Estimation Error Correction in Deep Reinforcement Learning for Deterministic Actor-Critic Methods

Baturay Saglam, Enes Duran, Dogan C. Cicek, Furkan B. Mutlu
and Suleyman S. Kozat, *Senior Member, IEEE*

Department of Electrical and Electronics Engineering, Bilkent University, Ankara, Turkey
{baturay, enesd, cicek, kozat}@ee.bilkent.edu.tr
{burak.mutlu}@bilkent.edu.tr

***Abstract*—In value-based deep reinforcement learning methods, approximation of value functions induces overestimation bias and leads to suboptimal policies. We show that in deep actor-critic methods that aim to overcome the overestimation bias, if the reinforcement signals received by the agent have a high variance, a significant underestimation bias arises. To minimize the underestimation, we introduce a parameter-free, novel deep Q-learning variant. Our Q-value update rule combines the notions behind Clipped Double Q-learning and Maxmin Q-learning by computing the critic objective through the nested combination of maximum and minimum operators to bound the approximate value estimates. We evaluate our modification on the suite of several OpenAI Gym continuous control tasks, improving the state-of-the-art in every environment tested.**

***Index Terms*—Deep reinforcement learning, deterministic actor-critic methods, estimation bias**

I. Introduction

A. Preliminaries

In recent years, utilization of deep approaches to approximate the policies of reinforcement learning (RL) agents achieved numerous successes in wide range of applications such as playing Atari games [1], autonomous driving [2], path planning [3], playing board games like chess, shogi [4], go [5] and even beating human players on StarCraft [6] [7]. Nevertheless, there are several issues regarding the function approximation in the deep reinforcement learning setting [8]. One of the problems resulting from the function approximation is the systematic estimation bias that prevents the learning agent from reaching the maximum performance and deep methods to be applied to the real-world problems [9], [10]. The estimation bias on value estimates in value-based RL algorithms has been studied for discrete action spaces [11]–[15]. Furthermore, similar work in the continuous control domain for actor-critic methods is done for a subcategory of estimation bias, that is, overestimation bias [8]. This paper shows that in deterministic actor-critic methods that aim to overcome the accumulated overestimation bias and high variance, there exists an underestimation bias on the value estimates [16], [17]. Our work addresses this issue from a probabilistic point of view and improves the current state-of-the-art performance on several continuous control RL tasks.

The estimation bias on the action value estimates in value-based deep reinforcement learning is usually studied in two categories: underestimation and overestimation. Overestimation bias, resulting from the maximization of noisy estimates in the standard Q-learning, induces an accumulated error through the learning stage [18]. In a function approximation setting, such an estimation noise is unavoidable as deep neural networks represent the action-value functions [8]. This inaccuracy in the action values is further exaggerated due to the temporal difference learning [9]. On the other hand, underestimation bias is an outcome of Q-learning variants that aim to eliminate the accumulated overestimation bias and high variance on the value estimates [14], [8]. Even though the standard deep Q-learning modifications are shown to overcome the overestimation bias and high variance build-up, the existence of underestimated state-action values can still degrade the performance of an RL agent by assigning low values to "good" state-action pairs, causing suboptimal and divergent behavior.

We begin by establishing that this underestimation phenomenon is present in delayed policy gradient that utilizes a pair of critics, namely, Clipped Double Q-learning [8], in the continuous actor-critic settings. During training, the minimum of the estimates by two critics is used to construct the target Q-value in the temporal difference learning [9]. Unfortunately, taking the minimum of Q-values in learning the targets produces a consistent underestimation of Q-value estimates despite the decoupled actor and critics. To address this problem, we first show that variance of the reinforcement signals that the agent encounters during the learning phase increases the underestimation bias. Then, to overcome

2375-0197/21/$31.00 ©2021 IEEE
DOI 10.1109/ICTAI52525.2021.00027

underestimation bias and improve the performance of the deterministic actor-critic methods, we introduce a deep Q-learning variant that combines the ideas behind Clipped Double Q-learning [8], and Maxmin Q-learning [14], Triplet Critic Update. Our approach leverages the notion that along with two value estimators, the value estimate of an additional estimator can be used to construct an approximate upper and lower bound to the value estimate.

We build our modification on the state-of-the-art deterministic actor-critic algorithm, Twin Delayed Deep Deterministic Policy Gradient (TD3) [8], and introduce the Triplet Critic Delayed Deep Deterministic Policy Gradient (TCD3) algorithm. Our deterministic actor-critic algorithm reduces the underestimation bias to a negligible margin and prevents the accumulated overestimation bias and high variance. We evaluate our algorithm on 12 continuous control tasks from OpenAI Gym [19], where we show that Triplet Critic Update significantly improves the performance of TD3 [8]. For the reproducibility concerns, we run our experiments across a large set of seeds for the sake of a fair evaluation procedure.

B. Prior Art and Comparisons

In reinforcement learning, prior works on the function approximation error in terms of the estimation bias and high variance build-up have been studied by [20] and [21]. Our work focuses on one of the outcomes of the function approximation error, namely, the underestimation bias on the action value estimates.

Several approaches reduce the effects of overestimation bias resulting from the function approximation and policy optimization in deep Q-learning. One of the successor works to the deep Q-learning [1], Double Q-learning [11], proposed by [12] and [11], employs two independent action value estimators to obtain unbiased estimates of Q-values. [14] modified the Double Q-learning [11] by utilizing multiple value estimators. This approach, Maxmin Q-learning [14], considers randomly selected value estimates of multiple critics, and a minimum of which is used in learning the Q-value target. Furthermore, methods that employ multi-step returns offer a trade-off between the variance build-up and accumulated estimation bias. Such methods are proven to be effective through distributed methods [22], [23], approximate lower and upper bounds [24], and importance sampling [13], [25]. Another trade-off in these approaches is using a longer horizon rather than finding a direct solution to the accumulated error. [26], on the other hand, offers a solution to diminish the discount factor for reducing the contribution of each erroneous estimate.

The concern with the direct solution to the accumulated approximation error has been overcome by [8] and [27] for actor-critic settings. [27] uses modified value function estimator which yields a better and lower variance gradient estimator. [8] shows the existence of overestimation bias and accumulated variance induced by the deep function approximation of Q-values. An extension of the Deep Deterministic Policy Gradient algorithm [28], Twin Delayed Deep Deterministic Policy Gradient [8] which our method is built on, introduces a direct solution to the problems of the function approximation error through the utilization of two critics, delayed actor updates and target policy smoothing regularization. TD3 [8] is shown to produce state-of-the-art results by a large margin with a sufficient number of training iterations. Although the improvements introduced by [8] can tame the accumulated error, the usage of two critics induces an underestimation bias on Q-value estimates. To overcome the underestimation error, [16] approaches the problem with a modification on the update rule for the critics in the TD3 [8]. In their approach, the target for the temporal difference learning [9] is computed through the weighted linear combination of the minimum and average of the two critics, Weighted Deep Deterministic Policy Gradient (WD3) [16]. [17] extends the WD3 [16] by increasing the number of Q-networks to three. Both studies have shown that the weighted linear combinations of Q-networks can obtain more accurate action value estimates and higher evaluation returns.

C. Contributions

Our contributions are as follows:

1) We show that if the agent receives reinforcement signals throughout the learning phase vary on a large scale, the estimation bias on the Q-value estimates severely increases.
2) We demonstrate by empirical results that for the cases in which the overestimation and variance of the value estimates are eliminated, the underestimation bias can still degrade the performance of the RL agent in terms of the evaluation returns and convergence speed.
3) We introduce a Double Q-learning [11] variant that upper and lower bounds the value estimates without any introduction of a hyper-parameter. In this way, we reduce underestimating to a negligible level and keep the overestimation and high variance build-up removed.
4) Through an extensive set of experiments, we show that our method improves the convergence speed and performance of the state-of-the-art on 12 challenging OpenAI Gym [19] continuous control tasks.

II. Background

Reinforcement learning paradigm considers an agent interacting with its environment to learn the optimal, reward-maximizing behavior. The standard reinforcement learning can be formalized by a Markov Decision Process (MDP) defined by the tuple $(\mathcal{S}, \mathcal{A}, p_M, r, \gamma)$ that involves a state space $\mathcal{S}$, an action space $\mathcal{A}$, and transition dynamics $p_M(s'|s,a)$ for the MDP in interest denoted by M. At each discrete time step t, given an observed state $s \in \mathcal{S}$, the agent chooses an action $a \in \mathcal{A}$ with respect to its policy $\pi : \mathcal{S} \rightarrow \mathcal{A}$ which can be either stochastic or deterministic, and receives a reward r from a reward function $\mathcal{R} : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$, and then observes a next state s'. The cumulative reward which the agent tries to maximize is defined as the discounted sum of rewards $R_t = \sum_{i=t}^{T} \gamma^{i-t} r(s_i, a_i)$ where discount factor $\gamma \in [0, 1)$ is scaling long-term rewards such that short-term rewards can be prioritized more.

Reinforcement learning aims to obtain the optimal policy π_ϕ^* parameterized by ϕ that maximizes the expected return $J(\phi) = \mathbb{E}_{s_i \sim p_\pi, a_i \sim \pi}[R_0]$. In continuous control domain, parametrized policies which are usually approximated by deep neural networks, can be updated by computing the gradient of the expected return $\nabla_\phi J(\phi)$. In an actor-critic setting, the policy π, also referred as the actor, can be updated via deterministic policy gradient (DPG) algorithm [29]:

$$\nabla_\phi J(\phi) = \mathbb{E}_{s \sim p_\pi}[\nabla_a Q^\pi(s,a)|_{a=\pi(s)} \nabla_\phi \pi_\phi(s)]. \quad (1)$$

The expected return when taking action a after observing the state s while following the policy π, $Q^\pi(s,a) = \mathbb{E}_{s_i \sim p_\pi, a_i \sim \pi}[R_t|s,a]$, is also called the critic or action-value function of the agent that values the quality of a state-action pair. The action-value function or the critic is used to evaluate a learning agent's current policy and improve the policy to obtain higher quality choices of actions, i.e., a better policy.

In Q-learning, when the transition probability of an environment is known, the state-value function, Q^π can be estimated recursively through Bellman equation [30] given the transition tuple (s, a, r, s'):

$$Q^\pi(s,a) = r + \gamma \mathbb{E}_{s', a'}[Q^\pi(s', a')], \quad a' \sim \pi(s'). \quad (2)$$

For a large state space, the action value can be estimated by a function approximator $Q_\theta(s,a)$ parameterized by θ. In the deep setting of Q-learning [1], the critic network is updated through the temporal difference learning by a secondary frozen network $Q'_\theta(s,a)$, also known as the target network. In this way, a fixed target y can be achieved to update the critic network:

$$y = r + \gamma Q_{\theta'}(s', a'), \quad a' \sim \pi_{\phi'}(s'), \quad (3)$$

where the actions for the next state are chosen from a separate target actor-network $\pi_{\phi'}$ in the actor-critic setting for continuous action spaces. The target networks are either updated by a small proportion τ at each time step, $\theta' \leftarrow \tau\theta + (1-\tau)\theta'$, called soft-update, or periodically to exactly match the current networks. Such an update rule is orthogonal to any deep reinforcement learning method that utilizes a separate target network. For instance, it can be applied to off-policy methods that sample mini-batches of transition tuples from the experience replay buffer [31] for the update.

III. The Underestimation in Deterministic Actor-Critic Methods

In discrete action domains, overestimation due to the analytical maximization of action values is an evident and widely studied artifact [11]–[15]. For actor-critic settings, the existence and effects of the overestimation have been proven by [8] through the policy updates via gradient descent. In their approach, the minimum value of these critics is used to compute the target action value at each iteration along with delayed actor updates and smoothed target policy value. However, such utilization of the minimum operator to overcome the overestimation of the Q-values may introduce an underestimation bias [8], [16], [17]. We begin by proving through basic assumptions and statements that the underestimation phenomenon occurs in continuous control, actor-critic methods in environments with different reinforcement signals. Then, we introduce our modified target Q-value update rule to reduce the underestimation bias while staying in the "safe zone" of function approximation error in the actor-critic setting.

In the TD3 algorithm [8], the policy is updated with respect to the value estimates of one of two approximate critics. Without loss of generality, we assume that the policy is updated with respect to the first approximate critic, which is denoted by $Q_{\theta_1}(s,a)$ through the deterministic policy gradient. We show that in an environment with rewards that vary on a large scale, the target update rule for the Q-values induces an underestimation bias.

Let ϕ_{approx} define the parameters from the actor update by the maximization of the first approximate critic $Q_{\theta_1}(s,a)$:

$$\phi_{\text{approx}} = \phi + \frac{\alpha}{Z_1} \mathbb{E}_{s \sim p_\pi}[\nabla_\phi \pi_\phi(s) \nabla_a Q_{\theta_1}(s,a)|_{a=\pi_\phi(s)}], \quad (4)$$

where Z_1 is the gradient normalizing term such that $Z^{-1}||\mathbb{E}[\cdot]|| = 1$. As the actor is optimized with respect to $Q_{\theta_1}(s,a)$ and the gradient direction is a local maximizer, there exists ξ sufficiently small such that if $\alpha < \xi$, then the *approximate* value of π_{approx} by the first critic will be bounded below by the *approximate* value of π_{approx} by the second critic:

$$\mathbb{E}[Q_{\theta_1}(s, \pi_{\text{approx}}(s))] \geq \mathbb{E}[Q_{\theta_2}(s, \pi_{\text{approx}}(s))]. \quad (5)$$

Note that in the latter equation, the estimated action values by both critics are overestimated. Then, we can treat the function approximation error as a Gaussian random variable:

$$\begin{aligned} Q_{\theta_1}(s,a) - Q^*(s,a) = G_1 \sim \mathcal{N}(\epsilon_1, \sigma_1), \\ Q_{\theta_2}(s,a) - Q^*(s,a) = G_2 \sim \mathcal{N}(\epsilon_2, \sigma_2). \end{aligned} \tag{6}$$

By (5) and (6), we have $\epsilon_1 \geq \epsilon_2 \geq 0$. Moreover, as the presence of the delayed actor updates, the mean function approximation errors by both critics are not very distant due to the decoupling the actor and first critic, i.e., $\epsilon_1 - \epsilon_2 \approx 0$. Then, by the first moments of the minimum of two correlated Gaussian random variables [32], the expected function approximation error for the Clipped Double Q-Learning algorithm [8] becomes:

$$\mathbb{E}[\min_{i=1,2}\{G_i\}] = \frac{\epsilon_1 + \epsilon_2}{2} - \frac{\theta}{\sqrt{2\pi}}, \tag{7}$$

where $\theta \coloneqq \sqrt{\sigma_1^2 + \sigma_2^2 - 2\rho\sigma_1\sigma_2}$ and ρ is the correlation coefficient between the error distributions, G_1 and G_2. The error Gaussian's are correlated since the critics are not entirely independent due to the use of the same experience replay buffer and opposite critics in learning the approximate targets [8]. If $\sigma_1, \sigma_2 > \sqrt{\frac{\pi}{1-\rho}}\epsilon_1$, then the action value estimate will be underestimated:

$$\mathbb{E}[\min_{i=1,2}\{Q_{\theta_i}(s,a)\} - Q^*(s,a)] < 0. \tag{8}$$

It can be observed from the underestimation condition that for a highly correlated pair of critics, underestimation does not exist. However, because of the delayed policy updates, the correlation between the critics is regularized [8]. Thus, a weak or moderate correlation between the pair of critics is expected [8], which increases the possibility of underestimation.

Although delayed policy updates and the minimization of the value estimates aim to reduce error growth, the variances of the value estimates are not eliminated since they are proportional to the variance of the future rewards and estimation errors [8]. In function approximation setting, the Bellman equation is never exactly satisfied, yielding erroneous value estimates as a function of the true TD-error [10], as expressed by (6). Then, it can be shown that the variance of the value estimates overgrow as the agent interacts with the environment and observes varying reward signals [33]. We can express the approximate Q-values in terms of the expected value of the discounted cumulative rewards, as shown in [8]:

$$Q_{\theta_i}(s,a) = \mathbb{E}_{s_i \sim p_\pi, a_i \sim \pi}[\sum_{i=t}^{T} \gamma^{i-t} r_i] + \epsilon_i \sum_{i=t}^{T} \gamma^{i-t}. \tag{9}$$

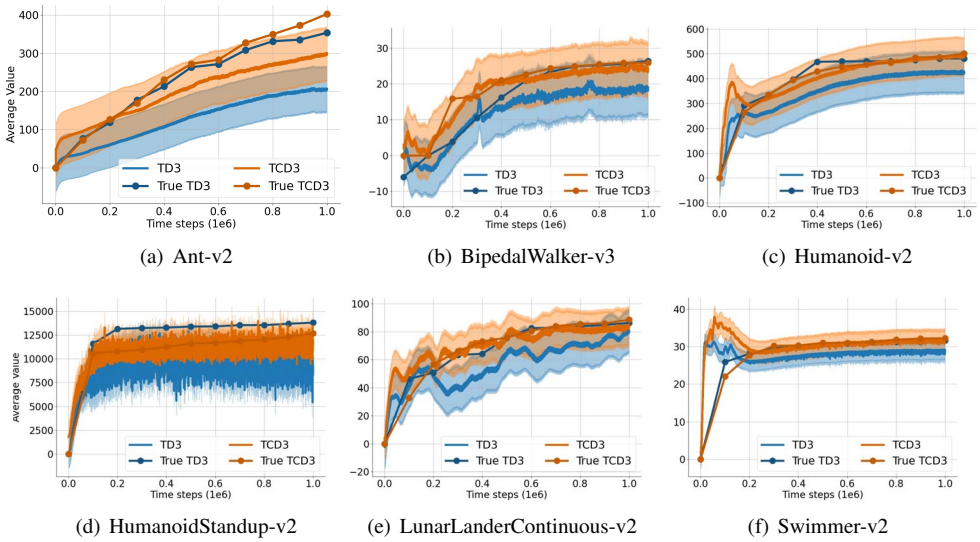
Suppose the expected value of the estimation error is constant for both critics. In that case, varying reinforcement signals increase the variance of the value estimates, which results in an underestimation error. The agent extensively explores the environment, a mandatory requirement for continuous action domain [10], the variance of the reward signals that the agent encounters start growing, and underestimation bias will become inevitable. Furthermore, the minimum operator eliminates the accumulating error due to the temporal difference learning, and thus, underestimation is far more preferable to overestimation bias in actor-critic setting [8]. Nevertheless, underestimating a value estimate may discourage the agent from choosing good state-action pairs for an extended period and reinforce the agent to value suboptimal state-action pairs more.

A. Does the theoretical underestimation due to the minimum operator occur in practice?

This question can be answered by observing value estimates produced by the target Q-value update with the minimum of two critics over a training duration along with the actual Q-values while the agent is learning on a set of OpenAI Gym [19] continuous control tasks. The true Q-values are estimated through the average discounted sum of rewards over randomly initialized 1000 episodes following the current policy. Estimated Q-values are also computed using the average Q-value estimates by the critics on the 1000 sampled episodes.

In Fig. 1, there exists an apparent underestimation bias that occurs on the learning procedure. The underestimation bias keeps growing or follows a pattern parallel with the actual values depending on the environment. These empirical findings verify our claims; that is, the approximate critics start by overestimating the actual Q-values. However, after a while in which the agent starts an extensive exploration procedure, the variances of the approximate critics increase, and an unavoidable underestimation arises. Moreover, for some environments such as Ant, as the actual state-action values increase due to the varying rewards, the variances of the approximate critics increase, and the underestimation on the value estimates keeps growing even with the delayed target and actor updates. Whereas, when the true Q-values reach a steady-state, the underestimation settles to a certain level which can also be observed from Humanoid, Swimmer, and BipedalWalker environments. Even though in the set of OpenAI Gym [19] tasks, the continuous state and multi-dimensional action spaces are contributing factors to the growth in the variance of the action-values, the scale of these tasks is still tiny compared to the real-world settings [10]. Therefore, underestimating value estimates would be a more detrimental and inevitable problem for tasks with larger-scaled state and action spaces.

To remedy the observed underestimation bias, we next introduce our novel modification on the target Q-



(a) Ant-v2 (b) BipedalWalker-v3 (c) Humanoid-v2

(d) HumanoidStandup-v2 (e) LunarLanderContinuous-v2 (f) Swimmer-v2

Fig. 1. Measuring estimation bias of fine-tuned TD3 versus TCD3 while learning on MuJoCo and Box2D environments over 1 million time steps. Estimated and true Q-values are computed through Monte Carlo simulation for 1000 samples.

value update, Triplet Critic Update, which significantly reduces the underestimation bias by the minimum of two approximate critics.

IV. Triplet Critic Update

Several approaches have been proposed to overcome the underestimation bias introduced by the minimum of two critics. These approaches differ in terms of the treatments of the function approximation error as random variables. Although these proposals can overcome the underestimation bias introduced by the target Q-value update, they introduce additional hyper-parameters to be tuned. This section presents our parameter-free, novel modification on the target update rule for deterministic actor-critic methods for continuous control, which is also a variant of Double Q-learning [11] but with three critics.

In the current approach to overcome the overestimated action values, we highlight that the bias introduced by the approximate critics is a function of the expected function approximation error by the critics and variance of the value estimates. As the expected function approximation error is assumed to be constant for both critics, depending on the variance of the value estimates, which is proportional to the encountered reinforcement signals, the function approximation error can either be an underestimation or overestimation. In practice, however, we observe that the variance of the value estimates is much greater than the expected function approximation error as the continuous control tasks require a comprehensive exploration procedure. Hence, an underestimation error occurs caused by the objective computation for the action value functions that employ the minimum operator as in TD3 [8]. As a result, the critic does not value good state-action pairs as much as they should be. To address this problem that has been approached from different perspectives, we propose to simply upper and lower-bound the biased value estimates of three critics without introducing any additional hyper-parameter. This results in taking the minimum of the maximum of two critics and a single distinct critic, which gives the modified target update rule, Triplet Critic Update:

$$y = r + \gamma \min \left(\max_{i=1,2} (Q_{\theta'_i}(s', \pi_{\phi'}(s')), Q_{\theta'_3}(s', \pi_{\phi'}(s')) \right). \tag{10}$$

As the actor is optimized with respect to the first critic, the second and third critics can be represented by the same probability distribution, i.e., $G_3 \sim \mathcal{N}(\epsilon_2, \sigma_2)$. Under the same assumptions presented in the previous section, the expected function approximation error for the Triplet Critic Update can be computed by the extensions of [32] and [34]:

$$\begin{aligned} \mathbb{E}[\min(\max(G_1, G_2), G_3)] &= (\epsilon_1 + 3\epsilon_2)/4 - \theta/2\sqrt{2\pi}, \\ &= \frac{(\mathbb{E}[\min_{i=1,2}\{G_i\}] + \epsilon_2)}{2}. \end{aligned} \tag{11}$$

This expected error value is the average of the biases introduced by the minimum of two target action values as in TD3 [8], and an approximate critic as in DDPG [28]. If the overestimation is relatively larger than the underestimation, there will be a slight overestimation by this update rule, and vice versa for a relatively larger underestimation. Regardless, such slight estimation errors can be tolerated by the agent compared to TD3 [8] and

DDPG [28]. However, in practice, the variance of the value estimates is usually large, and hence, the Triplet Critic Update will underestimate the value estimates. Nonetheless, the bias will be slightly larger than half of the bias in TD3 [8] which will greatly reduce the underestimation and obtain more accurate value estimates without introducing additional hyper-parameter to be tuned. In addition, as the underestimation is preferable to the overestimation due to non-accumulating error characteristics, [8], the Triplet Critic Update remains in the "safe zone" of the function approximation in general, by keeping the overestimation and accumulated variance eliminated. Finally, it is observable in our simulations that for the extreme cases in which the variance of the value estimates is very large or very small, the Triplet Critic Update offers more accurate estimates than TD3 [8] for both ends.

Algorithm 1 TripletCriticUpdate

Input $Q_{\theta'_1}, Q_{\theta'_2}, Q_{\theta'_3}$, s', $\tilde{a}$
$y \leftarrow r + \gamma \min(\max(Q_{\theta'_1}(s', \tilde{a}), Q_{\theta'_2}(s', \tilde{a})), Q_{\theta'_3}(s', \tilde{a}))$
return y

Algorithm 2 TCD3

Initialize critic networks $Q_{\theta_1}, Q_{\theta_2}, Q_{\theta_3}$, and actor network π_ϕ with randomly initialized parameters $\theta_1, \theta_2, \theta_3, \phi$
Initialize target networks $\theta'_i \leftarrow \theta_i, \phi' \leftarrow \phi$
Initialize replay buffer $\mathcal{B}$
for $t = 1$ **to** T **do**
  Select action with exploration noise $a \sim \pi_\phi(s) + \eta$, $\eta \sim \mathcal{N}(0, \sigma)$ and observe reward r and new state s'
  Store transition tuple (s, a, r, s') in $\mathcal{B}$
  Sample mini-batch of N transitions (s, a, r, s') from $\mathcal{B}$
  $\tilde{a} \leftarrow \pi_{\phi'}(s') + \eta; \eta \sim \text{clip}(\mathcal{N}(0, \tilde{\sigma}), -c, c)$
  $y \leftarrow$ **TripletCriticUpdate**$(Q_{\theta'_1}, Q_{\theta'_2}, Q_{\theta'_3}, s', \tilde{a})$
  Update critics $\theta_i \leftarrow \text{argmin}_{\theta_i} N^{-1} \sum (y - Q_{\theta_i}(s, a))^2$
  if t mod d **then**
    Update ϕ by the deterministic policy gradient:
    $\nabla_\phi J(\phi) = N^{-1} \sum \nabla_a Q_{\theta_1}(s, a)|_{a=\pi_\phi(s)} \nabla_\phi \pi_\phi(s)$
    Update target networks:
    $\theta'_i \leftarrow \tau\theta_i + (1 - \tau)\theta'_i$
    $\phi' \leftarrow \tau\phi + (1 - \tau)\phi'$
  end if
end for

In the implementation, for the computational efficiency, the actor again should be optimized with respect to the first critic, and the utilization of more than three critics should be avoided. Therefore, the order of the critics in the nested Triplet Critic Update must remain constant. Otherwise, instability may occur if the critic with respect to which the actor is optimized and the order of critics in the update rule is altered throughout the learning phase. However, the utilization of three critics introduces additional computational complexity. If the clipped Double Q-Learning algorithm [8] is assumed to have a runtime of $\mathcal{O}(n)$, then our approach would run on $2\mathcal{O}(n)$ which increases linearly. Thus, the runtime can be reduced to $2\mathcal{O}(n) \approx \mathcal{O}(n)$ and neglected if the number of training time steps is not very large. Consequently, our modification comprehensively considers computational complexity and estimation error accuracy trade-off by introducing only a single additional critic without any hyper-parameter that requires further tuning.

We now introduce our approach built on the TD3 algorithm [8] changing the target Q-value update. Our algorithm called Triplet Critic Deep Deterministic Policy Gradient (TCD3) is summarized in Algorithm 2. In the next section, we present the experimental results for our algorithm in terms of both the Q-value estimation comparisons with TD3 [8], and evaluations on several OpenAI Gym [19] continuous control environments.

V. Experiments

To evaluate our estimation error correction approach, we first demonstrate the estimated and actual Q-value curves for TCD3, and state-of-the-art off-policy continuous control algorithm TD3 on both MuJoCo [35], and Box2D [36] continuous control tasks interfaced by OpenAI Gym [19]. Subsequently, we evaluate and compare the learning RL agents under TCD3 and TD3 algorithms on the extended set of OpenAI Gym [19] control tasks. To provide reproducibility and allow a fair comparison, we directly use the same set of tasks from [19] with no modifications to the environment dynamics.

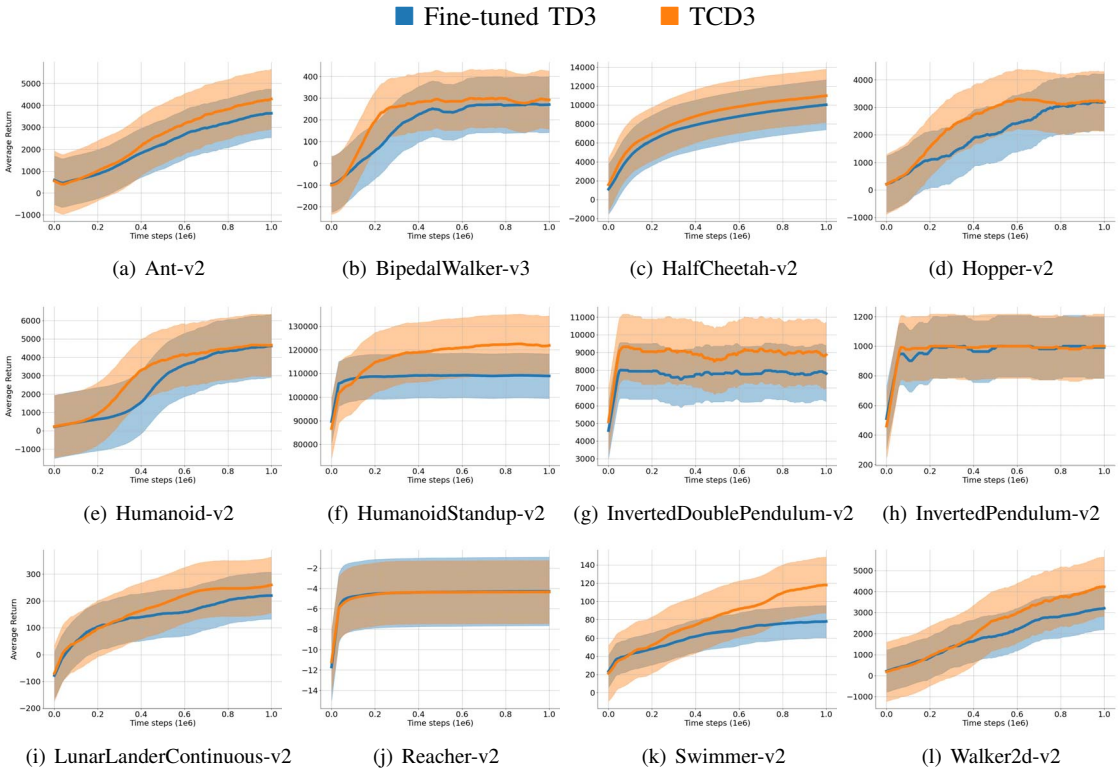
A. Implementation Details

For the implementation of TD3 [8], we utilize the open-source implementation from the author's GitHub (https://github.com/sfujim/TD3). The TD3 [8] implementation that we use is the most recent and fine-tuned version of the algorithm updated by the author as of April 2021. Our modification is built on top of the TD3 [8] implementation such that the number of critics and target Q-value computation are replaced.

B. Experimental Setup

We perform true and estimated Q-value comparisons for our approach and TD3 [8] over 6 OpenAI Gym [19] continuous control tasks are presented in Fig. 1. Each task is run for 1 million time steps, and curves are derived through the same procedure as explained in section III.

Our evaluations on each task are performed by re-running both algorithms over 1 million time steps and evaluating the performance of the agent every 5000 time steps without exploration noise. Each evaluation report is the average of 10 episode rewards on the distinct evaluation environment. The results are reported over 10 random seeds of the Gym [19] simulator, network initialization, and code dependencies.



(a) Ant-v2 (b) BipedalWalker-v3 (c) HalfCheetah-v2 (d) Hopper-v2

(e) Humanoid-v2 (f) HumanoidStandup-v2 (g) InvertedDoublePendulum-v2 (h) InvertedPendulum-v2

(i) LunarLanderContinuous-v2 (j) Reacher-v2 (k) Swimmer-v2 (l) Walker2d-v2

Fig. 2. Learning curves for the set of OpenAI Gym continuous control tasks. The shaded region represents half a standard deviation of the average evaluation over ten trials. Curves are smoothed uniformly with a sliding window of 20 for visual clarity.

C. Discussion

TCD3 obtains more accurate Q-value estimations than TD3 [8] on all environments. We observe two cases from our empirical results. First, our approach accurately estimates the Q-values by a negligible margin. Second, albeit the TCD3 greatly reduces the estimation bias, there still exists an underestimation error which is slightly bigger than half of the underestimation in TD3 [8]. These findings support our claim that an increasing variance of the reward signals encountered by the agent increases the underestimation, for example, in the Ant environment. Nonetheless, the expected value of the bias can be further reduced, as shown in this paper.

The evaluation results on the same set of tasks are depicted by Fig. 2. Our algorithm either outperforms or matches the TD3 algorithm [8] in terms of both learning speed and final performance. We observe that the estimation error prevents the agent from reaching higher possible reward potentials and stable returns. On top of the improvement that the minimum of two approximate critics can eliminate the overestimation bias, our approach obtains higher and smoother evaluation returns by reducing underestimation to a negligible margin and keeping the overestimation eliminated.

These simulation results also demonstrate that the underestimation induced by the minimum of two critics can cause "good" state-action values to be assigned low values, resulting in slower convergence and suboptimal action choices to be selected more frequently [10]. For example, from Fig. 1 and 2, we observe that for BipedalWalker and Humanoid environments, elimination of estimation bias yields faster convergence to the optimal evaluation returns. Overall, by significantly reducing this underestimation phenomenon for deterministic continuous control actor-critic methods, we show in this paper that agents can attain higher evaluation returns in fewer time steps with no estimation bias.

VI. Conclusion

Accumulated overestimation bias induced by the function approximation in deep reinforcement learning has been identified as a substantial issue. On the contrary, in deterministic actor-critic settings, techniques to overcome function approximation error build-up may lead to underestimation bias which has been a problematic drawback. In this work, we first show that encountering different reward signals increases the underestimation bias. Then, we develop a novel variant of deep Q-learning that significantly reduces the underestimation bias to a negligible level. Taken our claim and empirical results together, this improvement defines our efficient, parameter-free update rule, Triplet Critic Up-

date, which dramatically improves both the learning speed and performance of TD3 in several challenging continuous control tasks. Our algorithm, Triplet Critic Delayed Deep Deterministic Policy Gradient (TCD3), outperforms state-of-the-art by obtaining more accurate Q-value estimates. Our modification is orthogonal and can easily be adapted to any deterministic actor-critic method that employs temporal difference learning.

REFERENCES

[1] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. A. Riedmiller, "Playing atari with deep reinforcement learning," *CoRR*, vol. abs/1312.5602, 2013. [Online]. Available: http://arxiv.org/abs/1312.5602

[2] D. Isele and A. Cosgun, "Selective experience replay for lifelong learning," *CoRR*, vol. abs/1802.10269, 2018. [Online]. Available: http://arxiv.org/abs/1802.10269

[3] H. L. Chiang, J. Hsu, M. Fiser, L. Tapia, and A. Faust, "RL-RRT: kinodynamic motion planning via learning reachability estimators from RL policies," *CoRR*, vol. abs/1907.04799, 2019. [Online]. Available: http://arxiv.org/abs/1907.04799

[4] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel, T. Lillicrap, K. Simonyan, and D. Hassabis, "A general reinforcement learning algorithm that masters chess, shogi, and go through self-play," *Science*, vol. 362, pp. 1140–1144, 12 2018.

[5] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, Y. Chen, T. Lillicrap, F. Hui, L. Sifre, G. Driessche, T. Graepel, and D. Hassabis, "Mastering the game of go without human knowledge," *Nature*, vol. 550, pp. 354–359, 10 2017.

[6] O. Vinyals, I. Babuschkin, W. Czarnecki, M. Mathieu, A. Dudzik, J. Chung, D. Choi, R. Powell, T. Ewalds, P. Georgiev, J. Oh, D. Horgan, M. Kroiss, I. Danihelka, A. Huang, L. Sifre, T. Cai, J. Agapiou, M. Jaderberg, and D. Silver, "Grandmaster level in starcraft ii using multi-agent reinforcement learning," *Nature*, vol. 575, 11 2019.

[7] OpenAI, "Openai five," https://blog.openai.com/openai-five/, 2018.

[8] S. Fujimoto, H. van Hoof, and D. Meger, "Addressing function approximation error in actor-critic methods," *CoRR*, vol. abs/1802.09477, 2018. [Online]. Available: http://arxiv.org/abs/1802.09477

[9] R. Sutton, "Learning to predict by the method of temporal differences," *Machine Learning*, vol. 3, pp. 9–44, 08 1988.

[10] S. Fujimoto, D. Meger, and D. Precup, "Off-policy deep reinforcement learning without exploration," *CoRR*, vol. abs/1812.02900, 2018. [Online]. Available: http://arxiv.org/abs/1812.02900

[11] H. van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double q-learning," *CoRR*, vol. abs/1509.06461, 2015. [Online]. Available: http://arxiv.org/abs/1509.06461

[12] H. P. van Hasselt, "Insights in reinforcement learning : formal analysis and empirical evaluation of temporal-difference learning algorithms," in *Insights in reinforcement rearning : formal analysis and empirical evaluation of temporal-difference learning algorithms*, 2011.

[13] D. Precup, R. Sutton, and S. Dasgupta, "Off-policy temporal-difference learning with function approximation," *Proceedings of the 18th International Conference on Machine Learning*, 06 2001.

[14] Q. Lan, Y. Pan, A. Fyshe, and M. White, "Maxmin q-learning: Controlling the estimation bias of q-learning," *CoRR*, vol. abs/2002.06487, 2020. [Online]. Available: https://arxiv.org/abs/2002.06487

[15] O. Anschel, N. Baram, and N. Shimkin, "Deep reinforcement learning with averaged target DQN," *CoRR*, vol. abs/1611.01929, 2016. [Online]. Available: http://arxiv.org/abs/1611.01929

[16] Q. He and X. Hou, "Wd3: Taming the estimation bias in deep reinforcement learning," in *2020 IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI)*, 2020, pp. 391–398.

[17] D. Wu, X. Dong, J. Shen, and S. Hoi, "Reducing estimation bias via triplet-average deep deterministic policy gradient," *IEEE Transactions on Neural Networks and Learning Systems*, vol. PP, pp. 1–13, 01 2020.

[18] S. Thrun and A. Schwartz, "Issues in using function approximation for reinforcement learning," in *In Proceedings of the Fourth Connectionist Models Summer School*. Erlbaum, 1993.

[19] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, "Openai gym," *CoRR*, vol. abs/1606.01540, 2016. [Online]. Available: http://arxiv.org/abs/1606.01540

[20] M. D. Pendrith, M. R. Ryan, and C. C. Sammut, "Estimator variance in reinforcement learning: Theoretical problems and practical solutions," 05 2003.

[21] S. Mannor and J. N. Tsitsiklis, "Mean-variance optimization in markov decision processes," *CoRR*, vol. abs/1104.5601, 2011. [Online]. Available: http://arxiv.org/abs/1104.5601

[22] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. P. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," *CoRR*, vol. abs/1602.01783, 2016. [Online]. Available: http://arxiv.org/abs/1602.01783

[23] L. Espeholt, H. Soyer, R. Munos, K. Simonyan, V. Mnih, T. Ward, Y. Doron, V. Firoiu, T. Harley, I. Dunning, S. Legg, and K. Kavukcuoglu, "Impala: Scalable distributed deep-rl with importance weighted actor-learner architectures," 2018.

[24] F. S. He, Y. Liu, A. G. Schwing, and J. Peng, "Learning to play in a day: Faster deep reinforcement learning by optimality tightening," 2016.

[25] R. Munos, T. Stepleton, A. Harutyunyan, and M. G. Bellemare, "Safe and efficient off-policy reinforcement learning," 2016.

[26] M. Petrik and B. Scherrer, "Biasing approximate dynamic programming with a lower discount factor," in *Advances in Neural Information Processing Systems*, D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, Eds., vol. 21. Curran Associates, Inc., 2009. [Online]. Available: https://proceedings.neurips.cc/paper/2008/file/08c5433a60135c32e34f46a71175850c-Paper.pdf

[27] Y. Flet-Berliac, R. Ouhamma, O.-A. Maillard, and P. Preux, "Learning value functions in deep policy gradients using residual variance," 2021.

[28] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," 2019.

[29] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller, "Deterministic policy gradient algorithms," *31st International Conference on Machine Learning, ICML 2014*, vol. 1, 06 2014.

[30] R. Bellman, *Dynamic Programming*. Dover Publications, 1957.

[31] L. ji Lin, "Self-improving reactive agents based on reinforcement learning, planning and teaching," in *Machine Learning*, 1992, pp. 293–321.

[32] S. Nadarajah and S. Kotz, "Exact distribution of the max/min of two gaussian random variables," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 16, no. 2, pp. 210–212, 2008.

[33] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[34] B. Afonja, "The moments of the maximum of correlated normal and t-variates," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 34, no. 2, pp. 251–262, 1972. [Online]. Available: http://www.jstor.org/stable/2985184

[35] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 5026–5033.

[36] I. Parberry, *Introduction to Game Physics with Box2D*, 1st ed. USA: CRC Press, Inc., 2013.