag{7}$$

Hence we can use the following gradient estimator

$$\nabla_\theta \eta(\pi_\theta) = \mathbb{E}_{\rho_\pi,\pi}\left[\sum_{i=1}^m \nabla_\theta \log \pi_\theta(a_t^i|s_t)\left(\hat{Q}(s_t,a_t) - b_i(s_t, a_t^{-i})\right)\right] \tag{8}$$

4.2 OPTIMAL ACTION-DEPENDENT BASELINE

In this section, we derive the optimal action-dependent baseline and show that it is better than the state-only baseline. We seek the optimal baseline to minimize the variance of the policy gradient estimate. First, we write out the variance of the policy gradient under any action-dependent baseline. Let us define $\nabla \eta_i(\pi_\theta) := \mathbb{E}_{\rho_\pi,\pi}\left[\nabla_\theta \log \pi_\theta(a_t^i|s_t)\left(\hat{Q}(s_t,a_t) - b_i(s_t, a_t^{-i})\right)\right]$ and $z_i := \nabla_\theta \log \pi_\theta(a_t^i|s_t)$. For simplicity of exposition, we make the following assumption:

$$\nabla_\theta \log \pi_\theta(a_t^i|s_t)^T \nabla_\theta \log \pi_\theta(a_t^j|s_t) = z_i^T z_j \approx 0, \quad \forall i \neq j \tag{9}$$

which translates to meaning that different subsets of parameters strongly influence different action dimensions or factors. This is true in case of distributed systems by construction, and also true in a single agent system if different action coordinates are strongly influenced by different policy network channels. Under this assumption, we have:

$$\text{Var}(\nabla_\theta \eta(\pi_\theta)) = \mathbb{E}_{\rho_\pi,\pi}\left[\sum_i \sum_j z_i^T z_j \left(\hat{Q}(s_t,a_t) - b_i(s_t, a_t^{-i})\right)\left(\hat{Q}(s_t,a_t) - b_j(s_t, a_t^{-j})\right)\right] \tag{10}$$

$$= \mathbb{E}_{\rho_\pi,\pi}\left[\sum_i z_i^T z_i \left(\hat{Q}(s_t,a_t) - b_i(s_t, a_t^{-i})\right)^2\right] \tag{11}$$

$$= \sum_i \text{Var}(\nabla_\theta \eta_i(\pi_\theta)) \tag{12}$$

The overall variance is minimized when each component variance is minimized. We now derive the optimal baselines $b_i^*(s_t, a_t^{-i})$ which minimize each respective component.

$$\begin{aligned}
\text{Var}(\nabla_\theta \eta_i(\pi_\theta)) &= \mathbb{E}_{\rho_\pi,\pi}\left[z_i^T z_i \left(\hat{Q}(s_t,a_t) - b_i(s_t, a_t^{-i})\right)^2\right] \\
&= \mathbb{E}_{\rho_\pi,\pi}\left[z_i^T z_i \left(\hat{Q}(s_t,a_t)^2 - 2b_i(s_t, a_t^{-i})Q(s_t,a_t) + b_i(s_t, a_t^{-i})\right)^2\right] \\
&= \mathbb{E}_{\rho_\pi,\pi}\left[z_i^T z_i \hat{Q}(s_t,a_t)^2\right] \\
&\quad + \mathbb{E}_{\rho_\pi,a_t^{-i}}\left[-2b_i(s_t, a_t^{-i})\mathbb{E}_{a_t^i}\left[z_t^T z_i Q(s_t,a_t)\right] + b_i(s_t, a_t^{-i})^2 \mathbb{E}_{a_t^i}\left[z_t^T z_i\right]\right]
\end{aligned}$$

Having written down the expression for variance under any action-dependent baseline, we seek the optimal baseline that would minimize this variance.

$$\frac{\partial}{\partial b_i}\left[\text{Var}(\nabla_\theta \eta_i(\pi_\theta))\right] = 0 \tag{13}$$

$$\implies b_i^*(s_t, a_t^{-i}) = \frac{\mathbb{E}_{a_t^i}\left[z_t^T z_i Q(s_t, a_t)\right]}{\mathbb{E}_{a_t^i}\left[z_t^T z_i\right]} \tag{14}$$

The optimal action-dependent baseline is:

$$\boxed{b_i^*(s_t, a_t^{-i}) = \frac{\mathbb{E}_{a_t^i}\left[\nabla_\theta \log \pi_\theta(a_t^i|s_t)^T \nabla_\theta \log \pi_\theta(a_t^i|s_t)\hat{Q}(s_t, a_t)\right]}{\mathbb{E}_{a_t^i}\left[\nabla_\theta \log \pi_\theta(a_t^i|s_t)^T \nabla_\theta \log \pi_\theta(a_t^i|s_t)\right]}} \tag{15}$$

Since the optimal action-dependent baseline is different for different action coordinates, it is outside the family of state-dependent baselines barring pathological cases.

4.3 Variance reduction improvement

We now turn to quantifying the reduction in variance of the policy gradient estimate under the optimal baseline derived above. Let $\text{Var}^*(\nabla_\theta \eta(\pi_\theta))$ denote the variance resulting from the optimal action-dependent baseline, and let $\text{Var}(\nabla_\theta \eta(\pi_\theta))$ denote the variance resulting from another baseline $b(s_t, a_t)$, which may be suboptimal or action-independent. We use the notations:

$$Z_i := Z_i(s_t, a_t^{-i}) = \mathbb{E}_{a_t^i}\left[\nabla_\theta \log \pi_\theta(a_t^i|s_t)^T \nabla_\theta \log \pi_\theta(a_t^i|s_t)\right] \tag{16}$$

$$Y_i := Y_i(s_t, a_t^{-i}) = \mathbb{E}_{a_t^i}\left[\nabla_\theta \log \pi_\theta(a_t^i|s_t)^T \nabla_\theta \log \pi_\theta(a_t^i|s_t)\hat{Q}(s_t, a_t)\right] \tag{17}$$

$$X_i := X_i(s_t, a_t^{-i}) = \mathbb{E}_{a_t^i}\left[\nabla_\theta \log \pi_\theta(a_t^i|s_t)^T \nabla_\theta \log \pi_\theta(a_t^i|s_t)\hat{Q}(s_t, a_t)^2\right] \tag{18}$$

Finally, define the variance improvement $I := \text{Var}(\nabla_\theta \eta(\pi_\theta)) - \text{Var}^*(\nabla_\theta \eta_i(\pi_\theta))$. Using these definitions, the variance can be re-written as:

$$\text{Var}(\nabla_\theta \eta(\pi_\theta)) = \sum_i \mathbb{E}_{\rho_\pi, a_t^{-i}}\left[X_i - 2b_i(s_t, a_t^{-i})Y_i + b_i(s_t, a_t^{-i})^2 Z_i\right] \tag{19}$$

Furthermore, the variance of the gradient with the optimal baseline can be written as

$$\text{Var}^*(\nabla_\theta \eta(\pi_\theta)) = \sum_i \mathbb{E}_{\rho_\pi, a_t^{-i}}\left[X_i - \frac{Y_i^2}{Z_i}\right] \tag{20}$$

The difference in variance can be calculated as:

$$I = \sum_i \left(\mathbb{E}_{\rho_\pi, a_t^{-i}}\left[X_i - 2b_i(s_t, a_t^{-i})Y_i + b_i(s_t, a_t^{-i})^2 Z_i\right] - (\mathbb{E}_{\rho_\pi, a_t^{-i}}\left[X_i - \frac{Y_i^2}{Z_i}\right])\right) \tag{21}$$

$$= \sum_i \mathbb{E}_{\rho_\pi, a_t^{-i}}\left[-2b_i(s_t, a_t^{-i})Y_i + b_i(s_t, a_t^{-i})^2 Z_i + \frac{Y_i^2}{Z_i}\right] \tag{22}$$

$$= \sum_i \mathbb{E}_{\rho_\pi, a_t^{-i}}\left[\left(b_i(s_t, a_t^{-i})\sqrt{Z_i} - \frac{Y_i}{\sqrt{Z_i}}\right)^2\right] \tag{23}$$

$$= \sum_i \mathbb{E}_{\rho_\pi, a_t^{-i}}\left[Z_i\left(b_i(s_t, a_t^{-i}) - \frac{Y_i}{Z_i}\right)^2\right] \tag{24}$$

$$= \sum_i \mathbb{E}_{\rho_\pi, a_t^{-i}}\left[Z_i\left(b_i(s_t, a_t^{-i}) - b_i^*(s_t, a_t^{-i})\right)^2\right] \tag{25}$$

$$= \sum_i \mathbb{E}_{\rho_\pi, a_t^{-i}}\left[\mathbb{E}_{a_t^i}\left[\nabla_\theta \log \pi_\theta(a_t^i|s_t)^T \nabla_\theta \log \pi_\theta(a_t^i|s_t)\right]\left(b_i(s_t, a_t^{-i}) - b_i^*(s_t, a_t^{-i})\right)^2\right] \tag{26}$$

4.4 SUBOPTIMALITY OF THE OPTIMAL STATE-DEPENDENT BASELINE

How much do we reduce variance over a traditional baseline that only depends on state? Using Equation (25), we show the following improvement

$$I_{b=b^*(s)} := \sum_i \mathbb{E}_{\rho_\pi, a_t^{-i}} \left[Z_i \left(b_i^*(s_t) - b_i^*(s_t, a_t^{-i}) \right)^2 \right] \tag{27}$$

$$= \sum_i \mathbb{E}_{\rho_\pi, a_t^{-i}} \left[Z_i \left(\frac{\sum_j Y_j}{\sum_j Z_j} - \frac{Y_i}{Z_i} \right)^2 \right] \tag{28}$$

$$= \sum_i \mathbb{E}_{\rho_\pi, a_t^{-i}} \left[\frac{1}{Z_i} \left(\frac{Z_i}{\sum_j Z_j} \sum_j Y_j - Y_i \right)^2 \right] \tag{29}$$

This suggests that the variance difference to be a weighted sum of the deviation of the per-component score-weighted marginalized Q (denoted Y_i) from the weighted average of all the component scored-weighted marginalized Q values. This suggests that the difference is particularly large when the Q function is highly sensitive to the actions, especially along those directions that influence the gradient the most. Our empirical results in Section 5 additionally demonstrate the benefit of action-dependent over state-only baselines.

4.5 MARGINALIZATION OF THE GLOBAL ACTION-VALUE FUNCTION

Using the previous theory, we now consider various baselines that could be used in practice, and associated computational cost.

Marginalized Q baseline Even though the optimal state-only baseline is known, it is rarely used in practice (Duan et al., 2016). Rather, for both computational and conceptual benefit, the choice of $b(s_t) = \mathbb{E}_{a_t}[Q(s_t, a_t)] = V(s_t)$ is often used. Similarly, we propose to use $b_i(s_t, a_t^{-i}) = \mathbb{E}_{a_t^i}\left[Q_{\pi_\theta}(s_t, a_t)\right]$ which is the action-dependent analogue. In particular, when log probability of each policy factor is loosely correlated with the action-value function, then the proposed baseline is close to the optimal baseline.

$$I_{b=\mathbb{E}_{a^i}[\hat{Q}(a,s)]} = \sum_i \mathbb{E}_{\rho_\pi, a_t^{-i}} \left[Z_i \left(\mathbb{E}_{a^i} \left[\hat{Q}(a, s) \right] - \frac{\mathbb{E}_{a_t^i} \left[z_i^T z_i \hat{Q}(s_t, a_t) \right]}{\mathbb{E}_{a_t^i} \left[z_i^T z_i \right]} \right)^2 \right] \approx 0 \tag{30}$$

when $\mathbb{E}_{a_t^i} \left[z_i^T z_i \hat{Q}(s_t, a_t) \right] \approx \mathbb{E}_{a_t^i} \left[z_i^T z_i \right] \mathbb{E}_{a_t^i} \left[\hat{Q}(s_t, a_t) \right]$.

This has the added benefit of only needing to learn one function approximator, for estimating $Q(s_t, a_t)$, and implicitly using it to obtain the baselines for each action coordinate.

Monte Carlo marginalized Q baseline After learning $Q_{\pi_\theta}(s_t, a_t)$ we can obtain the baselines through Monte Carlo estimates:

$$b_i(s_t, a_t^{-i}) = \frac{1}{M} \sum_{j=0}^{M} Q_{\pi_\theta}(s_t, (a_t^{-i}, a_j)) \tag{31}$$

where $a_j \sim \pi_\theta(a_t^i | s_t)$ are samples of the action coordinate i.

Mean marginalized Q baseline Though we reduced the computational burden from learning m functions to one function, the use of Monte Carlo samples can still be computationally expensive. In particular, when using deep neural networks to approximate the Q-function, forward propagation through the network can be even more computationally expensive than stepping through a fast simulator (e.g. MuJoCo). In such settings, we further propose the following more computationally practical baseline:

$$b_i(s_t, a_t^{-i}) = Q_{\pi_\theta}(s_t, (a_t^{-i}, \bar{a}^i)) \tag{32}$$

where $\bar{a}^i = \mathbb{E}_{\pi_\theta} \left[a_t^i \right]$ is the average action for coordinate i.

4.6 BASELINES FOR GENERAL ACTIONS

In the preceding derivations, we have assumed policy actions are conditionally-independent across dimensions. In the more general case, we only assume that there are m factors a_t^1 through a_t^m which altogether forms the action a_t. Conditioned on s_t, the different factors form a certain directed acyclic graphical model (including the fully dependent case). Without loss of generality, we assume that the following factorization holds:

$$\pi_\theta(a_t|s_t) = \prod_{i=1}^{m} \pi_\theta(a_t^i|s_t, a_t^{f(i)}) \tag{33}$$

where $f(i)$ denotes the indices of the parents of the ith factor. Let $D(i)$ denote the indices of descendants of i in the graphical model (including i itself). In this case, we can set the ith baseline to be $b_i(s_t, a_t^{[m]\setminus D(i)})$, where $[m] = \{1, 2, \ldots, m\}$. In other words, the ith baseline can depend on all other factors which the ith factor does not influence. The overall gradient estimator is given by

$$\nabla_\theta \eta(\pi_\theta) = \mathbb{E}_{\rho_\pi, \pi}\left[\sum_{i=1}^{m} \nabla_\theta \log \pi_\theta(a_t^i|s_t, a_t^{f(i)}) \left(\hat{Q}(s_t, a_t) - b_i(s_t, a_t^{[m]\setminus D(i)})\right)\right] \tag{34}$$

In the most general case without any conditional independence assumptions, we have $f(i) = \{1, 2, \ldots, i-1\}$, and $D(i) = \{i, i+1, \ldots, m\}$. The above equation reduces to

$$\nabla_\theta \eta(\pi_\theta) = \mathbb{E}_{\rho_\pi, \pi}\left[\sum_{i=1}^{m} \nabla_\theta \log \pi_\theta(a_t^i|s_t, a_t^1, \ldots, a_t^{i-1}) \left(\hat{Q}(s_t, a_t) - b_i(s_t, a_t^1, \ldots, a_t^{i-1})\right)\right] \tag{35}$$

The above analysis for optimal baselines and variance suboptimality transfers also to the case of general actions.

Computing action-dependent baselines for general actions The marginalization presented in Section 4.5 does not apply for the general action setting. Instead, m individual baselines can be trained according to the factorization, and each of them can be fitted from data collected from the previous iteration. In the general case, this means fitting m functions $b_i(s_t, a_t^1, \ldots, a_t^i)$, for $i \in \{1, \ldots, m\}$.

4.7 COMPATIBILITY WITH GAE

Temporal Difference (TD) learning methods such as GAE Schulman et al. (2016) allow us to smoothly interpolate between high-bias, low-variance estimates; and low-bias, high-variance estimates of the policy gradient. These methods are based on the idea of being able to predict future returns, thereby bootstrapping the learning procedure. In particular, when using the value function as baseline, we have $A(s_t, a_t) = \mathbb{E}\left[r_t + \gamma V(s_{t+1}) - V(s_t)\right] = \left[r_t + \gamma b(s_{t+1}) - b(s_t)\right]]$ if $b(s)$ is an unbiased estimator for $V(s)$. GAE proposed in Schulman et al. (2016) use an exponential averaging of such temporal difference terms over a trajectory to significantly reduce the variance of the advantage at the cost of a small bias (it allows us to pick where we want to be on the bias-variance curve). Similarly, if we use $b_i(s_t, a_t^{-i})$ as an unbiased estimator for $\mathbb{E}_{a^i} Q(s, a)$, we have:

$$\mathbb{E}_{\pi, \mathcal{M}}\left[r_t + \gamma b_i(s_{t+1}, a_{t+1}^{-i}) - b_i(s_t, a_t^{-i})\right] = Q(s_t, a_t) - \mathbb{E}[Q(s_t, a_t)] = A(s_t, a_t) \tag{36}$$

Thus, the temporal difference error with the action dependent baselines is an unbiased estimator for the advantage function as well. This allows us to use the GAE procedure to further reduce variance at the cost of a bias.

5 EXPERIMENTS AND RESULTS

Continuous Control Benchmarks Firstly, we present the results of the proposed action-dependent baselines on popular benchmark tasks. These tasks have been widely studied in the

deep reinforcement learning community (Duan et al., 2016; Gu et al., 2017; Lillicrap et al., 2016; Rajeswaran et al., 2017). The studied tasks include the hopper, half-cheetah, and ant locomotion tasks simulated in MuJoCo (Todorov et al., 2012).[1] In addition to these tasks, we also consider a door opening task with a high dimensional multi-fingered hand, to study the effectiveness of the proposed approach in high dimensional tasks. Figure 1 presents the learning curves on these tasks. We compare the action-dependent baseline with a baseline that uses only information about the states, which is the most common approach in literature. We observe that the action-dependent baselines perform consistently better.

A popular baseline parameterization choice is a linear function on a small number of non-linear features of the state Duan et al. (2016), especially for policy gradient methods. In this work, to enable a fair comparison, we use a Random Fourier Feature representation for the baseline (Rajeswaran et al., 2017). The features are constructed as: $y(x) = \sin(\frac{1}{\nu}Px + \phi)$ where P is a matrix with each element independently drawn from the standard normal distribution, ϕ is a random phase shift in $[-\pi, \pi)$ and, and ν is a bandwidth parameter. These features approximate the RKHS features under an RBF kernel. Using these features, the baseline is parameterized as $b = w^T y(x)$ where x are the appropriate inputs to the baseline, and w are trainable parameters. P and ϕ are not trained in this parameterization. Such a representation was chosen for two reasons: (a) we wish to have the same number of trainable parameters for all the baseline architectures, and not have more parameters in the action-dependent case (which has a larger number of inputs to the baseline); (b) since the final representation is linear, it is possible to accurately estimate the optimal parameters with a Newton step, thereby alleviating the results from confounding optimization issues. For the experiments, we used 250 random Fourier features. For policy optimization, we use a variant of the natural policy gradient and TRPO methods as described in Rajeswaran et al. (2017).

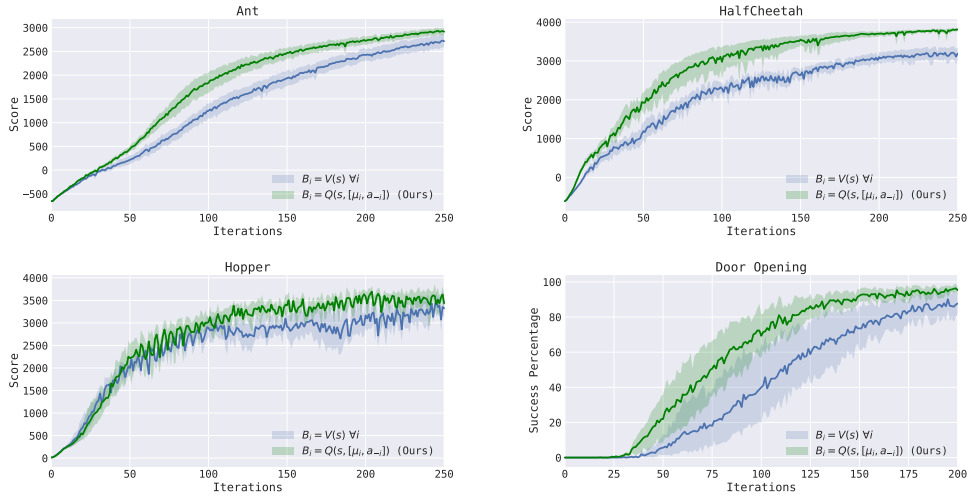


Figure 1: Comparison between value function baseline and action-conditioned baseline on various continuous control tasks. Action-conditioned baseline performs consistently better across all the tasks.

Choice of Action-Dependent Baseline Form Next, we study the influence of computing the baseline by using empirical averages sampled from the Q-function versus using the mean-action of the action-coordinate for computing the baseline (both described in 4.5). In our experiments, as shown in Figure 2 we find that the two variants perform comparably, with the latter performing slightly better towards the end of the learning process. This suggests that though sampling from the Q-function might provide a better estimate of the conditional expectation in theory, function approximation from finite samples injects errors that may degrade the quality of estimates. In particular, sub-sampling from the Q-function is likely to produce better results if the learned Q-function is accurate for a large fraction of the action space, but getting such high quality approximations might be hard in practice.

[1]We used physics parameters as recommended in Rajeswaran et al. (2017) and use the MuJoCo 1.5 simulator. Thus the reward numbers may not be consistent with numbers previously reported in literature.

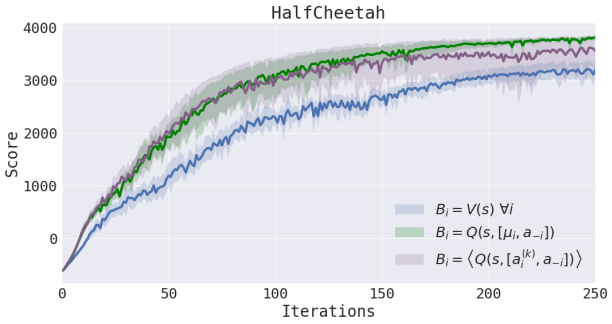


Figure 2: Variants of the action-dependent baseline that use: (i) sampling from the Q-function to estimate the conditional expectation; (ii) Using the mean action to form a linear approximation to the conditional expectation. We find that both variants perform comparably, with the latter being more computationally efficient.

Compatibility with GAE Temporal Difference (TD) based methods including GAE (Schulman et al., 2016) allow for a smooth interpolation between high-bias, low-variance estimates; and low-bias, high-variance estimates of the policy gradient. As shown in Section 4, the action-dependent baselines are consistent with TD procedures with their temporal differences being estimates of the advantage function. Our results summarized in Figure 3 suggests that slightly biasing the gradient to reduce variance produces the best results, while high-bias estimates perform poorly. Prior work with baselines that utilize global information (Foerster et al., 2017) employ the high-bias variant. The results here suggest that there is potential to further improve upon those results by carefully studying the bias-variance trade-off.

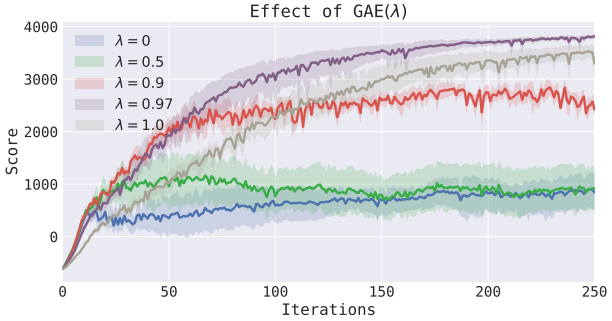
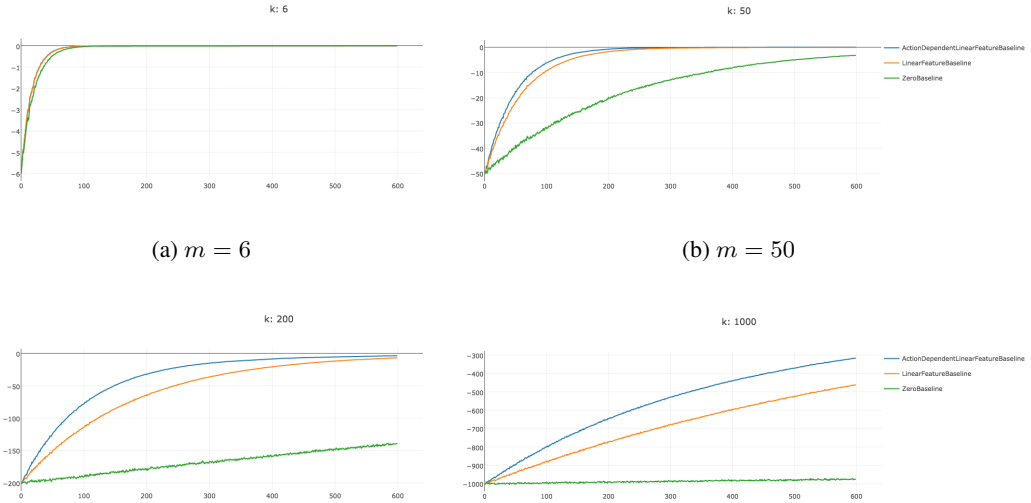


Figure 3: We study the influence of λ in GAE which allows to trade-off bias and variance as desired. High bias gradient corresponding to smaller values of λ do not make progress after a while. High variance gradient ($\lambda = 1$) has trouble learning initially. Allowing for a small bias to reduce the variance, corresponding to the intermediate $\lambda = 0.97$ produces the best overall result, consistent with the findings in Schulman et al. (2016).

High-Dimensional Action Spaces Intuitively, the benefit of the action-dependent baseline can be greater for higher dimensional problems. We show this effect on a simple synthetic example. The example is a one-step MDP comprising of a single state, $\mathcal{S} = \{0\}$ and an m-dimensional action space, $\mathcal{A} = \mathbb{R}^m$. The reward is given as the negative ℓ_2 loss of the action vector, $r(s, a) = -\|a\|_2$. The optimal action is thus to select the zero vector $a = \mathbf{0}$. The results for a demonstrative example are shown in Figure 4, which shows that the action-dependent baseline successfully improves convergent more for higher dimensional problems than lower dimensional problems. Due to the lack of state information, the linear baseline reduces to whitening the discounted returns. The action-dependent baseline, on the other hand, allows the learning algorithm to assess the advantage of each individual action dimension by utilizing information from all other action dimensions.



(a) $m = 6$

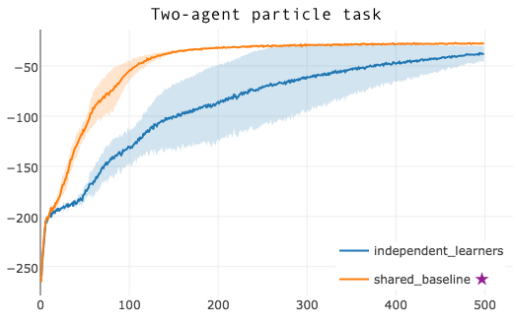
(b) $m = 50$

(c) $m = 200$

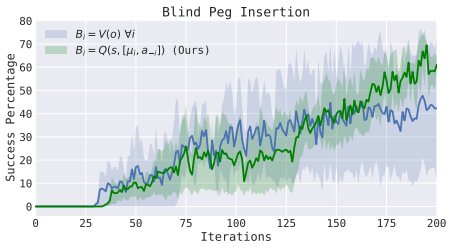
(d) $m = 1000$

Figure 4: At high dimensions, the action-dependent baseline provides considerable variance reduction for a single-state MDP, as compared to a linear feature baseline. For reference, the zero baseline (no baseline) is also shown.

Partial Observability Finally, we also consider the extension of the core idea of using global information, by studying a POMDP task and a multi-agent task. We use the blind peg-insertion task which is widely studied in robot learning literature Montgomery & Levine (2016). The task requires the robot to insert the peg into the hole (slot), but the robot is blind to the location of the hole. Thus, we expect a searching behavior to emerge from the robot, where it learns that the hole is present on the table and performs appropriate sweeping motions till it is able to find the hole. In this case, we consider a baseline that knows the location of the hole. We observe that a baseline with this additional information enables faster learning. For the multi-agent setting, we analyze a two-agent particle environment task in which the goal is for each agent to reach their goal, where their goal is known by the other agent and they have a continuous communication channel. Figure 5 shows that including the inclusion of information from other agents into the baseline improves the training performance, indicating that variance reduction may be key for multi-agent reinforcement learning.



(a) Even for a simple multi-agent particle task with two agents, using global state information (purple start) to fit the baseline results in much faster convergence.



(b) Success percentage on the blind peg insertion task. In our method, the policy still acts on the observations and does not know the hole location. However, the baseline has access to this information and helps to speed up the learning.

Figure 5: Experiments with additional information in the baseline.

6 CONCLUSION

An action-dependent baseline enables using additional signals beyond the state to achieve bias-free variance reduction. In this work, we consider both conditionally independent action spaces and general action spaces, and derive an optimal action-dependent baseline for a wide class of problems. We prodive analysis of the variance reduction improvement over non-optimal baselines, including the traditional optimal baseline that only depends on state. We additionally propose several practical action-dependent baselines which perform well on a variety of continuous control tasks and are demonstrated to give greater improvement for synthetic high-dimensional action problems. The use of additional signals beyond the local state generalizes to other problem settings, for instance in POMDP and multi-agent tasks. In future work, we propose to investigate related methods in such settings on large scale problems.

REFERENCES

Yan Duan, Xi Chen, Rein Houthooft, John Schulman, and Pieter Abbeel. Benchmarking deep reinforcement learning for continuous control. In *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, 2016.

Jakob Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual multi-agent policy gradients. *arXiv preprint arXiv:1705.08926*, 2017.

Evan Greensmith, Peter L Bartlett, and Jonathan Baxter. Variance reduction techniques for gradient estimates in reinforcement learning. *Journal of Machine Learning Research*, 5(Nov):1471–1530, 2004.

Shixiang Gu, Timothy Lillicrap, Zoubin Ghahramani, Richard E Turner, and Sergey Levine. Q-prop: Sample-efficient policy gradient with an off-policy critic. In *International Conference on Learning Representations (ICLR2017)*, 2017.

Sham M Kakade. A natural policy gradient. In *Advances in neural information processing systems*, pp. 1531–1538, 2002.

Vijay R Konda and John N Tsitsiklis. Actor-critic algorithms. In *Advances in neural information processing systems*, pp. 1008–1014, 2000.

S. Levine and V. Koltun. Guided policy search. In *ICML*, 2013.

Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. End-to-end training of deep visuomotor policies. *Journal of Machine Learning Research*, 17(39):1–40, 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. In *International Conference on Learning Representations (ICLR2016)*, 2016.

Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. *arXiv preprint arXiv:1706.02275*, 2017.

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):529–533, 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.

W. Montgomery and S. Levine. Guided policy search as approximate mirror descent. In *NIPS*, 2016.

I. Mordatch, K. Lowrey, G. Andrew, Z. Popovic, and E. Todorov. Interactive Control of Diverse Complex Characters with Neural Networks. In *NIPS*, 2015.

Jan Peters and Stefan Schaal. Natural actor-critic. *Neurocomputing*, 71(7):1180–1190, 2008.

A. Rajeswaran and J. Schulman E. Todorov S. Levine V. Kumar, A. Gupta. Learning complex dexterous manipulation with deep reinforcement learning and demonstrations. *ArXiv e-prints*, 2017.

A. Rajeswaran, K. Lowrey, E. Todorov, and S. Kakade. Towards generalization and simplicity in continuous control. *ArXiv e-prints*, 2017.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pp. 1889–1897, 2015.

John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. In *International Conference on Learning Representations (ICLR2016)*, 2016.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.

Richard S Sutton, David A McAllester, Satinder P Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Advances in neural information processing systems*, pp. 1057–1063, 2000.

Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *IROS*, pp. 5026–5033. IEEE, 2012. ISBN 978-1-4673-1737-5. URL `http://dblp.uni-trier.de/db/conf/iros/iros2012.html#TodorovET12`.

Christopher JCH Watkins and Peter Dayan. Q-learning. *Machine learning*, 8(3-4):279–292, 1992.

Lex Weaver and Nigel Tao. The optimal reward baseline for gradient-based reinforcement learning. In *Proceedings of the Seventeenth conference on Uncertainty in artificial intelligence*, pp. 538–545. Morgan Kaufmann Publishers Inc., 2001.

Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992.