Finite Sample Analysis of Mean-Volatility Actor-Critic for Risk-Averse Reinforcement Learning

Finite Sample Analysis of Mean-Volatility Actor-Critic for Risk-Averse Reinforcement Learning

Khaled Eldowa **Lorenzo Bisi** **Marcello Restelli**
Politecnico di Milano, Milan, Italy

Abstract

The goal in the standard reinforcement learning problem is to find a policy that optimizes the expected return. However, such an objective is not adequate in a lot of real-life applications, like finance, where controlling the uncertainty of the outcome is imperative. The mean-volatility objective penalizes, through a tunable parameter, policies with high variance of the per-step reward. An interesting property of this objective is that it admits simple linear Bellman equations that resemble, up to a reward transformation, those of the risk-neutral case. However, the required reward transformation is policy-dependent, and requires the (usually unknown) expected return of the used policy. In this work, we propose two general methods for policy evaluation under the mean-volatility objective: the direct method and the factored method. We then extend recent results for finite sample analysis in the risk-neutral actor-critic setting to the mean-volatility case. Our analysis shows that the sample complexity to attain an ϵ-accurate stationary point is the same as that of the risk-neutral version, using either policy evaluation method for training the critic. Finally, we carry out experiments to test the proposed methods in a simple environment that exhibits some trade-off between optimality, in expectation, and uncertainty of outcome.

Proceedings of the 25[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2022, Valencia, Spain. PMLR: Volume 151.

1 INTRODUCTION

Reinforcement Learning has recently gained much attention thanks to the exceptional results obtained in challenging benchmarks like robotic locomotion (Schulman et al., 2015, 2017), arcade video games (Mnih et al., 2013), multi-player games (Berner et al., 2019) and complex board games (Silver et al., 2016). Therefore, there is naturally a growing interest in translating its success from games and virtual domains to real-world contexts (Bisi et al., 2020a; Castelletti et al., 2010). In order to fill this gap, however, a number of issues need to be solved.

A particularly interesting case concerns high-stakes applications like finance, health and robotics. While in these domains the application of advanced strategies obtained with reinforcement learning might produce dramatic benefits, the sensitivity linked to these topics makes it mandatory to take into account the *risk* connected to the employed policies. The risk-averse reinforcement learning literature has studied the implications of explicitly considering risk in MDPs (Sobel, 1982) developing several possible solutions (Tamar et al., 2012; Chow et al., 2015, 2017a; Nass et al., 2019). Measuring risk is alone an important research topic, which has been deeply analysed in the financial mathematics domain (Artzner et al., 1999; Rockafellar et al., 2006). Selecting the correct risk-measure is, in the end, a task-dependent issue, and its choice should be a good trade-off between ensuring mathematical properties (Ruszczyński, 2010), allowing ease of interpretation by stakeholders, and guaranteeing acceptable optimization performance.

The Mean-Volatility objective, recently introduced by (Bisi et al., 2020b), has been shown to have these characteristics, and, thanks to its favourable mathematical properties, it can be optimized with state-of-the art techniques with slight modifications to the original algorithms (Bisi et al., 2020b; Zhang et al., 2021), thus, empirically enjoying competitive learning performance.

The development of novel analysis techniques has al-

lowed the RL literature to produce a number of interesting results on the *finite-sample* complexity of many RL algorithms (Lazaric et al., 2012; Farahmand, 2011; Liu et al., 2020). Establishing the correct sample complexity of state-of-the-art algorithms as, for instance, the well-known actor-critic scheme is a hot topic, which is receiving growing attention (Yang et al., 2018; Wu et al., 2020; Chen et al., 2021; Wang et al., 2019; Kumar et al., 2019; Xu et al., 2020). On the other hand, few works have been dedicated to derive the complexity of risk-averse approaches (Jiang and Powell, 2018; Fei et al., 2020). Penalized risk-averse objectives as Mean-Variance and Mean-Volatility (Tamar et al., 2012; Bisi et al., 2020b) need to estimate the expected return to compute the policy gradient. However, how the consequent estimation error translates in terms of convergence rate is an issue which has not been investigated yet. How do the various error sources compound in the gradient estimation? Is it possible to obtain the guarantees of risk-neutral algorithms in this risk-averse setting? This paper offers an answer to those questions by means of a finite-sample analysis of a mean-volatility actor-critic algorithm.

Our contributions are as follows: (i) We propose two methods (the *direct* one and the *factored* one) for the policy evaluation problem of the mean-volatility. We provide a finite sample bound for a semi-gradient TD(0) approach applied to the direct case. (ii) The previous contribution is used as input for an analysis on an actor-critic algorithm for which we bound the sample complexity necessary for reaching a ϵ-accurate stationary point. All provided bounds are valid in expectation. (iii) We validate our theoretical results by means of an empirical study on a stochastic environment.

2 RELATED WORKS

The mean-volatility objective has been first introduced in (Bisi et al., 2020b), where it was optimized through a trust-region approach, but without providing any finite sample analysis results. The technique has been extended in (Zhang et al., 2021), where the optimization of the same objective was pursued by means of a framework, which allows to decompose the problem into a series of standard MDPs. The authors were able to show the asymptotic convergence of the method to a local optimum (stationary point), but they did not provide any convergence rate. Convergence to a local optimum is typically the best that one can hope for even for risk-neutral policy optimization approaches, unless the problem presents particular favourable features (Agarwal et al., 2021). For what concerns the risk-neutral side, several finite sample analyses have been recently developed for the actor-critic approach (Yang et al., 2018; Wu et al., 2020; Chen et al., 2021; Wang et al., 2019; Kumar et al., 2019; Xu et al., 2020). In this work, we follow the approach suggested in (Xu et al., 2020) to analyse the mean-volatility case, evaluating to which extent our risk-averse extension impacts the risk-neutral convergence rate. This analysis is interesting because, differently from (Chen et al., 2021) for instance, it allows to considers the continuous action case. This is important because enabling the access to continuous actions without losing the advantages of TD-learning is one of the main advantages of actor-critic schemes.

As it is the case for (Zhang et al., 2021), many risk-averse policy gradient approaches offer asymptotic convergence guarantees (Tamar et al., 2012, 2015; Chow et al., 2017b), but they do not provide finite sample analyses. There are only few works focusing on this kind of analysis on algorithms optimizing risk-averse objective. The work in (Jiang and Powell, 2018) optimizes a dynamic coherent risk-measure that involves as static conditional risk-measure either CVaR or VaR, by means of an approximated dynamic programming approach, similar in spirit to Q-learning. The authors provide the convergence rate in terms of the expected deviation from the optimal Q-function. Recently, in (Fei et al., 2020), the authors analysed the model-free optimization of the Entropic Risk-Measure, through two different value-based algorithms. They prove a sub-linear regret bound which can be used to derive the finite-sample complexity of the approaches. While being interesting methods, we remark that these works are not directly comparable to our analysis, since they involve value-based approaches and different objectives.

3 PROBLEM FORMULATION

3.1 Preliminaries

We assume that our *Markov Decision Process* (MDP) is characterized by the following tuple: $\langle \mathcal{S}, \mathcal{A}, P, R, \gamma, \mu_0 \rangle$. Here, $\mathcal{S}$ and $\mathcal{A}$ are the state and action spaces, which we assume to be measurable sets, $P : \mathcal{S} \times \mathcal{A} \to \mathbb{P}(\mathcal{S})$ is the transition kernel, $R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward function, γ is the discount factor, and μ_0 is the initial state distribution. The policies we consider are assumed to belong to some class Π of policies parameterized by a vector $\theta \in \mathbb{R}^{d_\theta}$, and $\pi_\theta(a|s)$ is continuously differentiable with respect to θ for any state-action pair. For a policy π, we define its expected return as follows:

$$J_\pi := (1-\gamma) \mathop{\mathbb{E}}_{\substack{s_0 \sim \mu_0 \\ a_t \sim \pi(\cdot|s_t) \\ s_{t+1} \sim P(\cdot|s_t, a_t)}} \left[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t) \right]. \quad (1)$$

This definition uses the normalization factor $(1-\gamma)$, which ensures that J_π belongs to the interval $[-R_{\max}, R_{\max}]$, where $R_{\max}$ is an upper bound on the absolute value of the reward for any state-action pair. In this work, we adopt the *reward volatility*, introduced in (Bisi et al., 2020b), to measure risk. The reward volatility is concerned with the variance of the per-step reward instead of the variance of the return. Firstly, we define the discounted state distribution as

$$d_{\mu_0,\pi}(s) := (1-\gamma) \mathop{\mathbb{E}}_{s_0 \sim \mu_0(\cdot)} \left[\sum_{t=0}^{\infty} \gamma^t p_{\pi_\theta}(s_0 \xrightarrow{t} s) \right], \quad (2)$$

where $p_\pi(s_0 \xrightarrow{t} s)$ is the probability of ending up in s after t steps starting from s_0 and executing policy π. Using this definition, we can rewrite (1) as:

$$J_\pi = \mathop{\mathbb{E}}_{\substack{s \sim d_{\mu_0,\pi}(\cdot) \\ a \sim \pi(\cdot|s)}} [R(s,a)].$$

That is, J_π can be seen as the expected value of the step-reward random variable where the states are drawn from the discounted state distribution (which depends on the policy) and the actions are drawn according to the policy. The reward volatility ν_π^2 is the variance of this random variable. In other words,

$$\nu_\pi^2 \coloneqq \mathop{\mathbb{E}}_{\substack{s \sim d_{\mu_0,\pi}(\cdot) \\ a \sim \pi(\cdot|s)}} \left[\left(R(s,a) - J_\pi \right)^2 \right] \quad (3)$$

Our goal is to find a policy maximizing the mean-volatility (Bisi et al., 2020b), which is defined, for a policy π, as $\eta_\pi = J_\pi - \lambda \nu_\pi^2$, where $\lambda \geq 0$ is a parameter that penalizes large reward volatility. For the mean-volatility objective, we can define the transformed state-value function V_π^λ as:

$$V_\pi^\lambda(s) := \mathop{\mathbb{E}}_{\substack{a_t \sim \pi(\cdot|s_t) \\ s_{t+1} \sim P(\cdot|s_t,a_t)}} \left[\sum_{t=0}^{\infty} \gamma^t R_\pi^\lambda(s_t,a_t) \middle| s_0 = s \right], \quad (4)$$

where $R_\pi^\lambda(s,a) := R(s,a) - \lambda (R(s,a) - J_\pi)^2$ can be seen as a *policy-dependent* reward transformation. The transformed action-value function Q_π^λ can be defined in a similar manner. It is easy to see that these transformed value functions, unlike the mean-variance case, admit simple Bellman expectation equations. For example, we have that

$$V_\pi^\lambda(s) = \mathop{\mathbb{E}}_{a \sim \pi(\cdot|s)} \left[R_\pi^\lambda(s,a) \right] + \gamma \mathop{\mathbb{E}}_{s' \sim P^\pi(\cdot|s)} \left[V_\pi^\lambda(s') \right].$$

Unfortunately, since the transformed reward function is policy-dependent, one cannot directly use classic tools like policy iteration and value iteration to optimize the mean-volatility. This is because results like the policy improvement theorem (Sutton and Barto, 1998) or the contraction property of the Bellman optimality operator (Bertsekas and Tsitsiklis, 1996) do not necessarily hold anymore.

However, adapting policy gradient methods to the mean-volatility setting is indeed easier (Bisi et al., 2020b). When using parameterized policies (in the manner described before), the gradient of η_θ [1] with respect to θ can derived as:

$$\nabla_\theta \eta_\theta = \mathop{\mathbb{E}}_{\substack{s \sim d_{\mu_0,\pi_\theta}(\cdot) \\ a \sim \pi_\theta(\cdot|s)}} \left[A_{\pi_\theta}^\lambda(s,a) \nabla_\theta \log \pi_\theta(a|s) \right], \quad (5)$$

where $A_\pi^\lambda(s,a) := Q_\pi^\lambda(s,a) - V_\pi^\lambda(s)$ is the transformed advantage function.

3.2 Mean-Volatility Policy Evaluation Techniques

If we wish to estimate the transformed value function V_π^λ for a given policy π, the most immediate idea is to transform the rewards using R_π^λ, and then use any risk-neutral policy evaluation algorithm. We call this approach the *direct-method*. In practice, performing the aforementioned reward transformation requires one to first estimate (via sampling) the (normalized) expected return J_π of the policy under evaluation. Denote by $\hat{J}_\pi$ our estimate of J_π, and by $\hat{R}_\pi^\lambda$ the resulting reward transformation when using $\hat{J}_\pi$ instead of J_π. One can then see $\hat{R}_\pi^\lambda$ as an estimator for the true reward transformation R_π^λ. Clearly, this estimator is biased if $\hat{J}_\pi$ is biased. Even if $\hat{J}_\pi$ was unbiased, $\hat{R}_\pi^\lambda$ would still be biased if we do not use two independently estimated versions of $\hat{J}_\pi$ since it is involved in a squared term. It is then natural to wonder how using such an approximate reward transformation affects the adopted policy evaluation algorithm. One may ask whether this kind of algorithm converges, and how distant (according to some measure) the obtained solution is from the exact one. Answering this question could enable us to show that the overall algorithm is consistent when $\hat{J}_\pi$ is consistent. This could also enable us to infer the order of the number of samples (used either by the policy evaluation algorithm or the sampling process for estimating J_π) needed to keep the estimation error below some given level ϵ. An alternative approach, which we will call the *factored-method*, relies on the following alternative expression for V_π^λ (see Appendix B for the derivation):

$$V_\pi^\lambda(s) = (1 + 2\lambda J_\pi) V^\pi(s) - \lambda M^\pi(s) - \frac{\lambda}{1-\gamma} J_\pi^2, \quad (6)$$

[1] We usually write η_θ instead of η_{π_θ} for notational convenience.

where we call $M^\pi : \mathcal{S} \to \mathbb{R}$ the *second moment value function*[2], which is defined as follows:

$$M^\pi(s) := \mathop{\mathbb{E}}_{\substack{a_t \sim \pi(\cdot|s_t) \\ s_{t+1} \sim P(\cdot|s_t, a_t)}} \left[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t)^2 \middle| s_0 = s \right]. \tag{7}$$

Squaring the rewards can be seen as a deterministic (policy-independent) reward transformation. Thus, M^π can be learned by adapting any algorithm that can be used for learning V^π, and any accuracy guarantees (e.g. finite-time bounds) on the learned estimate of V^π can be adapted for M^π; we would just need to consider that the range of values of the step-reward is different. A natural choice would be to learn both functions in parallel using the same algorithm and the same data. The factored method involves estimating V^π, M^π, and J_π, and then plugging them in (6) to obtain an estimate of V_π^λ. Note that this approach bears some resemblance to the approach adopted in (Tamar et al., 2016) for estimating the variance of the reward to go. However, the approach in our case is simpler. This is mainly because the three quantities to be estimated (namely V^π, M^π, and J_π) can be learned *separately* using standard methods, and only combined at the end via (6).

3.3 Monte-Carlo Estimation of the Expected Return

No matter which policy evaluation method one chooses to use, the estimation of the expected return J_π is a crucial step. We will adopt a simple Monte-Carlo procedure for estimating $\hat{J}_\pi$. In this procedure, we simulate L trajectories each truncated at a fixed horizon of T_J steps, and then average the (normalized) truncated returns from these trajectories. That is, if $G_i \coloneqq \sum_{t=0}^{T_J-1} \gamma^t R(s_{i,t}, a_{i,t})$ denotes the truncated return from trajectory i, then $\hat{J}_\pi$ is given by:

$$\hat{J}_\pi \coloneqq \frac{1}{L} \sum_{i=0}^{L-1} (1-\gamma) G_i.$$

Note that $\hat{J}_\pi$ is not necessarily an unbiased estimate of J_π since we are truncating the returns. This bias, however, can be arbitrarily reduced by making the trajectories long enough. Also, we will use only a single estimate of J_π in our algorithms, which can introduce bias due to the involvement of J_π in squared terms in both policy evaluation methods. These issues will be taken into account in our analysis.

[2]It is the second moment of the step reward $R(s', a')$ (where $s' \sim d_\pi(\cdot|s)$ and $a' \sim \pi(\cdot|s')$), not of the return when starting from s.

3.4 The Critic Algorithm

We will extend the analysis in (Xu et al., 2020) conducted over the actor-critic algorithm in the risk-neutral setting to our mean-volatility problem. We start by describing our extension of the critic. In (Xu et al., 2020), the critic is a temporal difference algorithm that uses linear function approximation. More specifically, they use a mini-batch version of linear TD(0) in which a mini-batch of samples is used to perform the updates instead of just a single sample. Their motivation for adopting mini-batch updates is that the iterates can be driven arbitrarily close, in expectation, to the TD fixed point by increasing the mini-batch size while using a fixed step-size. Using this approach, they were able prove a better sample complexity than that provided in other works in the literature (e.g. (Bhandari et al., 2018)).

If we are to use the direct method, our aim will be to use that algorithm to learn V_π^λ by transforming the rewards using $\hat{R}_\pi^\lambda$ (which depends on $\hat{J}_\pi$). If we are to use the factored method instead, we can learn V^π by directly using the algorithm, and learn M^π in the same manner except that we square the rewards. For all three functions, we will consider a linear approximation scheme where candidate functions belong to the function space $\{f_\omega : \omega \in \mathbb{R}^{d_\omega} \text{ and } f_\omega(.) = \omega^\intercal \phi(.)\}$, where $\varphi_i : S \to \mathbb{R}, i = 1, \ldots, d_\omega$. are basis functions defined over the states, and $\phi(.) := (\varphi_1(.), \ldots, \varphi_{d_\omega}(.))^\intercal$ is the corresponding feature mapping.

Algorithm 1 is a generalization of Algorithm 2 in (Xu et al., 2020)[3], where the difference is that we get to choose the reward function f_R to be used in the algorithm. This could be:

- $f_R(s,a) = R(s,a)$, if we are learning V^π.
- $f_R(s,a) = R^2(s,a)$, if we are learning M^π.
- $f_R(s,a) = R(s,a) - \lambda(R(s,a) - \hat{J}_\pi)^2$, if we are learning V_π^λ using the direct method.[4]

Note that in the last case, f_R is a function of $\hat{J}_\pi$ and λ, which subsequently become parameters of the algorithm. As for the rest of the parameters, T_c is the number of iterations, M is the mini-batch size, and β is the step-size.

[3]Note that, unlike in (Xu et al., 2020), we use ω for the critic's parameters and the more common choice of θ for the policy's parameters.

[4]We refer to this version of the algorithm as direct mini-batch TD.

Algorithm 1 Mini-batch TD

Input: $s_{\text{ini}}, \theta, \phi(\cdot), \gamma, \beta, T_c, M, f_R$
Initialize: ω_0
Set $s_{-1,M} = s_{\text{ini}}$
for $k = 0, \dots, T_c - 1$ **do**
  $s_{k,0} = s_{k-1,M}$.
  for $j = 0, \dots, M-1$ **do**
    $a_{k,j} \sim \pi_\theta(s_{k,j})$, $s_{k,j+1} \sim P(.|s_{k,j}, a_{k,j})$
    $\widetilde{R}_{k,j} = f_R(s_{k,j}, a_{k,j})$
    $\delta_{k,j} = \widetilde{R}_{k,j} + \gamma\phi(s_{k,j+1})^\intercal\omega_k - \phi(s_{k,j})^\intercal\omega_k$
  end for
  $\omega_{k+1} = \omega_k + \beta\frac{1}{M}\sum_{j=0}^{M-1}\delta_{k,j}\phi(s_{k,j})$
end for
Output: $\omega_{T_c}, s_{k,M}$

3.5 The Actor Algorithm

In (Xu et al., 2020), they adopt an *advantage actor critic* (A2C) approach, where they also use mini-batches to perform the stochastic gradient ascent updates. This means that the policy updates take the following from:

$$\theta_{t+1} = \theta_t + \alpha\frac{1}{B}\sum_{i=0}^{B-1}\delta_{t,i}\nabla_\theta \log \pi_{\theta_t}(a_{t,i}|s_{t,i}), \tag{8}$$

where B is the mini-batch size, α is the step-size, $\delta_{t,i} = R(s_{t,i}, a_{t,i}) + \gamma\hat{V}_t(s_{t,i+1}) - \hat{V}_t(s_{t,i})$ is the *temporal difference (TD) error* at the i$^{\text{th}}$ step, and $\hat{V}_t$ is the critic learned at iteration t. Note that $\delta_{t,i}$ is, in effect, an estimate of the advantage function at $(s_{t,i}, a_{t,i})$. Thanks to the policy gradient expression of the mean-volatility (see (5)), adapting this approach to our case would just involve using the (estimated) transformed reward and the (estimated) transformed value function in place of their risk-neutral counterparts in the TD-error. To collect the samples of the mini-batch, the agent interacts with a slightly modified MDP characterized by the following transition kernel:

$$\tilde{P}(\cdot|s,a) = \gamma P(\cdot|s,a) + (1-\gamma)\mu_0(\cdot),$$

where P is the transition kernel of the original MDP. That is, at each step, the next state is sampled according to the original kernel with probability γ, while we draw the next state from the initial state distribution (i.e. restart) with probability $1-\gamma$. This sampling process causes the encountered states to be distributed, at steady-state, according to the discounted state distribution (Thomas, 2014). While this is indeed the desired distribution of states (see (5)), a side effect is that the next state (s_{t+1}) utilized in the TD-error expression is now sampled from $\tilde{P}(\cdot|s_t, a_t)$, whereas it should be sampled from $P(\cdot|s_t, a_t)$. This introduces a subtle bias in the algorithm, which is not accounted for in the analysis of (Xu et al., 2020). To remedy this, we employ a slightly altered sampling process. At any time step t, consider two different random variables for the next state, namely, s_{t+1} and s'_{t+1}, with different distributions. The latter is distributed according to the standard kernel (i.e., $s'_{t+1} \sim P(\cdot|s_t, a_t)$), while s_{t+1} is sampled from the following variant of the modified kernel[5] $\tilde{P}(\cdot|s_t, a_t, s'_{t+1}) := \gamma\boldsymbol{\delta}_{s'_{t+1}}(\cdot) + (1-\gamma)\mu_0(\cdot)$. That is, with probability γ, s_{t+1} is the same as s'_{t+1}, and with probability $1-\gamma$, s_{t+1} is drawn from the initial state distribution. In any case, s'_{t+1} is the one we use as the next state in the TD-error, whereas s_{t+1} is the state from which we resume sampling the rest of the actor mini-batch[6]. With the proposed modification, the analysis of (Xu et al., 2020) remains largely applicable, we just need to account for the extra performed sampling when we consider the sample complexity of the algorithm.

Algorithm 2 demonstrates our proposed adaptation of the mini-batch actor-critic algorithm to the mean-volatility setting. In the algorithm description, we used that (for any state-action pair) $\psi_\theta(s,a) := \nabla \log \pi_\theta(a|s)$, which is referred to as the *score function* of policy π_θ. Note that the algorithm leaves the choice of the critic procedure open. In particular, if we want to use the direct method, then we can call the mini-batch TD algorithm with $f_R(s,a) = R(s,a) - \lambda(R(s,a) - \hat{J}_t)^2$. If we name the learned parameter vector ω_t, then we can set $\hat{V}_t^\lambda(s) := \phi(s)^\intercal\omega_t, \forall s \in \mathcal{S}$. If we want to use the factored method, we can call the mini-batch TD algorithm with $f_R(s,a) = R(s,a)$ and $f_R(s,a) = R^2(s,a)$ for learning $\hat{V}_t$ and $\hat{M}_t$ respectively[7]. If we then denote by ω_t^v and ω_t^m the learned parameter vectors for $\hat{V}_t$ and $\hat{M}_t$ respectively, we can set ($\forall s \in \mathcal{S}$):

$$\hat{V}_t^\lambda(s) = (1 + 2\lambda\hat{J}_t)\phi(s)^\intercal\omega_t^v - \lambda\phi(s)^\intercal\omega_t^m - \frac{\lambda}{1-\gamma}\hat{J}_t^2.$$

Note that the algorithm takes L and T_J as parameters, which denote the number of trajectories and the number steps per trajectory used in the Monte-Carlo estimation of the expected return, which we have described before.

3.6 General Assumptions

Before describing our results, we highlight the main required technical assumptions.

[5] Here, $\boldsymbol{\delta}$ is the Dirac delta function.

[6] Note that the proposed sampling process does not require a generative model, it only requires that we can halt the trajectory at any time and restart from the initial state distribution.

[7] In practice, one would use the same sample path for learning both functions.

Algorithm 2 Mini-batch Mean-Volatility Actor-Critic (Mini-batch MVAC)

Input: Policy Class π_θ, $\phi(\cdot)$, $\mu_0(\cdot)$, λ, γ, L, T_J, T, B, α.
Initialize: θ_0, $s_{-1,B} \sim \mu_0(\cdot)$
for $t = 0, \dots, T-1$ **do**
  $s_{\text{ini}} = s_{t-1,B}$
  Estimate J:
    $\hat{J}_t = \text{Monte-Carlo-J}(\pi_{\theta_t}, \gamma, L, T_J)$.
  Estimate the Critic $\hat{V}_t^\lambda$ (utilizing $\hat{J}_t$, s_{ini}).
  Set $s_{t,0}$ as last state from the critic sampling.
  Actor mini-batch sampling:
  for $i = 0, \dots, B-1$ **do**
    $a_{t,i} \sim \pi_\theta(s_{t,i})$
    $s'_{t,i+1} \sim P(\cdot|s_{t,i}, a_{t,i})$
    $s_{t,i+1} \sim \tilde{P}(\cdot|s_{t,i}, a_{t,i}, s'_{t,i+1})$
    $\tilde{R}_{t,i} = R(s_{t,i}, a_{t,i}) - \lambda(R(s_{t,i}, a_{t,i}) - \hat{J}_t)^2$
    $\delta_{t,i} = \tilde{R}_{t,i} + \gamma V_t^\lambda(s'_{t,i+1}) - V_t^\lambda(s_{t,i})$
  end for
  Actor update:
  $\theta_{t+1} = \theta_t + \alpha \frac{1}{B} \sum_{i=0}^{B-1} \delta_{t,i} \, \psi_{\theta_t}(s_{t,i}, a_{t,i})$
end for
Output: $\theta_{\hat{T}}$ with $\hat{T}$ chosen uniformly from $\{1, \dots, T\}$.

Assumption 1. $\forall (s,a) \in S \times A$:

(i) $|R(s,a)| \leq R_{max}$.
(ii) $\pi_\theta(a|s)$ *is differentiable w.r.t.* θ.
(iii) $\exists\, C_\psi > 0 : \forall \theta \; \|\psi_\theta(s,a)\|_2 \leq C_\psi$.
(iv) $\exists\; L_\psi > 0 : \forall \theta_1, \theta_2 \; \|\psi_{\theta_1}(s,a) - \psi_{\theta_2}(s,a)\|_2 \leq L_\psi \|\theta_1 - \theta_2\|_2$.
(v) $\exists\; C_\pi > 0 : \forall \theta_1, \theta_2 \; \|\pi_{\theta_1}(.|s) - \pi_{\theta_2}(.|s)\|_{TV} \leq C_\pi \|\theta_1 - \theta_2\|_2$,

where, for a probability density function $q(.)$, $\|q(.)\|_{TV} := \frac{1}{2}\int_s |q(ds)|$.

Assumptions 1.iii and 1.iv assert that, for any policy in our class of policies, the score function is bounded and smooth, while assumption 1.v asserts that the chosen class of policies is smooth in the described sense. Note that by Assumption 1.i and the definition of J_π in (1), $\forall (s,a) \in S \times A$ and $\lambda \geq 0$, we have that

$$\left|R(s,a) - \lambda(R(s,a) - J_\pi)^2\right| \leq R_{\lambda,\max},$$

where $R_{\lambda,\max} := R_{\max} + 4\lambda R_{\max}^2$. We also make the following assumption on the basis functions and the feature mapping that we use to learn V_π^λ.

Assumption 2. $\exists\, C_\phi > 0 : \forall s \in S \; \|\phi(s)\|_2 \leq C_\phi$. *Furthermore, the basis functions* $\varphi_i(\cdot), i = 1, ..., d_\omega$ *are mutually linearly independent.*

The following is an assumption on the regularity of the MDP.

Assumption 3 (Uniform Ergodicity, Adapted from (Xu et al., 2020))**.** *For any* $\theta \in \mathbb{R}^{d_\theta}$*, consider the MDP with policy* π_θ *and the transition kernel* $P(\cdot|s,a)$ *or* $\tilde{P}(\cdot|s,a) = \gamma P(\cdot|s,a) + (1-\gamma)\xi(\cdot)$*, where* $\xi(\cdot)$ *can be* μ_0 *or* $P(\cdot|\hat{s},\hat{a})$ *for any* $(\hat{s},\hat{a}) \in \mathcal{S} \times \mathcal{A}$*. Let* μ_{π_θ} *be the stationary state distribution of the MDP when acting with policy* π_θ*. There exists constants* $\kappa > 0$ *and* $\rho \in (0,1)$ *such that:*

$$\sup_{s \in \mathcal{S}} \|\mathbb{P}(s_t \in \cdot | s_0 = s) - \mu_{\pi_\theta}(\cdot)\|_{TV} \leq \kappa \rho^t, \forall t \geq 0.$$

4 MAIN RESULTS

In this section, we present the the main finite sample analysis results. We will first consider the analysis of the direct mini-batch TD algorithm for learning the transformed value function, and then we will consider the full mean-volatility actor-critic procedure, where the critic is learned using direct mini-batch TD. The analysis of the factored method case is provided in Appendix B.

4.1 Direct Mini-Batch TD Analysis

In the direct method, if $\hat{J}$ (and subsequently, the estimated transformed reward function) is fixed, we can invoke the results from (Tsitsiklis and Van Roy, 1997) about the convergence of TD learning with linear function approximation. In particular, if we define[8] $b(\hat{J}) := \mathbb{E}_{\mu_\theta}\left[\phi(s_t) R^\lambda(s_t, a_t, \hat{J})\right]$, and $A := \mathbb{E}_{\mu_\theta}[\phi(s_t)(\gamma\phi(s_{t+1}) - \phi(s_t))^\intercal]$, then the algorithm converges to a point $\omega^*_{\hat{J}}$ such that $A\omega^*_{\hat{J}} + b(\hat{J}) = 0$. However, our goal is to describe the convergence rate, in expectation, of the critic to ω^*_J, where J is the true expected return of the policy under evaluation, not to $\omega^*_{\hat{J}}$. Moreover, $\hat{J}$ is not fixed; it is a random variable whose properties depend on the number of trajectories L (and their length T_J) used to estimate it. The main idea of the analysis is thus to bound how far we expect $\omega^*_{\hat{J}}$ to be from ω^*_J in terms of L and T_J. Before presenting the bound, we state the following result[9], which is an adaptation of a similar statement in (Xu et al., 2020). There exists a positive constant χ_A such that, for any $\omega \in \mathbb{R}^{d_\omega}$ and any value of our (bounded) estimate $\hat{J}$, we have that

$$\langle (\omega - \omega^*_{\hat{J}}), A(\omega - \omega^*_{\hat{J}}) \rangle \leq -\frac{\chi_A}{2} \left\|\omega - \omega^*_{\hat{J}}\right\|_2^2.$$

[8] Here, μ_θ is the stationary distribution of policy π_θ, which is the policy under evaluation. Note that since π_θ is fixed in this subsection, we drop the θ subscripts from the notation for simplicity.

[9] This can be seen as a consequence of Lemmas 1 and 3 in (Bhandari et al., 2018).

Theorem 1 (Critic's Bound)**.** *Suppose Assumptions 1 to 3 hold, and suppose we are given a policy π_θ (with normalized expected return J) and risk parameter λ. Suppose that a Monte-Carlo estimate $\hat{J}$ is obtained for π_θ as described before, and then Algorithm 1 is run for T_c steps using $f_R(s,a) = R(s,a) - \lambda(R(s,a) - \hat{J})^2$. Then, for $\beta \leq \min\{\mathcal{O}(\chi_A), \mathcal{O}(\chi_A^{-1})\}$, we have that*

$$\mathbb{E}\left[\left\|\omega_{T_c}^{\hat{J}} - \omega_J^*\right\|_2^2\right] \leq 4\|\omega_0 - \omega_J^*\|_2^2(1 - \mathcal{O}(\chi_A\beta))^{T_c} + \mathcal{O}\left(\frac{\chi_A^{-1} + \beta}{\chi_A M}\right) + \frac{2}{\bar{\sigma}^2}\left[1 + 2(1 - \mathcal{O}(\chi_A\beta))^{T_c}\right]\mathcal{O}\left(\lambda^2\left(\gamma^{2T_J} + \frac{1}{L}\right)\right),$$

where $\omega_{T_c}^{\hat{J}}$ is the parameter vector obtained after T_c iterations of the algorithm while using $\hat{J}$ to perform the reward transformation, $\bar{\sigma}$ is the smallest singular value of the matrix A, and the expectation is over both the Monte-Carlo estimation of $\hat{J}$ and the TD algorithm. Furthermore, for a sufficiently small $\epsilon > 0$, to achieve an ϵ-accurate solution, that is,

$$\mathbb{E}\left[\left\|\omega_{T_c}^{\hat{J}} - \omega_J^*\right\|_2^2\right] \leq \epsilon,$$

the sample complexity of the algorithm is

$$T_c M + LT_J = \mathcal{O}\big(\epsilon^{-1} log\big(\epsilon^{-1}\big)\big).$$

The proof of this theorem and the next one can be found in Appendix A. The first two terms of the bound are (up to constants) the risk-neutral bound of (Xu et al., 2020). The third term primarily quantifies the inaccuracy of $\hat{J}$, and decays by increasing L and T_J. Interestingly, the obtained sample complexity is the same as the risk-neutral version in (Xu et al., 2020). In fact, only the third term depends on λ, and upon setting it to zero, the risk-neutral bound is recovered. Although a higher degree of risk-aversion (i.e., greater λ) has a negative impact on the bound, it does not affect the order of the required number of samples. It is important to note that the conducted analysis requires that the transformed value function and the expected return are estimated using different data. We highlight in Appendix C the challenges that arise when analyzing the case where the same data is used for estimating both quantities.

4.2 Mean-Volatility Actor-Critic Analysis

Since $\eta(\theta)$[10] is, in general, a non-concave function of θ, we do not expect that we reach a global maximum using a gradient ascent algorithm. Instead, we strive to reach a stationary point of $\eta(\theta)$, and the goal of the analysis is thus to bound $\mathbb{E}\left[\left\|\nabla\eta(\theta_{\hat{T}})\right\|_2^2\right]$ in terms of the number of used samples. Crucial to the analysis of the actor is for the gradient of $\eta(\theta)$ to be Lipschitz continuous. That is, for any $\theta_1, \theta_2 \in \mathbb{R}^{d_\theta}$, there exists a real constant $L_\eta \geq 0$ such that

$$\|\nabla\eta(\theta_1) - \nabla\eta(\theta_2)\|_2 \leq L_\eta\|\theta_1 - \theta_2\|_2.$$

The proof of this statement and other intermediary results can be found in Appendix A. Since the actor relies on the critic for the estimation of the gradient, the convergence of the actor naturally relies on the accuracy of the critic. However, the analysis of the last section was only concerned with how far the critic was from the TD fixed point. We will thus need an additional notion to describe the approximation error incurred due to not only using a linear function, but also for using TD learning, which, in general, leads to a fixed point different from the best approximation in our space of candidate function (Tsitsiklis and Van Roy, 1997). Thus, we define the following quantity to be used in the actor's bound:

$$\xi_{appr} := \sup_{\theta\in\mathbb{R}^{d_\theta}} \mathbb{E}_{s\sim d_{\mu_0,\pi_\theta}(\cdot)}\left[\left|V_{\pi_\theta}^\lambda(s) - \phi(s)^\intercal\omega_{J_\theta}^*\right|^2\right].$$

Theorem 2 (Actor's Bound)**.** *Suppose Assumptions 1 to 3 hold, and suppose we run Algorithm 2 for T iterations with the critic learned as described in Theorem 1, then if $\alpha = \frac{1}{8L_\eta}$, we have:*

$$\mathbb{E}\left[\left\|\nabla\eta(\theta_{\hat{T}})\right\|_2^2\right] \leq \sum_{t=0}^{T-1}\mathbb{E}\left[\|\omega_{J_t}^* - \omega_t\|_2^2\right]\mathcal{O}\left(\frac{1}{T}\right) + \mathcal{O}\left(\frac{1}{B}\right) + \mathcal{O}(\xi_{appr}) + \mathcal{O}\left(\frac{L_\eta}{T}\right) + \mathcal{O}\left(\lambda^2\left(\gamma^{2T_J} + \frac{1}{L}\right)\right),$$

where ω_t is the parameter vector of the learned critic at the t^{th} iteration, and $\omega_{J_t}^*$ is the TD fixed point for the true transformed value function of policy π_{θ_t}. Furthermore, for a sufficiently small $\epsilon > 0$, to achieve an ϵ-accurate stationary point, that is,

$$\mathbb{E}\left[\left\|\nabla\eta(\theta_{\hat{T}})\right\|_2^2\right] \leq \epsilon + \mathcal{O}(\xi_{appr}),$$

the total sample complexity is:

$$T((2-\gamma)B + MT_c + LT_J) = \mathcal{O}\left(\epsilon^{-2}\log\left(\epsilon^{-1}\right)\right).$$

The bound in Theorem 2 is made up of five terms. The first is proportional to the average (across iterations) of how far we expect the critic to be from the TD fixed

[10] $\eta(\theta) := \eta_\theta$.

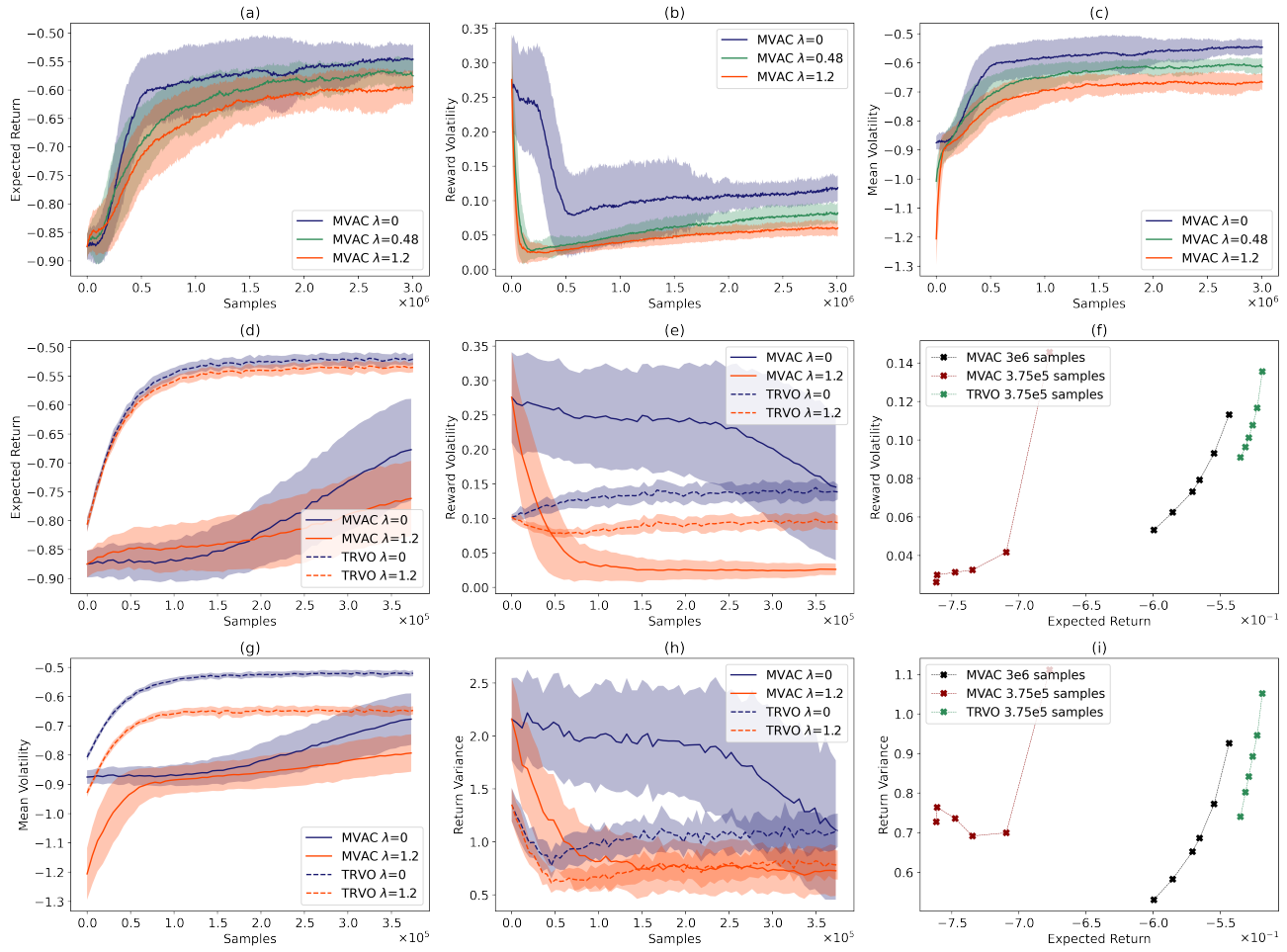

Figure 1: This figures reports the performance of the mini-batch MVAC algorithm in the Point-Reacher environment, and a comparison with TRVO on the same environment. Plots (a), (b), and (c) show the progress of the expected return, reward volatility, and mean-volatility (over 60 runs) as the number of samples increases for three values of λ. Plots (d), (e), (g), and (h) compare the progress of MVAC and TRVO for two values of λ up to 3.75e5 samples. Plot (f) shows the approximated Pareto frontiers (for six values of λ between 0 and 1.2) in terms of expected return and reward volatility, while plot (i) shows the same but for the return variance.

point of the true transformed value function. This is precisely the quantity bounded in Theorem 1. The second term is related to the error due to the variance of the mini-batch estimates of the gradient, and it decays by increasing the mini-batch size. The third term is the approximation error discussed before. The fourth one is an error term that decays as the number of actor iterations increases. The last term, much like the third term in Theorem 2, represents the error due to the inaccuracy of $\hat{J}$, and it decays by increasing L and T_J. Compared to the bound in (Xu et al., 2020), our bound assumes a similar form (albeit some of the quantities are naturally defined differently in our setting), with the exception of the last term. Although estimating the expected return requires extra sampling, the theorem asserts that the sample complexity is still not worsened compared to the risk neutral case.

5 EXPERIMENTS

In this section we empirically validate our algorithms by means of an experimental analysis on an environment called Point-Reacher. In this environment, the agent controls a point mass that moves along the real line in the interval $[-10, 10]$, by taking actions in $[-2, 2]$, denoting the size and the direction of the desired step. The target location is at point 0, and the considered problem is of the continuing type. The closer the agent is to point 0, the larger the received reward. And while larger actions might take the agent

faster to the goal zone, they lead to higher variance of the reached state and the received reward.

We tested[11] the performance of Algorithm 2 in this environment, where the used critic is again direct mini-batch TD. We considered Gaussian policies, where the mean and standard deviation are linear functions of the state. The features we used for the states are Gaussian radial basis functions. The critic also used these same features. A more detailed description of the environment and the used parameters is provided in Appendix A. The first row of Figure 1 shows the performance of mini-batch MVAC in terms of expected return, reward volatility, and mean-volatility for three values of the risk-aversion parameter lambda. Moreover, in plots (f) and (i) of the same figure, the black points form approximated Pareto frontiers obtained by the algorithm for the reward volatility/expected return and return variance/expected return problems respectively. It has been shown in (Bisi et al., 2020b; Zhang et al., 2021) that reward volatility minimization can also be employed as a proxy for return variance reduction. It can be seen then that the algorithm is able to obtain different trade-offs between performance (in terms of expected return) and risk (in terms of either reward volatility or return variance).

While the algorithms presented in this paper are simple enough to facilitate the analysis and keep it focused on the more interesting and less explored aspects of the problem, it is still interesting to compare our empirical results with a more advanced algorithm like TRVO (Bisi et al., 2020b) which optimizes the same objective. Hence, in plots (d), (e), (g), and (h) we compare the performance of mini-batch MVAC with TRVO for two values of lambda but only up to 3.75e5 samples, at which point TRVO has already converged. It can be easily seen that TRVO converges a lot faster (mostly due to the fact that MVAC needs a slower learning rate) and enjoys a smoother learning process. In plots (f) and (i), the approximated Pareto frontiers achieved by TRVO are shown in green. Interestingly, the frontiers obtained by the two algorithms are different although they use the same policy space. Since we stopped MVAC after 3M steps, we conjecture that the optimization may have reached a plateau, and not a stationary point, thus, it could have obtained the same frontier of TRVO, with a larger number of training steps.

6 CONCLUSIONS

The goal of this paper was to shed light on the impact of risk-aversion on the sample complexity of RL algorithms. We analysed the mean-volatility case, focusing, in particular, on an actor-critic algorithm. We developed two different methods for mean volatility policy evaluation: the factored method and the direct method. Firstly, we provided a finite-sample bound for the critic algorithm, which applied the direct method to a mini-batch TD algorithm. Secondly, we extended the analysis to the actor procedure, deriving the sample-complexity of the whole algorithm. Our results show that while increasing risk-aversion negatively affects the error bounds, the sample complexity of the algorithms remains of the same order as that of their risk-neutral counterparts. Finally, we tested the mini-batch MVAC algorithm on a stochastic environment to assess its soundness. We showed that the algorithm is effective in obtaining different trade-offs between the expected return and the reward-volatility according to the desired level of risk-aversion. A challenging future research direction could be analysing the case in which a single batch of samples is used for each iteration, in order to discover the impact of the resulting bias. Furthermore, it would be interesting to analyse the performance of more powerful algorithms like TRVO in order to understand how their empirically superior performance can be justified by theory.

[11] The code of the performed experiments can be found at https://github.com/Khaled-Eldowa/MVAC.

References

A. Agarwal, S. M. Kakade, J. D. Lee, and G. Mahajan. On the theory of policy gradient methods: Optimality, approximation, and distribution shift. *Journal of Machine Learning Research*, 22(98):1–76, 2021.

G. Ajjanagadde, A. Makur, J. Klusowski, S. Xu, et al. Lecture notes on information theory. 2017.

J. Angelova. On moments of sample mean and variance. *International Journal of Pure and Applied Mathematics*, 79, 01 2012.

P. Artzner, F. Delbaen, J.-M. Eber, and D. Heath. Coherent measures of risk. *Mathematical finance*, 9 (3):203–228, 1999. Publisher: Wiley Online Library.

C. Berner, G. Brockman, B. Chan, V. Cheung, P. Debiak, C. Dennison, D. Farhi, Q. Fischer, S. Hashme, C. Hesse, et al. Dota 2 with large scale deep reinforcement learning. *arXiv preprint arXiv:1912.06680*, 2019.

D. P. Bertsekas and J. N. Tsitsiklis. *Neuro-dynamic programming.* Athena Scientific, Belmont, MA, 1996.

J. Bhandari, D. Russo, and R. Singal. A finite time analysis of temporal difference learning with linear function approximation. *Oper. Res.*, 69:950–973, 2018.

L. Bisi, P. Liotet, L. Sabbioni, G. Reho, N. Montali, M. Restelli, and C. Corno. Foreign exchange

trading: a risk-averse batch reinforcement learning approach. In *Proceedings of the First ACM International Conference on AI in Finance*, pages 1–8, 2020a.

L. Bisi, L. Sabbioni, E. Vittori, M. Papini, and M. Restelli. Risk-averse trust region optimization for reward-volatility reduction. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 4583–4589. International Joint Conferences on Artificial Intelligence Organization, 7 2020b. doi: 10.24963/ijcai.2020/632. URL `https://doi.org/10.24963/ijcai.2020/632`. Special Track on AI in FinTech.

A. Castelletti, S. Galelli, M. Restelli, and R. Soncini-Sessa. Tree-based reinforcement learning for optimal water reservoir operation. *Water Resources Research*, 46(9), 2010.

Z. Chen, S. Khodadadian, and S. T. Maguluri. Finite-sample analysis of off-policy natural actor-critic with linear function approximation. *arXiv preprint arXiv:2105.12540*, 2021.

Y. Chow, A. Tamar, S. Mannor, and M. Pavone. Risk-Sensitive and Robust Decision-Making: a CVaR Optimization Approach. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *NeurIPS 28*, pages 1522–1530. Curran Associates, Inc., 2015.

Y. Chow, M. Ghavamzadeh, L. Janson, and M. Pavone. Risk-constrained reinforcement learning with percentile risk criteria. *The Journal of Machine Learning Research*, 18(1):6070–6120, 2017a.

Y. Chow, M. Ghavamzadeh, L. Janson, and M. Pavone. Risk-constrained reinforcement learning with percentile risk criteria. *JMLR*, 18(1):6070–6120, 2017b. Publisher: JMLR. org.

M. Dahleh, M. A. Dahleh, and G. Verghese. Lectures on dynamic systems and control, 2004.

A.-m. Farahmand. Regularization in reinforcement learning. 2011.

Y. Fei, Z. Yang, Y. Chen, Z. Wang, and Q. Xie. Risk-sensitive reinforcement learning: Near-optimal risk-sample tradeoff in regret. *arXiv preprint arXiv:2006.13827*, 2020.

G. H. Golub and C. F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, third edition, 1996.

D. R. Jiang and W. B. Powell. Risk-averse approximate dynamic programming with quantile-based risk measures. *Mathematics of Operations Research*, 43(2):554–579, 2018.

H. Kumar, A. Koppel, and A. Ribeiro. On the sample complexity of actor-critic method for reinforcement learning with function approximation. *arXiv preprint arXiv:1910.08412*, 2019.

A. Lazaric, M. Ghavamzadeh, and R. Munos. Finite-sample analysis of least-squares policy iteration. *Journal of Machine Learning Research*, 13:3041–3074, 2012.

B. Liu, J. Liu, M. Ghavamzadeh, S. Mahadevan, and M. Petrik. Finite-sample analysis of proximal gradient td algorithms. *arXiv preprint arXiv:2006.14364*, 2020.

M. Madiman. On the entropy of sums. In *2008 IEEE Information Theory Workshop*, pages 303–307, 2008. doi: 10.1109/ITW.2008.4578674.

V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller. Playing atari with deep reinforcement learning, 2013.

D. Nass, B. Belousov, and J. Peters. Entropic Risk Measure in Policy Search. *arXiv preprint arXiv:1906.09090*, 2019.

R. T. Rockafellar, S. Uryasev, and M. Zabarankin. Generalized deviations in risk analysis. *Finance and Stochastics*, 10(1):51–74, 2006.

A. Ruszczyński. Risk-averse dynamic programming for Markov decision processes. *Mathematical programming*, 125(2):235–261, 2010. Publisher: Springer.

J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz. Trust region policy optimization. In F. Bach and D. Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1889–1897, Lille, France, 07–09 Jul 2015. PMLR. URL `https://proceedings.mlr.press/v37/schulman15.html`.

J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal Policy Optimization Algorithms. *CoRR*, abs/1707.06347, 2017.

R. Sharma, R. Kumar, R. Saini, and G. Kapoor. Complementary upper bounds for fourth central moment with extensions and applications, 2015.

D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, Jan 2016. ISSN 1476-4687. doi: 10.1038/nature16961. URL `https://doi.org/10.1038/nature16961`.

M. J. Sobel. The variance of discounted markov decision processes. *Journal of Applied Probability*, 19 (4):794–802, 1982. doi: 10.2307/3213832.

R. S. Sutton and A. G. Barto. *Introduction to Reinforcement Learning*. MIT Press, Cambridge, MA, USA, 1st edition, 1998. ISBN 0-262-19398-1.

A. Tamar, D. Di Castro, and S. Mannor. Policy gradients with variance related risk criteria. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, pages 1651–1658, 2012.

A. Tamar, Y. Chow, M. Ghavamzadeh, and S. Mannor. Policy Gradient for Coherent Risk Measures. *CoRR*, page 9, 2015.

A. Tamar, D. D. Castro, and S. Mannor. Learning the variance of the reward-to-go. *Journal of Machine Learning Research*, 17(13):1–36, 2016. URL `http://jmlr.org/papers/v17/14-335.html`.

P. Thomas. Bias in natural actor-critic algorithms. In E. P. Xing and T. Jebara, editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 441–448, Bejing, China, 22–24 Jun 2014. PMLR. URL `https://proceedings.mlr.press/v32/thomas14.html`.

J. Tsitsiklis and B. Van Roy. An analysis of temporal-difference learning with function approximation. *IEEE Transactions on Automatic Control*, 42(5): 674–690, 1997. doi: 10.1109/9.580874.

A. B. Tsybakov. *Introduction to Nonparametric Estimation*. Springer Publishing Company, Incorporated, 1st edition, 2008. ISBN 0387790519.

L. Wang, Q. Cai, Z. Yang, and Z. Wang. Neural policy gradient methods: Global optimality and rates of convergence. *arXiv preprint arXiv:1909.01150*, 2019.

Y. Wu, W. Zhang, P. Xu, and Q. Gu. A finite time analysis of two time-scale actor critic methods. *arXiv preprint arXiv:2005.01350*, 2020.

T. Xu, Z. Wang, and Y. Liang. Improving sample complexity bounds for (natural) actor-critic algorithms. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 4358–4369. Curran Associates, Inc., 2020. URL `https://proceedings.neurips.cc/paper/2020/file/2e1b24a664f5e9c18f407b2f9c73e821-Paper.pdf`.

Z. Yang, K. Zhang, M. Hong, and T. Başar. A finite sample analysis of the actor-critic algorithm. In *2018 IEEE conference on decision and control (CDC)*, pages 2759–2764. IEEE, 2018.

S. Zhang, B. Liu, and S. Whiteson. Mean-variance policy iteration for risk-averse reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 10905–10913, 2021.

A Proofs of the Main Results

A.1 Auxiliary Lemmas

Lemma 1. *For a generic random variable X with mean $\mathbb{E}[X]$ and sample mean $\hat{X} = \frac{1}{N}\sum_{i=1}^{N} X_i$, where $X_1 \dots X_N$ are i.i.d. copies of X, we have that*

$$Var[\hat{X}^2] = 4\,\mathbb{E}[X]^2\frac{\mu_2}{N} + \frac{2\mu_2^2 + 4\mu_3\,\mathbb{E}[X]}{N^2} + \frac{\mu_4 - 3\mu_2^2}{N^3},$$

where μ_i is $X's$ ith central moment defined as: $\mu_i = \mathbb{E}[(X - \mathbb{E}[X])^i]$.

Proof.

$$\begin{aligned}
\mathrm{Var}[\hat{X}^2] &= \mathbb{E}[\hat{X}^4] - \mathbb{E}[\hat{X}^2]^2 \\
&= \mathbb{E}[\hat{X}^4] - (\mathbb{E}[X]^2 + \mathrm{Var}[\hat{X}])^2 \\
&= \mathbb{E}[\hat{X}^4] - \mathbb{E}[X]^4 - 2\,\mathbb{E}[X]^2\mathrm{Var}[\hat{X}] - \mathrm{Var}[\hat{X}]^2 \\
&= \mathbb{E}[\hat{X}^4] - \mathbb{E}[X]^4 - 2\,\mathbb{E}[X]^2\frac{\mathrm{Var}[X]}{N} - \frac{\mathrm{Var}[X]^2}{N^2} \\
&= \mathbb{E}[\hat{X}^4] - \mathbb{E}[X]^4 - 2\,\mathbb{E}[X]^2\frac{\mu_2}{N} - \frac{\mu_2^2}{N^2}
\end{aligned}$$

From (Angelova, 2012), we have:

$$\mathbb{E}[\hat{X}^4] = \mathbb{E}[X]^4 + 6\,\mathbb{E}[X]^2\frac{\mu_2}{N} + \frac{3\mu_2^2 + 4\mu_3\,\mathbb{E}[X]}{N^2} + \frac{\mu_4 - 3\mu_2^2}{N^3}.$$

The result follows by plugging this back in the previous equation. □

Lemma 2. *Consider d real valued vectors $a_1, \dots, a_d \in \mathbb{R}^n$, we have that:*

(i) $\forall i, j \in \{1, \dots, d\}:\ |\langle a_i, b_j\rangle| \le \frac{1}{2}\left(\|a_i\|_2^2 + \|b_i\|_2^2\right)$.

(ii) $\left\|\sum_{i=1}^{d} a_i\right\|_2^2 \le d\sum_{i=1}^{d}\|a_i\|_2^2$

Proof. For any i, j we have:

$$\|a_i + a_j\|_2^2 = \|a_i\|_2^2 + \|a_j\|_2^2 + 2\langle a_i, a_j\rangle \ge 0, \ \text{ and } \ \|a_i - a_j\|_2^2 = \|a_i\|_2^2 + \|a_j\|_2^2 - 2\langle a_i, a_j\rangle \ge 0,$$

hence, trivially:

$$-\frac{1}{2}\|a_i\|_2^2 - \frac{1}{2}\|a_j\|_2^2 \le \langle a_i, a_j\rangle, \ \text{ and } \ \frac{1}{2}\|a_i\|_2^2 + \frac{1}{2}\|a_j\|_2^2 \ge \langle a_i, a_j\rangle,$$

which proves **(i)**.

By repeatedly applying **(i)** to the cross-terms of $\left\|\sum_{i=1}^{d} a_i\right\|_2^2$, we obtain:

$$\left\|\sum_{i=1}^{d} a_i\right\|_2^2 = \sum_{i=1}^{d}\|a_i\|_2^2 + 2\sum_{i>j}^{d}\langle a_i, a_j\rangle \le \sum_{i=1}^{d}\|a_i\|_2^2 + \sum_{i>j}^{d}\left(\|a_i\|_2^2 + \|a_j\|_2^2\right) = d\sum_{i=1}^{d}\|a_i\|_2^2,$$

since each index is counted $d-1$ times in the summation. □

Lemma 3. *Suppose A is an $n \times n$ invertible matrix, then*

$$\left\|A^{-1}\right\|_2 = \frac{1}{\min_i \sigma_i},$$

where, for a matrix, $\|\cdot\|_2$ denotes its spectral norm, and σ_i is the i^{th} singular value of A.

Proof. [12] By Theorem 4.3 in (Dahleh et al., 2004), we have that

$$\min_i \sigma_i = \inf_{x \neq 0} \frac{\|Ax\|_2}{\|x\|_2}.$$

And since A is invertible, $\min_i \sigma_i > 0$. We then have that

$$\begin{aligned}
\frac{1}{\min_i \sigma_i} &= \sup_{x \neq 0} \frac{\|x\|_2}{\|Ax\|_2} \\
&= \sup_{A^{-1}y \neq 0} \frac{\|A^{-1}y\|_2}{\|y\|_2} \\
&= \sup_{y \neq 0} \frac{\|A^{-1}y\|_2}{\|y\|_2} \\
&= \|A^{-1}\|_2,
\end{aligned}$$

where we have made the substitution $Ax = y$ and utilized the fact that $A^{-1}y = 0$ iff $y = 0$ since A is invertible. □

A.2 Analysing the Monte-Carlo Estimation of the Expected Return

In Subsection 3.3, we described the Monte-Carlo procedure that we adopted for estimating the expected return. The procedure is summarized in Algorithm 3. The idea is to average the (normalized) returns from L simulated

Algorithm 3 Monte-Carlo-J

Input: π, γ, L, T_J
Initialize: $\tilde{G}_0, \dots, \tilde{G}_{L-1} = 0$
for $i = 0, \dots, L-1$ **do**
  $s_0 \sim \mu_0(\cdot)$
  for $t = 0, \dots, T_J - 1$ **do**
    $a_t \sim \pi(s_t)$, $s_{t+1} \sim P(\cdot|s_t, a_t)$
    $\tilde{G}_i = \tilde{G}_i + \gamma^t R(s_t, a_t)$
  end for
end for
$\hat{J} = \frac{1}{L} \sum_{i=0}^{L-1} (1-\gamma) \tilde{G}_i$
Output: $\hat{J}$

trajectories each truncated at T_J steps. Our goal here is to derive finite-time bounds on $\mathbb{E}[(J - \hat{J})^2]$ and $\mathbb{E}[(J^2 - \hat{J}^2)^2]$, where J is the true expected return (as defined in (1)) and $\hat{J}$ is the estimated one[13]. That is, we want to bound these quantities in terms of L and T_J. These bounds shall be used when analyzing both the critic and the actor algorithms. We can start by defining the following quantities:

- $G_{0:T_J-1}$: a random variable representing the discounted sum of rewards from the beginning of a trajectory up to time $T_J - 1$ multiplied by a factor of $1 - \gamma$. That is:

$$G_{0:T_J-1} := (1-\gamma) \sum_{t=0}^{T_J-1} \gamma^t R(s_t, a_t).$$

  We denote its expected value by

$$\overline{G}_{0:T_J-1} := (1-\gamma) \mathop{\mathbb{E}}_{\substack{s_{t+1} \sim P(\cdot|s_t, a_t) \\ a_t \sim \pi(\cdot|s_t)}} \left[\sum_{t=0}^{T_J-1} \gamma^t R(s_t, a_t) \middle| s_0 \sim \mu(\cdot) \right].$$

  Note that we have, by Assumption 1.i, that $|\overline{G}_{0:T_J-1}| \leq (1 - \gamma^{T_J}) R_{\max} \leq R_{\max}$.

[12] The following proof is taken from `https://scicomp.stackexchange.com/q/10465`.
[13] To simplify the notation, we drop the dependence on the policy (i.e. π) since it is fixed in this setting.

- ζ_i: i$^{\text{th}}$ central moment[14] of $G_{0:T_J-1}$.
- $G_i : i = 0, \dots, L-1$: i.i.d. versions of $G_{0:T_J-1}$ corresponding to each of the L simulated trajectory. This way, we have that:
$$\hat{J} = \frac{1}{L}\sum_{i=0}^{L-1} G_i.$$
- $G_{T_J:\infty}$: a random variable representing the discounted sum of rewards collected starting from time T_J onward multiplied by a factor of $\gamma^{T_J}(1-\gamma)$. That is:
$$G_{T_J:\infty} := (1-\gamma)\gamma^{T_J}\sum_{t=0}^{\infty}\gamma^t R(s_{t+T_J}, a_{t+T_J}) = (1-\gamma)\sum_{t=T_J}^{\infty}\gamma^t R(s_t, a_t).$$
Its expected value can then be defined as:
$$\overline{G}_{T_J:\infty} := (1-\gamma)\mathop{\mathbb{E}}_{\substack{s_{t+1}\sim P(\cdot|s_t,a_t)\\ a_t\sim\pi(\cdot|s_t)}}\left[\sum_{t=T_J}^{\infty}\gamma^t R(s_t,a_t)\middle| s_{T_J}\sim\int_S p(s_0\xrightarrow{T_J}\cdot)\mu_0(ds_0)\right].$$
This way, we have that $J = \overline{G}_{0:T_J-1} + \overline{G}_{T_J:\infty}$. Also, by Assumption 1.i, $|\overline{G}_{T_J:\infty}| \le \gamma^{T_J} R_{\max} \le R_{\max}$.

The following simple lemma provides upper bounds on the second, third, and fourth central moments of $G_{0:T_J-1}$. We will need these bounds in the two forthcoming propositions.

Lemma 4. *Suppose that Assumption 1.i holds. With ζ_i denoting the i^{th} central moment of $G_{0:T_J-1}$, we have*

i. $\zeta_2 \le R_{max}^2$.

ii. $|\zeta_3| \le \frac{4\sqrt{3}}{9}R_{max}^3 \le R_{max}^3$.

iii. $\zeta_4 \le \frac{4}{3}R_{max}^4 \le 2R_{max}^4$.

Proof. For a random variable X upper-bounded by M and lower bounded by m, with μ_i denoting its ith central moment, we have by Popoviciu's inequality that
$$\mu_2 \le \frac{(M-m)^2}{4}.$$
Thus, we have that
$$\zeta_2 \le \frac{(2R_{\max})^2}{4} = R_{\max}^2,$$
since we can take $M = R_{\max}$ and $m = -R_{\max}$, proving the first item. For the second item we have from Theorem (2.3) in (Sharma et al., 2015) that
$$|\mu_3| \le \frac{(M-m)^3}{6\sqrt{3}}.$$
Which means that in our case we shall have that
$$|\zeta_3| \le \frac{(2R_{\max})^3}{6\sqrt{3}} = \frac{8R_{\max}^3}{6\sqrt{3}} = \frac{4\sqrt{3}}{9}R_{\max}^3 \le R_{\max}^3.$$
Finally, for the third item, Theorem (2.1) in (Sharma et al., 2015) states that
$$\mu_4 \le \frac{(M-m)^4}{12}.$$
And for us,
$$\zeta_4 \le \frac{(2R_{\max})^4}{12} \le \frac{4}{3}R_{\max}^4 \le 2R_{\max}^4.$$
□

[14]For a random variable X, its i$^{\text{th}}$ central moment is $\mathbb{E}[(X - \mathbb{E}[X])^i]$. Note that the second central moment of X is its variance.

Proposition 1. *Suppose, for a given policy, an estimate $\hat{J}$ is obtained using Algorithm 3, and that Assumption 1.i holds, then we have*

$$\mathbb{E}[(J-\hat{J})^2] \leq \gamma^{2T_J} R_{\max}^2 + \frac{R_{\max}^2}{L}.$$

Proof. We begin with a bias-variance decomposition:

$$\begin{aligned}\mathbb{E}[(J-\hat{J})^2] &= \mathbb{E}[(J-\mathbb{E}[\hat{J}]+\mathbb{E}[\hat{J}]-\hat{J})^2] \\ &= \mathbb{E}[(J-\mathbb{E}[\hat{J}])^2] + \mathbb{E}[(\mathbb{E}[\hat{J}]-\hat{J})^2] \\ &= (J-\mathbb{E}[\hat{J}])^2 + Var(\hat{J}),\end{aligned}$$

where the second equality holds since $2(J-\mathbb{E}[\hat{J}])\,\mathbb{E}[(\mathbb{E}[\hat{J}]-\hat{J})] = 0$. The first term represents the (squared) bias of $\hat{J}$, which, in general, is not zero since we are using truncated returns. Since $\hat{J}$ is an average of instances of $G_{0:T_J-1}$, its expected value is the same as that of $G_{0:T_J-1}$, which is $\overline{G}_{0:T_J-1}$. Moreover, we remarked earlier that $J = \overline{G}_{0:T_J-1} + \overline{G}_{T_J:\infty}$, this then means that

$$|J-\mathbb{E}[\hat{J}]| = |J-\overline{G}_{0:T_J-1}| = |\overline{G}_{T_J:\infty}| \leq \gamma^{T_J} R_{max},$$

Thus, $(J-\mathbb{E}[\hat{J}])^2 \leq \gamma^{2T_J} R_{\max}^2$. As for the variance, since $\hat{J}$ is the mean of L samples of $\overline{G}_{0:T_J-1}$, then $Var(\hat{J}) = \frac{\zeta_2}{L}$. Combining both terms and applying Lemma 4.i, we get

$$\mathbb{E}[(J-\hat{J})^2] \leq \gamma^{2T_J} R_{\max}^2 + \frac{\zeta_2}{L} \leq \gamma^{2T_J} R_{\max}^2 + \frac{R_{\max}^2}{L}.$$

□

Proposition 2. *Suppose, for a given policy, an estimate $\hat{J}$ is obtained using Algorithm 3, and that Assumption 1.i holds, then we have*

$$\begin{aligned}\mathbb{E}[(J^2-\hat{J}^2)^2] &\leq 4R_{\max}^4\gamma^{2T_J} + 4R_{\max}^2\frac{\zeta_2}{L} + \frac{3\zeta_2^2 + 4|\zeta_3|R_{\max}}{L^2} + \frac{\zeta_4 - 3\zeta_2^2}{L^3} \\ &\leq 4R_{\max}^4\gamma^{2T_J} + R_{\max}^4\left(\frac{4}{L} + \frac{7}{L^2} + \frac{5}{L^3}\right).\end{aligned}$$

Proof. We, again, start with a bias-variance decomposition:

$$\begin{aligned}\mathbb{E}[(J^2-\hat{J}^2)^2] &= \mathbb{E}[(J^2-\mathbb{E}[\hat{J}^2]+\mathbb{E}[\hat{J}^2]-\hat{J}^2)^2] \\ &= \mathbb{E}[(J^2-\mathbb{E}[\hat{J}^2])^2] + \mathbb{E}[(\mathbb{E}[\hat{J}^2]-\hat{J}^2)^2] \\ &= (J^2-\mathbb{E}[\hat{J}^2])^2 + Var(\hat{J}^2).\end{aligned}$$

Starting with the bias term, we proceed as follows:

$$\begin{aligned}|J^2-\mathbb{E}[\hat{J}^2]| &= \left|J^2-\mathbb{E}[\hat{J}]^2 - \frac{\zeta_2}{L}\right| \\ &\leq |J^2-\mathbb{E}[\hat{J}]^2| + \frac{\zeta_2}{L} \\ &= \left|(\overline{G}_{0:T_J-1}+\overline{G}_{T_J:\infty})^2 - \overline{G}_{0:T_J-1}^2\right| + \frac{\zeta_2}{L} \\ &= \left|\overline{G}_{T_J:\infty}(2\overline{G}_{0:T_J-1}+\overline{G}_{T_J:\infty})\right| + \frac{\zeta_2}{L} \\ &\leq \gamma^{T_J}R_{\max}\left(2(1-\gamma^{T_J})R_{\max} + \gamma^{T_J}R_{\max}\right) + \frac{\zeta_2}{L} \\ &= \gamma^{T_J}(2-\gamma^{T_J})R_{\max}^2 + \frac{\zeta_2}{L} \\ &\leq 2\gamma^{T_J}R_{\max}^2 + \frac{\zeta_2}{L},\end{aligned}$$

where the first equality holds since, for a random variable X, $\mathbb{E}[X^2] = \mathbb{E}[X]^2 + Var(X)$. We then have that

$$(J^2 - \mathbb{E}[\hat{J}^2])^2 \leq (2\gamma^{T_J} R_{\max}^2 + \frac{\zeta_2}{L})^2 = 4\gamma^{2T_J} R_{\max}^4 + 4\gamma^{T_J} R_{\max}^2 \frac{\zeta_2}{L} + \frac{\zeta_2^2}{L^2}.$$

For the variance, we apply Lemma 1:

$$Var(\hat{J}^2) = 4\overline{G}_{0:T_J-1}^2 \frac{\zeta_2}{L} + \frac{2\zeta_2^2 + 4\zeta_3 \overline{G}_{0:T_J-1}}{L^2} + \frac{\zeta_4 - 3\zeta_2^2}{L^3}.$$

Putting everything together, we have

$$\begin{aligned}
\mathbb{E}[(J^2 - \hat{J}^2)^2] &\leq 4\gamma^{2T_J} R_{\max}^4 + 4\gamma^{T_J} R_{\max}^2 \frac{\zeta_2}{L} + \frac{\zeta_2^2}{L^2} \\
&\quad + 4\overline{G}_{0:T_J-1}^2 \frac{\zeta_2}{L} + \frac{2\zeta_2^2 + 4\zeta_3 \overline{G}_{0:T_J-1}}{L^2} + \frac{\zeta_4 - 3\zeta_2^2}{L^3} \\
&\leq 4\gamma^{2T_J} R_{\max}^4 + 4(\overline{G}_{0:T_J-1}^2 + \gamma^{T_J} R_{\max}^2) \frac{\zeta_2}{L} + \frac{3\zeta_2^2 + 4|\zeta_3| R_{\max}}{L^2} + \frac{\zeta_4 - 3\zeta_2^2}{L^3} \\
&\leq 4\gamma^{2T_J} R_{\max}^4 + 4R_{\max}^2((1-\gamma^{T_J})^2 + \gamma^{T_J}) \frac{\zeta_2}{L} + \frac{3\zeta_2^2 + 4|\zeta_3| R_{\max}}{L^2} + \frac{\zeta_4 - 3\zeta_2^2}{L^3} \\
&= 4\gamma^{2T_J} R_{\max}^4 + 4R_{\max}^2(1 + \gamma^{2T_J} - \gamma^{T_J}) \frac{\zeta_2}{L} + \frac{3\zeta_2^2 + 4|\zeta_3| R_{\max}}{L^2} + \frac{\zeta_4 - 3\zeta_2^2}{L^3} \\
&\leq 4\gamma^{2T_J} R_{\max}^4 + 4R_{\max}^2 \frac{\zeta_2}{L} + \frac{3\zeta_2^2 + 4|\zeta_3| R_{\max}}{L^2} + \frac{\zeta_4 - 3\zeta_2^2}{L^3}.
\end{aligned}$$

Furthermore, we can apply Lemma 4 and get

$$\begin{aligned}
\mathbb{E}[(J^2 - \hat{J}^2)^2] &\leq 4\gamma^{2T_J} R_{\max}^4 + \frac{4R_{\max}^4}{L} + \frac{3R_{\max}^4 + 4R_{\max}^4}{L^2} + \frac{2R_{\max}^4 + 3R_{\max}^4}{L^3} \\
&= 4\gamma^{2T_J} R_{\max}^4 + \left(\frac{4}{L} + \frac{7}{L^2} + \frac{5}{L^3}\right) R_{\max}^4.
\end{aligned}$$

□

A.3 Proof of the Critic's Bound

In this section, we develop the proof of Theorem 1, which provided a finite time bound for the direct mini-batch TD algorithm[15]. In Section 4.1, we defined the two quantities A and $b(\hat{J})$, where $\hat{J}$ is some estimate of the expected return. These quantities are relevant for describing the expected behaviour of the algorithm at steady-state, and characterizing its fixed point. Analogously, we can define the following quantities (for the i^{th} sample in the mini-batch of the t^{th} iteration) $b_{i,t}(\hat{J}) := \phi(s_{i,t}) R^{\lambda}(s_{i,t}, a_{i,t}, \hat{J})$, and $A_{i,t} := \phi(s_{i,t})(\gamma\phi(s_{i,t+1}) - \phi(s_{i,t}))^{\intercal}$. Using these definitions, the update rule at the i^{th} iteration of the algorithm can be written as

$$\omega_{i+1} = \omega_i + \beta\left(\frac{1}{M}\sum_{t=0}^{M-1} b_{i,t}(\hat{J}) + \frac{1}{M}\sum_{t=0}^{M-1} A_{i,t}\,\omega_i\right).$$

To be able to leverage the risk-neutral results in (Xu et al., 2020), we make the following assumption:

Assumption 4. *For any triple $(s_{i,t}, a_{i,t}, s_{i,t+1}) \in \mathcal{S} \times \mathcal{A} \times \mathcal{S}$ and any $\hat{J}$ estimate bounded, in absolute value, by $R_{\max}$, there exists real constants C_A and C_b such that $\|A_{i,t}\|_F \leq C_A$ and $\|b_{i,t}(\hat{J})\|_2 \leq C_b$, where $\|\cdot\|_F$ is the Frobenius norm[16] of a matrix.*

While stated as an assumption, the previous statement is justified since Assumptions 1.i and 2 state that, respectively, the reward function and the norm of the feature vectors are bounded. Also, note that $\hat{J}$ is bounded,

[15] That is, Algorithm 1 with $f_R(s,a) = R(s,a) - \lambda(R(s,a) - \hat{J})^2$.

[16] For an $m \times n$ matrix X, its Frobenius norm (Golub and Van Loan, 1996) is defined as $\|X\|_F := \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2} = \sqrt{\sum_{i=1}^{\min\{m,n\}}\sigma_i^2(A)}$, where $\sigma_i(A)$ are the singular value of A.

in absolute value, by $R_{\max}$ if it is learned using Algorithm 3. Our assumptions should also guarantee that for any such $\hat{J}$, $\|\omega^*_{\hat{J}}\|_2$ is uniformly upper bounded by some positive constant C_ω. To see this, first note that $\|\omega^*_{\hat{J}}\|_2 = \|A^{-1}b(\hat{J})\|_2$. For an $n \times m$ matrix A, its spectral norm (or induced l2 norm) is defined as $\|A\|_2 = \sup_{x \neq 0} \frac{\|Ax\|_2}{\|x\|_2}$. This means that $\|Ax\|_2 \leq \|A\|_2 \|x\|_2$, for any vector $x \in \mathbb{R}^m$. For us, this means that $\|A^{-1}b(\hat{J})\|_2 \leq \|A^{-1}\|_2 \|b(\hat{J})\|_2 \leq \frac{C_b}{\bar{\sigma}} := C_\omega$, where we denote by $\bar{\sigma}$ the smallest singular value of A, and the last inequality follows by Assumption 4 and Lemma 3. Note that, unlike C_A and C_b, the matrix A and consequently $\bar{\sigma}$ and C_ω all depend on the policy π_θ under evaluation. We don't make this dependence explicit in the notation to keep it simple. However, since the critic's bound is expressed in terms of these quantities, we need to define two further quantities in order to make the bound of critic meaningful for the analysis of the full algorithm later on. Namely, define $\tilde{\sigma} := \inf_\theta \bar{\sigma}_\theta$ and $\tilde{C}_\omega := \frac{C_b}{\tilde{\sigma}}$, assuming of course that (for our policy class Π) $\tilde{\sigma} > 0$. Finally, the following assumption serves to simplify the expressions of the bounds, without loss of generality[17].

Assumption 5. ***(i)*** $C_\psi = 1$. ***(ii)*** $C_\phi = 1$.

In our proof, we will make use of the critic bound of (Xu et al., 2020) for the risk-neutral case. We will not reiterate their proof here, but we can briefly mention the main idea of their approach. Suppose we are at iteration $t+1$ of the critic's algorithm, they start by bounding[18] $\mathbb{E}[\|\omega_{t+1} - \omega^*\|_2^2]$ in terms of the expected error at the previous iteration (i.e. $\mathbb{E}[\|\omega_t - \omega^*\|_2^2]$) and the expected[19] value of the squared norm of the difference between the performed update from ω_t to ω_{t+1} and its expected value at steady state. The latter quantity is then bounded in terms of the mixing properties of the MDP (see Assumption 3), the mini-batch size M, and again, $\mathbb{E}[\|\omega_t - \omega^*\|_2^2]$. By recursively repeating the same analysis on $\mathbb{E}[\|\omega_t - \omega^*\|_2^2]$ and the resulting terms, they obtain the following bound (under some conditions that we will mention later on the mini-batch size and the step-size):

$$\mathbb{E}[\|\omega_{T_c} - \omega^*\|_2^2] \leq \left(1 - \frac{\chi_A}{8}\beta\right)^{T_c} \|\omega_0 - \omega^*\|_2^2 + \left(\frac{2}{\chi_A} + 2\beta\right) \frac{192(C_A^2 C_\omega^2 + C_b^2)[1 + (\kappa - 1)\rho]}{(1-\rho)\chi_A M}.$$

The first term, which depends on the initial value of the parameter vector, decays geometrically with the number of iteration. The second term decays with a rate of $\frac{1}{M}$, where M is the size of the mini-batch of samples used at each iteration. Note that the step-size is kept constant across the iterations.

As we discussed in the main paper, if we imagine using the reward transformation performed using some *fixed* estimate $\hat{J}$ of the expected return, the problem at hand is akin to the risk neutral one, just with a different reward function. We can then directly leverage a form of the bound above to establish the convergence rate to the TD fixed point under the this reward transformation (i.e. to $\omega^*_{\hat{J}}$). The proof of the next theorem uses this idea, along with (among other things) the bounds in Propositions 1 and 2, to establish the convergence rate of the critic to the true fixed point ω^*_J (i.e. under the true reward transformation) when $\hat{J}$ is learned using Algorithm 3. The following theorem is a restatement of Theorem 1 which presents the bound in a explicit form. That the algorithm has the sample complexity stated in Theorem 1 is demonstrated in a designated corollary.

Theorem 3 (Explicit Statement of the Bound in Theorem 1)**.** *Suppose Assumptions 1 to 5 hold, and suppose we are given a policy π_θ (with normalized expected return J) and risk parameter λ. Suppose that a Monte-Carlo estimate $\hat{J}$ is obtained for π_θ using Algorithm 3, and then Algorithm 1 is run for T_c steps with $f_R(s,a) = R(s,a) - \lambda(R(s,a) - \hat{J})^2$. Then, for $M \geq \left(\frac{2}{\chi_A} + 2\beta\right) \frac{192 C_A^2 [1 + (\kappa - 1)\rho]}{(1-\rho)\chi_A}$ and $\beta \leq \min\left\{\frac{\chi_A}{8C_A^2}, \frac{4}{\chi_A}\right\}$, we have that*

$$\begin{aligned}
\mathbb{E}\left[\left\|\omega^{\hat{J}}_{T_c} - \omega^*_J\right\|_2^2\right] \leq\ & 4\|\omega_0 - \omega^*_J\|_2^2 \left(1 - \frac{\chi_A}{8}\beta\right)^{T_c} \\
& + \left(\frac{2}{\chi_A} + 2\beta\right) \frac{384(C_A^2 C_\omega^2 + C_b^2)[1 + (\kappa - 1)\rho]}{(1-\rho)\chi_A M} \\
& + \frac{2}{\bar{\sigma}^2}\left[1 + 2\left(1 - \frac{\chi_A}{8}\beta\right)^{T_c}\right] \xi_J,
\end{aligned}$$

where $\omega^{\hat{J}}_{T_c}$ is the parameter vector obtained after T_c iterations of the algorithm while using $\hat{J}$ to perform the

[17] C_ψ and C_ϕ were introduced in Assumptions 1 and 2.

[18] We remind the reader that, unlike in (Xu et al., 2020), we use ω for the critic's parameters and the more common choice of θ for the policy's parameters.

[19] Note that the performed updates are random due to the stochasticity of the sampling process.

reward transformation, $\xi_J := 2\lambda^2 R_{\max}^4\big(8\gamma^{2T_J} + \frac{8}{L} + \frac{7}{L^2} + \frac{5}{L^3}\big)$*,* $\bar{\sigma}$ *is the smallest singular value of the matrix* A*, and the expectation is over both the Monte-Carlo estimation of* $\hat{J}$ *and the TD algorithm.*

Proof. We begin by adding and subtracting $\omega^*_{\hat{J}}$, which is the TD fixed point when using $\hat{J}$. Note that, at this point, $\omega^*_{\hat{J}}$ is a random variable due to its dependence on the estimator $\hat{J}$.

$$\begin{aligned}\mathbb{E}\Big[\Big\|\omega^{\hat{J}}_{T_c} - \omega^*_J\Big\|_2^2\Big] &= \mathbb{E}\Big[\Big\|\omega^{\hat{J}}_{T_c} - \omega^*_{\hat{J}} + \omega^*_{\hat{J}} - \omega^*_J\Big\|_2^2\Big] \\ &\leq 2\,\mathbb{E}\Big[\Big\|\omega^{\hat{J}}_{T_c} - \omega^*_{\hat{J}}\Big\|_2^2\Big] + 2\,\mathbb{E}\Big[\big\|\omega^*_{\hat{J}} - \omega^*_J\big\|_2^2\Big]. \end{aligned} \tag{9}$$

where the inequality follows from Lemma 2.ii. Focusing on the first term, we have

$$\mathbb{E}\Big[\Big\|\omega^{\hat{J}}_{T_c} - \omega^*_{\hat{J}}\Big\|_2^2\Big] = \mathbb{E}\Big[\mathbb{E}\Big[\Big\|\omega^{\hat{J}}_{T_c} - \omega^*_{\hat{J}}\Big\|_2^2\Big|\hat{J}\Big]\Big]. \tag{10}$$

For the inner expectation, as remarked before, we can apply the risk-neutral bound from theorem 4 in (Xu et al., 2020). Namely for $M \geq \left(\frac{2}{\chi_A} + 2\beta\right)\frac{192C_A^2[1+(\kappa-1)\rho]}{(1-\rho)\chi_A}$ and $\beta \leq \min\Big\{\frac{\chi_A}{8C_A^2}, \frac{4}{\chi_A}\Big\}$, we have

$$\mathbb{E}\Big[\Big\|\omega^{\hat{J}}_{T_c} - \omega^*_{\hat{J}}\Big\|_2^2\Big|\hat{J}\Big] \leq \Big(1 - \frac{\chi_A}{8}\beta\Big)^{T_c}\|\omega_0 - \omega^*_{\hat{J}}\|_2^2 + \left(\frac{2}{\chi_A} + 2\beta\right)\frac{192(C_A^2C_\omega^2 + C_b^2)[1+(\kappa-1)\rho]}{(1-\rho)\chi_A M},$$

where C_A, C_b, and C_ω have been defined in Assumption 4 and the paragraph that followed. Note that $\|\omega_0 - \omega^*_{\hat{J}}\|_2^2$ is the only part that depends on $\hat{J}$ in the previous bound. Plugging back in (10), we get that

$$\begin{aligned}\mathbb{E}\Big[\Big\|\omega^{\hat{J}}_{T_c} - \omega^*_{\hat{J}}\Big\|_2^2\Big] &\leq \Big(1 - \frac{\chi_A}{8}\beta\Big)^{T_c}\mathbb{E}\Big[\|\omega_0 - \omega^*_{\hat{J}}\|_2^2\Big] + \left(\frac{2}{\chi_A} + 2\beta\right)\frac{192(C_A^2C_\omega^2 + C_b^2)[1+(\kappa-1)\rho]}{(1-\rho)\chi_A M} \\ &\leq \Big(1 - \frac{\chi_A}{8}\beta\Big)^{T_c}\mathbb{E}\Big[\|\omega_0 - \omega^*_J + \omega^*_J - \omega^*_{\hat{J}}\|_2^2\Big] + \left(\frac{2}{\chi_A} + 2\beta\right)\frac{192(C_A^2C_\omega^2 + C_b^2)[1+(\kappa-1)\rho]}{(1-\rho)\chi_A M} \\ &\leq 2\Big(1 - \frac{\chi_A}{8}\beta\Big)^{T_c}\|\omega_0 - \omega^*_J\|_2^2 + \left(\frac{2}{\chi_A} + 2\beta\right)\frac{192(C_A^2C_\omega^2 + C_b^2)[1+(\kappa-1)\rho]}{(1-\rho)\chi_A M} \\ &\quad + 2\Big(1 - \frac{\chi_A}{8}\beta\Big)^{T_c}\mathbb{E}\Big[\big\|\omega^*_{\hat{J}} - \omega^*_J\big\|_2^2\Big],\end{aligned}$$

where the last inequality again follows from Lemma 2.ii. Plugging back in (9), we get that

$$\begin{aligned}\mathbb{E}\Big[\Big\|\omega^{\hat{J}}_{T_c} - \omega^*_J\Big\|_2^2\Big] &\leq 4\Big(1 - \frac{\chi_A}{8}\beta\Big)^{T_c}\|\omega_0 - \omega^*_J\|_2^2 + \left(\frac{2}{\chi_A} + 2\beta\right)\frac{384(C_A^2C_\omega^2 + C_b^2)[1+(\kappa-1)\rho]}{(1-\rho)\chi_A M} \\ &\quad + \left[2 + 4\Big(1 - \frac{\chi_A}{8}\beta\Big)^{T_c}\right]\mathbb{E}\Big[\big\|\omega^*_{\hat{J}} - \omega^*_J\big\|_2^2\Big].\end{aligned} \tag{11}$$

Thus, we need to bound $\mathbb{E}\Big[\Big\|\omega^*_{\hat{J}} - \omega^*_J\Big\|_2^2\Big]$. We proceed as follows[20]:

$$\begin{aligned}\mathbb{E}\Big[\big\|\omega^*_{\hat{J}} - \omega^*_J\big\|_2^2\Big] &= \mathbb{E}\Big[\Big\|A^{-1}b(J) - A^{-1}b(\hat{J})\Big\|_2^2\Big] \\ &= \mathbb{E}\Big[\Big\|A^{-1}\Big(b(J) - b(\hat{J})\Big)\Big\|_2^2\Big] \\ &\leq \frac{1}{\bar{\sigma}^2}\,\mathbb{E}\Big[\Big\|b(J) - b(\hat{J})\Big\|_2^2\Big],\end{aligned} \tag{12}$$

where $\bar{\sigma}$ is the smallest singular value of A, and the last inequality holds since, as demonstrated before, for an $m \times n$ matrix X and a vector $y \in \mathbb{R}^n$, $\|Xy\|_2^2 \leq \|X\|_2^2\|y\|_2^2$, where $\|X\|_2$ is the spectral norm of X. Furthermore,

[20]Note that our assumptions ensure that A is negative definite (Tsitsiklis and Van Roy, 1997), and the existence and uniqueness of $\omega^*_{\hat{J}}$ (for any fixed $\hat{J}$) and ω^*_J.

we used that, by Lemma 3, $\left\|A^{-1}\right\|_2 = \frac{1}{\bar{\sigma}}$. Moving on, recall that μ_θ is the stationary distribution of the MDP when using policy π_θ.

Define $R^\lambda(s, a, \hat{J}) \coloneqq R(s, a) - \lambda(R(s, a) - \hat{J})^2$. We then have that

$$\begin{aligned}
\mathbb{E}\Big[\Big\|b(J) - b(\hat{J})\Big\|\Big]_2^2 &= \mathbb{E}\left[\left\|\mathop{\mathbb{E}}_{\mu_\theta}\big[\phi(s_t)R^\lambda(s_t, a_t, J)\big] - \mathop{\mathbb{E}}_{\mu_\theta}\Big[\phi(s_t)R^\lambda(s_t, a_t, \hat{J})\Big]\right\|_2^2\right] \\
&= \mathbb{E}\left[\left\|\mathop{\mathbb{E}}_{\mu_\theta}\Big[\phi(s_t)\Big(R^\lambda(s_t, a_t, J) - R^\lambda(s_t, a_t, \hat{J})\Big)\Big]\right\|_2^2\right] \\
&= \mathbb{E}\left[\left\|\mathop{\mathbb{E}}_{\mu_\theta}\Big[\phi(s_t)\Big(2\lambda R(s_t, a_t)\Big(J - \hat{J}\Big) + \lambda\Big(\hat{J}^2 - J^2\Big)\Big)\Big]\right\|_2^2\right] \\
&= \mathbb{E}\left[\left\|2\lambda\Big(J - \hat{J}\Big)\mathop{\mathbb{E}}_{\mu_\theta}[\phi(s_t)R(s_t, a_t)] + \lambda\Big(\hat{J}^2 - J^2\Big)\mathop{\mathbb{E}}_{\mu_\theta}[\phi(s_t)]\right\|_2^2\right] \\
&\leq \mathbb{E}\left[2\left\|2\lambda\Big(J - \hat{J}\Big)\mathop{\mathbb{E}}_{\mu_\theta}[\phi(s_t)R(s_t, a_t)]\right\|_2^2 + 2\left\|\lambda\Big(\hat{J}^2 - J^2\Big)\mathop{\mathbb{E}}_{\mu_\theta}[\phi(s_t)]\right\|_2^2\right] \\
&= 8\lambda^2\,\mathbb{E}\left[\Big(J - \hat{J}\Big)^2\left\|\mathop{\mathbb{E}}_{\mu_\theta}[\phi(s_t)R(s_t, a_t)]\right\|_2^2\right] + 2\lambda^2\,\mathbb{E}\left[\Big(\hat{J}^2 - J^2\Big)^2\left\|\mathop{\mathbb{E}}_{\mu_\theta}[\phi(s_t)]\right\|_2^2\right] \\
&\leq 8\lambda^2 R_{\max}^2\,\mathbb{E}\left[\Big(J - \hat{J}\Big)^2\right] + 2\lambda^2\,\mathbb{E}\left[\Big(\hat{J}^2 - J^2\Big)^2\right], \qquad (13)
\end{aligned}$$

where the first inequality follows from Lemma 2, and the last inequality follows (keeping in mind Assumptions 1.i, 2, and 5) since

$$\left\|\mathop{\mathbb{E}}_{\mu_\theta}[\phi(s_t)R(s_t, a_t)]\right\|_2^2 \leq \mathop{\mathbb{E}}_{\mu_\theta}\Big[\|\phi(s_t)R(s_t, a_t)\|_2^2\Big] \leq R_{\max}^2,$$

and

$$\left\|\mathop{\mathbb{E}}_{\mu_\theta}[\phi(s_t)]\right\|_2^2 \leq \mathop{\mathbb{E}}_{\mu_\theta}\Big[\|\phi(s_t)\|_2^2\Big] \leq 1.$$

Now, we can plug the results of Propositions 1 and 2 in inequality (13) to get that

$$\begin{aligned}
\mathbb{E}\Big[\Big\|b(J) - b(\hat{J})\Big\|\Big]_2^2 &\leq 8\lambda^2 R_{\max}^2\,\mathbb{E}\left[\Big(J - \hat{J}\Big)^2\right] + 2\lambda^2\,\mathbb{E}\left[\Big(\hat{J}^2 - J^2\Big)^2\right] \\
&\leq 8\lambda^2 R_{\max}^2\left(\gamma^{2T_J}R_{\max}^2 + \frac{R_{\max}^2}{L}\right) + 2\lambda^2\left(4\gamma^{2T_J}R_{\max}^4 + \left(\frac{4}{L} + \frac{7}{L^2} + \frac{5}{L^3}\right)R_{\max}^4\right) \\
&= 8\lambda^2\gamma^{2T_J}R_{\max}^4 + \frac{8\lambda^2}{L}R_{\max}^4 + 8\lambda^2\gamma^{2T_J}R_{\max}^4 + 2\lambda^2\left(\frac{4}{L} + \frac{7}{L^2} + \frac{5}{L^3}\right)R_{\max}^4 \\
&= 16\lambda^2\gamma^{2T_J}R_{\max}^4 + 2\lambda^2\left(\frac{8}{L} + \frac{7}{L^2} + \frac{5}{L^3}\right)R_{\max}^4.
\end{aligned}$$

Plugging back in (12), we get

$$\mathbb{E}\Big[\big\|\omega_{\hat{J}}^* - \omega_J^*\big\|_2^2\Big] \leq \frac{2\lambda^2}{\bar{\sigma}^2}\left(8\gamma^{2T_J}R_{\max}^4 + \left(\frac{8}{L} + \frac{7}{L^2} + \frac{5}{L^3}\right)R_{\max}^4\right) = \frac{\xi_J}{\bar{\sigma}^2},$$

where $\xi_J \coloneqq 2\lambda^2 R_{\max}^4\big(8\gamma^{2T_J} + \frac{8}{L} + \frac{7}{L^2} + \frac{5}{L^3}\big)$. We can now plug back the last result into (11) to get the desired

bound:

$$\mathbb{E}\left[\left\|\omega_{T_c}^{\hat{J}} - \omega_J^*\right\|_2^2\right] \leq 4\|\omega_0 - \omega_J^*\|_2^2\Big(1 - \frac{\chi_A}{8}\beta\Big)^{T_c} + \left(\frac{2}{\chi_A} + 2\beta\right)\frac{384(C_A^2 C_\omega^2 + C_b^2)[1 + (\kappa - 1)\rho]}{(1-\rho)\chi_A M} + \frac{2}{\bar{\sigma}^2}\left[1 + 2\Big(1 - \frac{\chi_A}{8}\beta\Big)^{T_c}\right]\xi_J.$$

□

Corollary 3.1 (Restatement of the Sample Complexity Result in Theorem 1)**.** *Suppose we are again in the same setting of Theorem 3, and suppose the assumptions mentioned therein hold. Then, for a sufficiently small* $\epsilon > 0$*, if* $\beta \leq \min\left\{\frac{\chi_A}{8C_A^2}, \frac{4}{\chi_A}\right\}$*, and*

- $T_J \geq \frac{\log\left(\frac{192\lambda^2 R_{\max}^4}{\epsilon\bar{\sigma}^2}\right)}{2(1-\gamma)}$,
- $L \geq \frac{576\lambda^2 R_{\max}^4}{\epsilon\bar{\sigma}^2}$,
- $T_c \geq \frac{8\log\left(\frac{24}{\epsilon}\|\omega_0 - \omega_J^*\|_2^2\right)}{\chi_A\beta}$,
- $M \geq \left(\frac{2}{\chi_A} + 2\beta\right)\frac{2304(C_A^2 C_\omega^2 + C_b^2)[1+(\kappa-1)\rho]}{(1-\rho)\chi_A\epsilon}$,

then

$$\mathbb{E}\left[\left\|\omega_{T_c}^{\hat{J}} - \omega_J^*\right\|_2^2\right] \leq \epsilon,$$

and the total sample complexity is

$$T_c M + L T_J = \mathcal{O}\big(\epsilon^{-1}\log\big(\epsilon^{-1}\big)\big).$$

Proof. By expanding and rearranging the bound in Theorem 3, we have that

$$\begin{aligned}\mathbb{E}\left[\left\|\omega_{T_c}^{\hat{J}} - \omega_J^*\right\|_2^2\right] \leq\ & 4\|\omega_0 - \omega_J^*\|_2^2\Big(1 - \frac{\chi_A}{8}\beta\Big)^{T_c} \\ &+ \left(\frac{2}{\chi_A} + 2\beta\right)\frac{384(C_A^2 C_\omega^2 + C_b^2)[1 + (\kappa - 1)\rho]}{(1-\rho)\chi_A M} \\ &+ \frac{32\lambda^2 R_{\max}^4}{\bar{\sigma}^2}\gamma^{2T_J} \\ &+ \frac{4\lambda^2 R_{\max}^4}{\bar{\sigma}^2}\left(\frac{8}{L} + \frac{7}{L^2} + \frac{5}{L^3}\right) \\ &+ \frac{64\lambda^2 R_{\max}^4}{\bar{\sigma}^2}\gamma^{2T_J}\Big(1 - \frac{\chi_A}{8}\beta\Big)^{T_c} \\ &+ \frac{8\lambda^2 R_{\max}^4}{\bar{\sigma}^2}\left(\frac{8}{L} + \frac{7}{L^2} + \frac{5}{L^3}\right)\Big(1 - \frac{\chi_A}{8}\beta\Big)^{T_c}.\end{aligned}$$

Note that $\left(1 - \frac{\chi_A}{8}\beta\right)^{T_c} \leq e^{-\frac{\chi_A}{8}\beta T_c}$. This holds since $(1-x) \leq e^{-x}$, and if $x \leq 1$, then $(1-x)^r \leq e^{-rx}$ for $r \geq 0$. The claim then follows since $\beta < \frac{8}{\chi_A}$ and $T_c \geq 0$. By a similar argument, $\gamma^{2T_J} = (1 - (1-\gamma))^{2T_J} \leq e^{-2(1-\gamma)T_J}$.

Plugging back these bounds, we get

$$\begin{aligned}\mathbb{E}\left[\left\|\omega_{T_c}^{\hat{J}}-\omega_J^*\right\|_2^2\right] \leq\ & 4\|\omega_0-\omega_J^*\|_2^2\, e^{-\frac{\chi_A}{8}\beta T_c} \\ &+\left(\frac{2}{\chi_A}+2\beta\right)\frac{384(C_A^2C_\omega^2+C_b^2)[1+(\kappa-1)\rho]}{(1-\rho)\chi_A M} \\ &+\frac{32\lambda^2R_{\max}^4}{\bar{\sigma}^2}\, e^{-2(1-\gamma)T_J} \\ &+\frac{4\lambda^2R_{\max}^4}{\bar{\sigma}^2}\left(\frac{8}{L}+\frac{7}{L^2}+\frac{5}{L^3}\right) \\ &+\frac{64\lambda^2R_{\max}^4}{\bar{\sigma}^2}\, e^{-2(1-\gamma)T_J}\, e^{-\frac{\chi_A}{8}\beta T_c} \\ &+\frac{8\lambda^2R_{\max}^4}{\bar{\sigma}^2}\left(\frac{8}{L}+\frac{7}{L^2}+\frac{5}{L^3}\right) e^{-\frac{\chi_A}{8}\beta T_c}.\end{aligned}$$

To bound the whole expression by ϵ, we can bound each of the six terms by $\frac{\epsilon}{6}$. By rearranging each of the resulting inequalities, we obtain, by the conditions on the parameters indicated in the statement, the desired error, provided that ϵ is sufficiently small. Thus, the sample complexity is given by

$$T_cM+LT_J=\mathcal{O}\left(\frac{1}{\epsilon}\log\left(\frac{1}{\epsilon}\right)\right)+\mathcal{O}\left(\frac{1}{\epsilon}\log\left(\frac{1}{\epsilon}\right)\right)=\mathcal{O}\left(\frac{1}{\epsilon}\log\left(\frac{1}{\epsilon}\right)\right).$$

□

A.4 Smoothness Proofs

As remarked in the main paper, for the analysis of the actor, it is necessary to establish the Lipschitz continuity of $\nabla\eta(\theta)$, analogous to what was done in (Xu et al., 2020) for ∇J_θ. The following four lemmas fulfill this purpose.

Lemma 5. *Suppose Assumptions 1 and 3 hold, then $\forall\,\theta_1,\theta_2\in\mathbb{R}^{d_\theta}$, we have*

$$\|d_{I,\theta_1}(.,.)-d_{I,\theta_2}(.,.)\|_{TV}\leq C_d\|\theta_1-\theta_2\|_2,$$

where $C_d:=C_\pi\left(1+\lceil log_\rho\kappa^{-1}\rceil+\frac{1}{1-\rho}\right)$, and $d_{I,\theta}(s,a):=d_{I,\theta}(s)\pi(a|s)$, where $d_{I,\theta}(\cdot)$ is the (normalized) discounted state distribution when using policy π_θ and starting from $I(\cdot)$, which is an initialization distribution over the states; it can be taken as $\mu_0(.)$ (the initial state distribution) or $P(.|s',a')$ for any fixed state-action pair (s',a').

Proof. See Lemma 3 in (Xu et al., 2020). □

Lemma 6. *Suppose Assumptions 1 and 3 hold, then $\forall\,\theta_1,\theta_2\in\mathbb{R}^{d_\theta}$, we have*

$$|J_{\theta_1}-J_{\theta_2}|\leq L_J\|\theta_1-\theta_2\|_2,$$

where $L_J:=2R_{\max}(C_d+C_\pi)$.

Proof.

$$
\begin{aligned}
|J_{\theta_1} - J_{\theta_2}| &= \left|(1-\gamma)\int_s (V_{\theta_1}(s) - V_{\theta_2}(s))\mu(ds)\right| \\
&\leq (1-\gamma)\int_s |V_{\theta_1}(s) - V_{\theta_2}(s)|\mu(ds) \\
&\leq (1-\gamma)\int_s \left|\int_a Q_{\theta_1}(a,s)\pi_{\theta_1}(da|s) - \int_a Q_{\theta_2}(a,s)\pi_{\theta_2}(da|s)\right|\mu(ds) \\
&\leq (1-\gamma)\int_s \left|\int_a Q_{\theta_1}(a,s)\pi_{\theta_1}(da|s) \pm \int_a Q_{\theta_2}(a,s)\pi_{\theta_1}(da|s) - \int_a Q_{\theta_2}(a,s)\pi_{\theta_2}(da|s)\right|\mu(ds) \\
&\leq (1-\gamma)\int_s \int_a |(Q_{\theta_1}(a,s) - Q_{\theta_2}(a,s))|\pi_{\theta_1}(da|s)\mu(ds) \\
&\quad + (1-\gamma)\int_s \int_a |Q_{\theta_2}(a,s)|\,|\pi_{\theta_1}(da|s) - \pi_{\theta_2}(da|s)|\mu(ds)
\end{aligned}
$$

By Lemma 4 in (Xu et al., 2020), $|Q_{\theta_1}(s,a) - Q_{\theta_2}(s,a)| \leq \frac{2R_{\max}C_d}{1-\gamma}\|\theta_1 - \theta_2\|_2 \;\forall\,(s,a) \in S \times A$. Using this, and assumption 1.v, we have that

$$
\begin{aligned}
|J_{\theta_1} - J_{\theta_2}| &\leq 2R_{\max}C_d\|\theta_1 - \theta_2\|_2 + R_{\max}\int_s\int_a |\pi_{\theta_1}(da|s) - \pi_{\theta_2}(da|s)|\mu(ds) \\
&\leq 2R_{\max}C_d\|\theta_1 - \theta_2\|_2 + 2R_{\max}C_\pi\|\theta_1 - \theta_2\|_2 \\
&= 2R_{\max}(C_d + C_\pi)\|\theta_1 - \theta_2\|_2
\end{aligned}
$$

□

Lemma 7. *Suppose Assumptions 1 and 3 hold, then* $\forall\,\theta_1, \theta_2 \in \mathbb{R}^{d_\theta}$ *and* $\forall\,(s,a) \in S \times A$*, we have*

$$
\left|Q^\lambda_{\theta_1}(s,a) - Q^\lambda_{\theta_2}(s,a)\right| \leq L_{Q^\lambda}\|\theta_1 - \theta_2\|_2,
$$

where $L_{Q^\lambda} := \frac{2C_d R_{\lambda,\max} + 4\lambda L_J R_{\max}}{1-\gamma} = \frac{2C_d R_{\max} + 8\lambda R^2_{\max}(2C_d + C_\pi)}{1-\gamma}$*, and* $\lambda \geq 0$*.*

Proof. By definition,

$$
\begin{aligned}
Q^\lambda_\theta(s,a) &= \frac{1}{1-\gamma}\mathop{\mathbb{E}}_{\substack{s'\sim d_\theta(\cdot|s,a)\\ a'\sim\pi_\theta(\cdot|s')}}\left[R^\lambda_\theta(s,a)\right] \\
&= \frac{1}{1-\gamma}\int_{s'}\int_{a'} R^\lambda_\theta(s',a')d_\theta(ds'|s,a)\pi_\theta(da'|s') \\
&= \frac{1}{1-\gamma}\int_{(s',a')} R^\lambda_\theta(s',a')d_\theta(ds',da'|s,a),
\end{aligned}
$$

where $d_\theta(s',a'|s,a) := d_\theta(s'|s,a)\pi_\theta(a'|s')$, and $d_\theta(\cdot|s,a)$ is the (normalized) discounted state distribution when

using policy π_θ after taking action a in state s. We then have that

$$
\begin{aligned}
&(1-\gamma)\left|Q^\lambda_{\theta_1}(s,a) - Q^\lambda_{\theta_2}(s,a)\right| \\
&= \left|\int_{(s',a')} \left[R^\lambda_{\theta_1}(s',a') d_{\theta_1}(ds',da'|s,a) - R^\lambda_{\theta_2}(s',a') d_{\theta_2}(ds',da'|s,a)\right]\right| \\
&= \left|\int_{(s',a')} \left[R^\lambda_{\theta_1}(s',a') d_{\theta_1}(ds',da'|s,a) \pm R^\lambda_{\theta_1}(s',a') d_{\theta_2}(ds',da'|s,a) - R^\lambda_{\theta_2}(s',a') d_{\theta_2}(ds',da'|s,a)\right]\right| \\
&= \left|\int_{(s',a')} R^\lambda_{\theta_1}(s',a')(d_{\theta_1}(ds',da'|s,a) - d_{\theta_2}(ds',da'|s,a))\right| + \left|\int_{(s',a')} (R^\lambda_{\theta_1}(s',a') - R^\lambda_{\theta_2}(s',a')) d_{\theta_2}(ds',da'|s,a)\right| \\
&\le \int_{(s',a')} \left|R^\lambda_{\theta_1}(s',a')\right| \left|d_{\theta_1}(ds',da'|s,a) - d_{\theta_2}(ds',da'|s,a)\right| + \int_{(s',a')} \left|R^\lambda_{\theta_1}(s',a') - R^\lambda_{\theta_2}(s',a')\right| d_{\theta_2}(ds',da'|s,a) \\
&\le R_{\lambda,\max} \int_{(s',a')} \left|d_{\theta_1}(ds',da'|s,a) - d_{\theta_2}(ds',da'|s,a)\right| \\
&\quad + \int_{(s',a')} \left|2\lambda R(s',a')(J_{\theta_1} - J_{\theta_2}) - \lambda(J^2_{\theta_1} - J^2_{\theta_2})\right| d_{\theta_2}(ds',da'|s,a) \\
&\le 2C_d R_{\lambda,\max} \|\theta_1 - \theta_2\|_2 + \int_{(s',a')} \left|2\lambda R(s',a')(J_{\theta_1} - J_{\theta_2}) - \lambda(J_{\theta_1} + J_{\theta_2})(J_{\theta_1} - J_{\theta_2})\right| d_{\theta_2}(ds',da'|s,a) \\
&\le 2C_d R_{\lambda,\max} \|\theta_1 - \theta_2\|_2 + \int_{(s',a')} \left|\lambda(2R(s',a') - (J_{\theta_1} + J_{\theta_2}))\right| \left|J_{\theta_1} - J_{\theta_2}\right| d_{\theta_2}(ds',da'|s,a) \\
&\le 2C_d R_{\lambda,\max} \|\theta_1 - \theta_2\|_2 + 4\lambda R_{\max} |J_{\theta_1} - J_{\theta_2}| \\
&\le 2C_d R_{\lambda,\max} \|\theta_1 - \theta_2\|_2 + 4\lambda L_J R_{\max} \|\theta_1 - \theta_2\|_2 \\
&= \left(2C_d R_{\lambda,\max} + 4\lambda L_J R_{\max}\right) \|\theta_1 - \theta_2\|_2 \\
&= \left(2C_d R_{\max} + 8\lambda R^2_{\max}(2C_d + C_\pi)\right) \|\theta_1 - \theta_2\|_2.
\end{aligned}
$$

□

Lemma 8. *Suppose Assumptions 1 and 3 hold, then* $\forall\, \theta_1, \theta_2$, *we have*

$$\|\nabla \eta_{\theta_1} - \nabla \eta_{\theta_2}\|_2 \le L_\eta \|\theta_1 - \theta_2\|_2,$$

where $L_\eta := \frac{2R_{\lambda,\max} C_\psi C_d}{1-\gamma} + C_\psi L_{Q^\lambda} + \frac{R_{\lambda,\max} L_\psi}{1-\gamma}$, *and* $\lambda \ge 0$.

Proof.

$$
\begin{aligned}
\|\nabla \eta_{\theta_1} - \nabla \eta_{\theta_2}\|_2 &= \left\| \int_{(s,a)} \left[\psi_{\theta_1}(s,a) Q^\lambda_{\theta_1}(s,a) d_{\mu,\theta_1}(ds,da) - \psi_{\theta_2}(s,a) Q^\lambda_{\theta_2}(s,a) d_{\mu,\theta_2}(ds,da)\right] \right\|_2 \\
&\le \int_{(s,a)} \left\|Q^\lambda_{\theta_1}(s,a)\psi_{\theta_1}(s,a)\right\|_2 \left|d_{\mu,\theta_1}(ds,da) - d_{\mu,\theta_2}(ds,da)\right| \\
&\quad + \int_{(s,a)} \left|Q^\lambda_{\theta_1}(s,a) - Q^\lambda_{\theta_2}(s,a)\right| \|\psi_{\theta_1}(s,a)\|_2 \, d_{\mu,\theta_2}(ds,da) \\
&\quad + \int_{(s,a)} \left|Q^\lambda_{\theta_2}(s,a)\right| \|\psi_{\theta_1}(s,a) - \psi_{\theta_2}(s,a)\|_2 \, d_{\mu,\theta_2}(ds,da) \\
&\le \frac{2R_{\lambda,\max} C_\psi C_d}{1-\gamma} \|\theta_1 - \theta_2\|_2 + C_\psi L_{Q^\lambda} \|\theta_1 - \theta_2\|_2 + \frac{R_{\lambda,\max} L_\psi}{1-\gamma} \|\theta_1 - \theta_2\|_2 \\
&= \left(\frac{2R_{\lambda,\max} C_\psi C_d}{1-\gamma} + C_\psi L_{Q^\lambda} + \frac{R_{\lambda,\max} L_\psi}{1-\gamma}\right) \|\theta_1 - \theta_2\|_2,
\end{aligned}
$$

where the last inequality follows from Assumption 1, Lemma 5, and Lemma 7. □

A.5 Proof of the Actor's Bound

In this section, we develop the proof of Theorem 2, which provided a finite time bound for Algorithm 2, where the critic is learned using direct mini-batch TD, whose analysis was the subject of Theorem 1 and Section A.3. Remember that our aim is to bound $\mathbb{E}\left[\left\|\nabla\eta(\theta_{\hat{T}})\right\|_2^2\right]$. To do this, we will extend the analysis in (Xu et al., 2020) to our risk-averse case. We first define the following quantities, which will help us in the analysis[21]:

- the TD-error[22] $\delta_\omega(s,a,s') = R^\lambda(s,a,J) + \gamma\phi(s')^\top\omega - \phi(s)^\top\omega$, which employs the exact expected return J;
- the *approximated* TD-error $\hat{\delta}_\omega(s,a,s') = R^\lambda(s,a,\hat{J}) + \gamma\phi(s')^\top\omega - \phi(s)^\top\omega$, which employs, instead, the current Monte-Carlo estimate of the expected return $\hat{J}$;
- $v_t(\omega,\theta) = \frac{1}{B}\sum_{i=0}^{B-1}\delta_\omega(s_{t,i},a_{t,i},s'_{t,i+1})\psi_{\theta_t}(s_{t,i},a_{t,i})$, which would have been the estimated gradient at time t (using a critic with parameters ω) if we had access to the true J_θ;
- $\hat{v}_t(\omega,\theta) = \frac{1}{B}\sum_{i=0}^{B-1}\hat{\delta}_\omega(s_{t,i},a_{t,i},s'_{t,i+1})\psi_{\theta_t}(s_{t,i},a_{t,i})$, which is the estimated gradient at time t (using a critic with parameters ω) based on $\hat{J}_\theta$;
- $A_\omega(s,a) = \mathbb{E}_{s'\sim P(\cdot|s,a)}[\delta_\omega(s,a,s')]$, which is the expected value of the TD-error δ_ω at a given state-action pair when the next state is sampled from the transition kernel of the original MDP;
- $g(\omega,\theta) = \mathbb{E}_{\substack{s\sim d_{\mu_0,\pi_\theta}(\cdot)\\ a\sim\pi_\theta(\cdot|s)}}[A_\omega(s,a)\psi_\theta(s,a)]$, which is the expectation of the estimated gradient when using a critic with parameter vector ω and the true expected return J_θ.

Next, we prove two propositions, which will be combined to bound the expectation on the gradient norm.

Proposition 3. *Suppose Assumption 1 holds, then the following holds at the t^{th} iteration of Algorithm 2:*

$$\left(\frac{\alpha}{2} - 2L_\eta\alpha^2\right)\|\nabla\eta(\theta_t)\|_2^2 \leq \eta(\theta_{t+1}) - \eta(\theta_t) + \left(\frac{\alpha}{2} + 2L_\eta\alpha^2\right)\|\hat{v}_t(\omega_t,\theta_t) - \nabla\eta(\theta_t)\|_2^2.$$

Proof. By applying the Mean-Value Theorem, for some $0 \leq \Delta \leq 1$ there is some $\tilde{\theta} = \Delta\theta_t + (1-\Delta)\theta_{t+1}$ such that:

$$\begin{aligned}\eta(\theta_{t+1}) &= \eta(\theta_t) + (\theta_{t+1}-\theta_t)^\top\nabla\eta(\tilde{\theta}) = \eta(\theta_t) + (\theta_{t+1}-\theta_t)^\top\nabla\eta(\tilde{\theta}) \pm (\theta_{t+1}-\theta_t)^\top\nabla\eta(\theta_t)\\ &= \eta(\theta_t) + (\theta_{t+1}-\theta_t)^\top\left(\nabla\eta(\tilde{\theta}) - \nabla\eta(\theta_t)\right) + (\theta_{t+1}-\theta_t)^\top\nabla\eta(\theta_t).\end{aligned}$$

By using Cauchy-Schwarz we also have:

$$\begin{aligned}(\theta_{t+1}-\theta_t)^\top\left(\nabla\eta(\tilde{\theta}) - \nabla\eta(\theta_t)\right) &\geq -\|\theta_{t+1}-\theta_t\|_2\|\nabla\eta(\tilde{\theta}) - \nabla\eta(\theta_t)\|_2\\ &\geq -L_\eta\|\theta_{t+1}-\theta_t\|_2\|\tilde{\theta}-\theta_t\|_2\\ &\geq -L_\eta\|\theta_{t+1}-\theta_t\|_2^2\end{aligned}$$

where we also used that the gradient of η is Lipschitz (Lemma 8).

We exploit this relationship in the previous equation, together with the definition of the policy parameters

[21] Remember that $R^\lambda(s,a,\hat{J}) \coloneqq R(s,a) - \lambda(R(s,a)-\hat{J})^2$.

[22] Note that the $\delta_\omega(s,a,s')$ and $\hat{\delta}_\omega(s,a,s')$ do depend on the current policy since they depend on its expected return, or an estimate of it. However, we do not explicitly express this dependence as to not burden the notation since it is usually clear from the context.

update:

$$\begin{aligned}
\eta(\theta_{t+1}) &\geq \eta(\theta_t) - L_\eta\|\theta_{t+1} - \theta_t\|_2^2 + (\theta_{t+1} - \theta_t)^\top \nabla\eta(\theta_t) \\
&= \eta(\theta_t) - \alpha^2 L_\eta\|\hat{v}_t(\omega_t, \theta_t)\|_2^2 + \alpha\hat{v}_t(\omega_t, \theta_t)^\top \nabla\eta(\theta_t) \\
&= \eta(\theta_t) - \alpha^2 L_\eta\|\hat{v}_t(\omega_t, \theta_t) \pm \nabla\eta(\theta_t)\|_2^2 + \alpha\langle\hat{v}_t(\omega_t, \theta_t) \pm \nabla\eta(\theta_t), \nabla\eta(\theta_t)\rangle \\
&\overset{(1)}{\geq} \eta(\theta_t) - 2\alpha^2 L_\eta\|\nabla\eta(\theta_t)\|_2^2 - 2\alpha^2 L_\eta\|\hat{v}_t(\omega_t, \theta_t) - \nabla\eta(\theta_t)\|_2^2 \\
&\quad + \alpha\|\nabla\eta(\theta_t)\|_2^2 + \alpha\langle\hat{v}_t(\omega_t, \theta_t) - \nabla\eta(\theta_t), \nabla\eta(\theta_t)\rangle \\
&\overset{(2)}{\geq} \eta(\theta_t) - 2\alpha^2 L_\eta\|\nabla\eta(\theta_t)\|_2^2 - 2\alpha^2 L_\eta\|\hat{v}_t(\omega_t, \theta_t) - \nabla\eta(\theta_t)\|_2^2 \\
&\quad + \alpha\|\nabla\eta(\theta_t)\|_2^2 - \frac{\alpha}{2}\|\hat{v}_t(\omega_t, \theta_t) - \nabla\eta(\theta_t)\|_2^2 - \frac{\alpha}{2}\|\nabla\eta(\theta_t)\|_2^2,
\end{aligned}$$

where in the last two steps we used, respectively, Lemma 2.ii and Lemma 2.i in (1) and (2). By re-ordering terms we obtain the desired result. □

The last term in the bound of the last proposition represents how far the estimated gradient is from the true one. Analogous to the approach in (Xu et al., 2020), the next proposition bounds the expected value of this quantity.

Proposition 4. *Suppose Assumptions 1 to 5 hold, then the following holds for Algorithm 2 (when using direct mini-batch TD for the critic), where $\mathcal{F}_t$ is the filtration on the samples up to iteration t:*

$$\begin{aligned}
\mathbb{E}\left[\|\hat{v}_t(\omega_t, \theta_t) - \nabla\eta(\theta_t)\|_2^2|\mathcal{F}_t\right] \leq &\frac{24(R_{\lambda,\max} + 2\tilde{C}_\omega)^2[1 + (k-1)\rho]}{B(1-\rho)} \\
&+ 48\lambda^2 R_{\max}^2\,\mathbb{E}\left[|J - \hat{J}|^2|\mathcal{F}_t\right] + 12\lambda^2\,\mathbb{E}\left[|\hat{J}^2 - J^2|^2|\mathcal{F}_t\right] \\
&+ 24\|\omega_{J_t}^* - \omega_t\|_2^2 + 12\;\xi_{appr},
\end{aligned}$$

where J_t is short for $J_{\pi_{\theta_t}}$, $\omega_{J_t}^*$ is the TD fixed point for the transformed value function of policy π_{θ_t}, and ω_t is its learned estimate.

Proof. Consider $\|\hat{v}_t(\omega_t, \theta_t) - \nabla\eta(\theta_t)\|_2^2$, we can decompose it in the following way (followed by an application of Lemma 2.ii):

$$\begin{aligned}
&\|\hat{v}_t(\omega_t, \theta_t) - \nabla\eta(\theta_t)\|_2^2 \\
&= \left\|\hat{v}_t(\omega_t, \theta_t) \pm v_t(\omega_{J_t}^*, \theta_t) \pm g(\omega_{J_t}^*, \theta_t) - \nabla\eta(\theta_t)\right\|_2^2 \\
&\leq 3\underbrace{\left\|\hat{v}_t(\omega_t, \theta_t) - v_t(\omega_{J_t}^*, \theta_t)\right\|_2^2}_{(a)} + 3\left\|v_t(\omega_{J_t}^*, \theta_t) - g(\omega_{J_t}^*, \theta_t)\right\|_2^2 + 3\underbrace{\left\|g(\omega_{J_t}^*, \theta_t) - \nabla\eta(\theta_t)\right\|_2^2}_{(b)}.
\end{aligned} \tag{14}$$

We now focus on (a):

$$\begin{aligned}
\left\|\hat{v}_t(\omega_t,\theta_t)-v_t(\omega^*_{J_t},\theta_t)\right\|_2^2 =& \left\|\frac{1}{B}\sum_{i=0}^{B-1}\psi_{\theta_t}(s_{t,i},a_{t,i})\left[\hat{\delta}_{\omega_t}(s_{t,i},a_{t,i},s'_{t,i+1})-\delta_{\omega^*_{J_t}}(s_{t,i},a_{t,i},s'_{t,i+1})\right]\right\|_2^2 \\
\leq& \frac{1}{B}\sum_{i=0}^{B-1}\underbrace{\|\psi_{\theta_t}(s_{t,i},a_{t,i})\|_2^2}_{\leq C_\psi=1}\left|\hat{\delta}_{\omega_t}(s_{t,i},a_{t,i},s'_{t,i+1})-\delta_{\omega^*_{J_t}}(s_{t,i},a_{t,i},s'_{t,i+1})\right|^2 \\
\leq& \frac{1}{B}\sum_{i=0}^{B-1}\Big|R^\lambda(s_{t,i},a_{t,i},\hat{J})-R^\lambda(s_{t,i},a_{t,i},J)+\gamma\left(\phi(s'_{t,i+1})^\top\omega_t-\phi(s'_{t,i+1})^\top\omega^*_{J_t}\right)+ \\
&\qquad +\left(\phi(s_{t,i})^\top\omega^*_{J_t}-\phi(s_{t,i})^\top\omega_t\right)\Big|^2 \\
\overset{(1)}{\leq}& \frac{1}{B}\sum_{i=0}^{B-1}2\left|R^\lambda(s_{t,i},a_{t,i},\hat{J})-R^\lambda(s_{t,i},a_{t,i},J)\right|^2+2\left|(\gamma\phi(s'_{t,i+1})-\phi(s_{t,i}))^\top(\omega_t-\omega^*_{J_t})\right|^2 \\
\overset{(2)}{\leq}& \frac{1}{B}\sum_{i=0}^{B-1}2\left|R^\lambda(s_{t,i},a_{t,i},\hat{J})-R^\lambda(s_{t,i},a_{t,i},J)\right|^2+8\left\|\omega^*_{J_t}-\omega_t\right\|_2^2 \\
\overset{(3)}{=}& \frac{1}{B}\sum_{i=0}^{B-1}2\lambda^2\left|2R(s_{t,i},a_{t,i})(J-\hat{J})+\hat{J}^2-J^2\right|^2+8\left\|\omega^*_{J_t}-\omega_t\right\|_2^2 \\
\overset{(4)}{\leq}& 16\lambda^2R_{\max}^2|J-\hat{J}|^2+4\lambda^2|\hat{J}^2-J^2|^2+8\|\omega^*_{J_t}-\omega_t\|_2^2.
\end{aligned}$$

where (1) is an application of Lemma 2.ii, (2) is due to Cauchy–Schwarz, Lemma 2.ii, and Assumption 5.ii, (3) to definition of R^λ, and in (4) Lemma 2.ii is applied again.

We can then exploit results from Theorem 5 in Xu et al. (2020), to bound (b) as:

$$\left\|g(\omega^*_{J_t},\theta_t)-\nabla\eta(\theta_t)\right\|_2^2\leq 4\xi_{appr}.$$

Substituting back to inequality (14) and taking the expectation w.r.t. the filtration $\mathcal{F}_t$, we get:

$$\begin{aligned}
\mathbb{E}\left[\|\hat{v}_t(\omega_t,\theta_t)-\nabla\eta(\theta_t)\|_2^2|\mathcal{F}_t\right] \leq& \,3\,\mathbb{E}\left[\left\|v_t(\omega^*_{J_t},\theta_t)-g(\omega^*_{J_t},\theta_t)\right\|_2^2|\mathcal{F}_t\right] \\
&+48\lambda^2R_{max}^2\,\mathbb{E}\left[|J-\hat{J}|^2|\mathcal{F}_t\right]+12\lambda^2\,\mathbb{E}\left[|\hat{J}^2-J^2|^2|\mathcal{F}_t\right] \\
&+24\|\omega^*_{J_t}-\omega_t\|_2^2+12\;\xi_{appr}.
\end{aligned}$$

To bound the conditional expectation on the RHS, we follow again the proof in Xu et al. (2020) to have:

$$\mathbb{E}\left[\left\|v_t(\omega^*_{J_t},\theta_t)-g(\omega^*_{J_t},\theta_t)\right\|_2^2|\mathcal{F}_t\right]\leq\frac{8(R_{\lambda,\max}+2\tilde{C}_\omega)^2(1+(k-1)\rho)}{B(1-\rho)},$$

where $R_{\lambda,\max}+2\tilde{C}_\omega$ serves[23], in our case, as a uniform (over any policy π_θ) upper bound for the TD-error (or estimated transformed advantage function) evaluated at any state-action pair using the estimated transformed value function at its fixed point. □

Similar to what we did in the critic's section, the following theorem is an explicit restatement of the bound in Theorem 2, while the sample complexity derivation is shown in a separate corollary.

Theorem 4 (Explicit Statement of the Bound in Theorem 2)**.** *Suppose Assumptions 1 to 5 hold, and suppose we run Algorithm 2 for T iterations with the critic learned as described in Theorem 1, then if $\alpha=\frac{1}{8L_\eta}$, we have:*

$$\mathbb{E}\left[\left\|\nabla\eta(\theta_{\hat{T}})\right\|_2^2\right]\leq\frac{64L_\eta R_{\lambda,\max}}{T}+\xi_{distr}+18\xi_J+72\frac{\sum_{t=0}^{T-1}\mathbb{E}[\|\omega^*_{J_t}-\omega_t\|_2^2]}{T}+36\xi_{appr},$$

[23] See the beginning of Section A.3 for the definition of $\tilde{C}_\omega$.

where ω_t is the parameter vector of the learned critic at the t^{th} iteration, $\omega^*_{J_t}$ is the TD fixed point for the true transformed value function of policy π_{θ_t}, ξ_J is the same as the one defined in Theorem 3, and

$$\xi_{distr} \coloneqq \frac{72(R_{\lambda,\max} + 2\tilde{C}_\omega)^2(1 + (k-1)\rho)}{B(1-\rho)}.$$

Proof. Taking the conditioned expectation on the result of Proposition 3 and plugging what we obtained with Proposition 4 we obtain the following:

$$\begin{aligned}
&\left(\frac{\alpha}{2} - 2L_\eta\alpha^2\right)\mathbb{E}\left[\|\nabla\eta(\theta_t)\|_2^2|\mathcal{F}_t\right] \\
&\leq \mathbb{E}\left[\eta(\theta_{t+1})|\,\mathcal{F}_t\right]] - \eta(\theta_t) + \left(\frac{\alpha}{2} + 2L_\eta\alpha^2\right)\left[\frac{24(R_{\lambda,\max} + 2\tilde{C}_\omega)^2[1 + (k-1)\rho]}{B(1-\rho)}\right. \\
&\quad \left. + 48\lambda^2 R_{\max}^2\,\mathbb{E}\left[|J - \hat{J}|^2|\mathcal{F}_t\right] + 12\lambda^2\,\mathbb{E}\left[|\hat{J}^2 - J^2|^2|\mathcal{F}_t\right] + 24\|\omega^*_{J_t} - \omega_t\|_2^2 + 12\;\xi_{appr}\right] \\
&\leq \mathbb{E}\left[\eta(\theta_{t+1})|\,\mathcal{F}_t\right]] - \eta(\theta_t) + \left(\frac{\alpha}{2} + 2L_\eta\alpha^2\right)\left[\frac{24(R_{\lambda,\max} + 2\tilde{C}_\omega)^2(1 + (k-1)\rho)}{B(1-\rho)}\right. \\
&\quad + 48\lambda^2 R_{\max}^2\left(\gamma^{2T_J}R_{\max}^2 + \frac{R_{\max}^2}{L}\right) + 12\lambda^2\left(4\gamma^{2T_J}R_{\max}^4 + \left(\frac{4}{L} + \frac{7}{L^2} + \frac{5}{L^3}\right)R_{\max}^4\right) + \\
&\quad \left. + 24\|\omega^*_{J_t} - \omega_t\|_2^2 + 12\;\xi_{appr}\right],
\end{aligned}$$

We let $\alpha = \frac{1}{8L_\eta}$ and we multiply both sides by $32L_\eta$ to get:

$$\mathbb{E}\left[\|\nabla\eta(\theta_t)\|_2^2|\mathcal{F}_t\right] \leq\; 32L_\eta\left(\mathbb{E}\left[\eta(\theta_{t+1})|\mathcal{F}_t\right]] - \eta(\theta_t)\right) + \xi_{distr} + 18\xi_J + 72\|\omega^*_{J_t} - \omega_t\|_2^2 + 36\xi_{appr},$$

with

$$\xi_{distr} \coloneqq \frac{72(R_{\lambda,\max} + 2\tilde{C}_\omega)^2(1 + (k-1)\rho)}{B(1-\rho)},$$

which bounds the variance of the mini-batch estimate of the gradient if the critic was at the TD fixed point, while ξ_J, the error arising from the expected return estimation, has been already defined in Theorem 3.

We take the expectation w.r.t. $\mathcal{F}_t$ to both sides to yield:

$$\mathbb{E}\left[\|\nabla\eta(\theta_t)\|_2^2\right] \leq\; 32L_\eta(\mathbb{E}\left[\eta(\theta_{t+1})\right] - \mathbb{E}\left[\eta(\theta_t)\right]) + \xi_{distr} + 18\xi_J + 72\,\mathbb{E}[\|\omega^*_{J_t} - \omega_t\|_2^2] + 36\xi_{appr}.$$

Taking the summation of the last result over $t = 0, \ldots, T-1$ and dividing both sides by T gives:

$$\begin{aligned}
\mathbb{E}\left[\left\|\nabla\eta(\theta_{\hat{T}})\right\|_2^2\right] &= \frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\|\nabla\eta(\theta_t)\|_2^2\right] \\
&\leq 32L_\eta\frac{\mathbb{E}\left[\eta(\theta_T)\right] - \eta(\theta_0)}{T} + \xi_{distr} + 18\xi_J + 72\frac{\sum_{t=0}^{T-1}\mathbb{E}[\|\omega^*_{J_t} - \omega_t\|_2^2]}{T} + 36\xi_{appr} \\
&\leq \frac{64L_\eta R_{\lambda,\max}}{T} + \xi_{distr} + 18\xi_J + 72\frac{\sum_{t=0}^{T-1}\mathbb{E}[\|\omega^*_{J_t} - \omega_t\|_2^2]}{T} + 36\xi_{appr}.
\end{aligned}$$

□

Corollary 4.1 (Restatement of the Sample Complexity Result in Theorem 2)**.** *Suppose we are in the same setting of Theorem 4, and assume that the parameters used in the critic are conditioned as to make $\mathbb{E}\left[\left\|\omega_t - \omega^*_{J_t}\right\|_2^2\right] \leq \frac{\epsilon}{360}$ for all $t = 0, \ldots, T-1$. Then, if additionally*

- $T \geq \frac{320L_\eta(R_{\max} + 4\lambda R_{\max}^2)}{\epsilon}$,

- $B \geq \frac{360((R_{\max}+4\lambda R_{\max}^2)+2\tilde{C}_\omega)^2(1+(k-1)\rho)}{(1-\rho)\epsilon}$,
- $T_J \geq \frac{\log\left(\frac{1440\lambda^2 R_{max}^4}{\epsilon}\right)}{2(1-\gamma)}$,
- $L \geq \frac{3600\lambda^2 R_{\max}^4}{\epsilon}$,

we have that

$$\mathbb{E}\left[\left\|\nabla\eta(\omega_{\hat{T}})\right\|_2^2\right] \leq \epsilon + \mathcal{O}(\xi_{appr}),$$

with the total sample complexity given by:

$$T((2-\gamma)B + MT_c + LT_J) = \mathcal{O}\left(\epsilon^{-2}\log\left(\epsilon^{-1}\right)\right).$$

Proof. In order to compute the different contributions to sample complexity, we will split the error bound obtained in Theorem 4 in its components. We then bound the components in the following way:

- $\frac{64L_\eta(R_{\max}+4\lambda R_{\max}^2)}{T} \leq \epsilon_1$,
- $\frac{72(R_{\max}+4\lambda R_{\max}^2+2\tilde{C}_\omega)^2(1+(k-1)\rho)}{B(1-\rho)} \leq \epsilon_2$,
- $288\lambda^2 R_{max}^4\gamma^{2T_J} \leq 288\lambda^2 R_{max}^4 e^{-2(1-\gamma)T_J} \leq \epsilon_3$,
- $36\lambda^2 R_{max}^4(\frac{8}{L}+\frac{7}{L^2}+\frac{5}{L^3}) \overset{L>1}{\leq} 36\lambda^2 R_{max}^4(\frac{20}{L}) \leq \epsilon_4$,
- $72\frac{\sum_{t=0}^{T-1}\mathbb{E}[\|\omega_{J_t}^*-\omega_t\|_2^2]}{T} \leq \epsilon_5$,

where we have split $18\xi_J$ in two parts, and we have ignored the approximation error ξ_{appr}, which is irreducible. We set, then, each ϵ_i to $\frac{\epsilon}{5}$. Rearranging terms in each inequality, we obtain then, by the conditions on the parameters indicated in the statement[24], the desired error. In order to obtain it, the following sample complexity is incurred:

$$T((2-\gamma)B + MT_c + LT_J) = \mathcal{O}\left(\frac{1}{\epsilon}\left(\frac{1}{\epsilon}+\frac{1}{\epsilon}\log\left(\frac{1}{\epsilon}\right)+\frac{1}{\epsilon}\log\left(\frac{1}{\epsilon}\right)\right)\right) = \mathcal{O}\left(\epsilon^{-2}\log\left(\epsilon^{-1}\right)\right)$$

where the $(2-\gamma)$ extra factor is due to the actor sampling process, which needs to sample twice at each restart, which can happen at each step with probability $1-\gamma$. □

A.6 A Note About the Critic's Sampling Process

In this section, we highlight a subtle point about the sampling process of the critic. The following applies both to our analysis and to that of (Xu et al., 2020). We start by recalling our definition of the approximation error:

$$\xi_{appr} := \sup_{\theta\in\mathbb{R}^{d_\theta}} \mathbb{E}_{s\sim d_{\mu_0,\pi_\theta}(\cdot)}\left[\left|V_{\pi_\theta}^\lambda(s) - \phi(s)^\intercal\omega_{J_\theta}^*\right|^2\right],$$

which represents, for the worst possible policy, the mean squared error, over the discounted state distribution, between the true transformed value function and the approximated transformed value function at the TD fixed point. We use the discounted state distribution in the definition as this quantity is used in the analysis of the actor, whose mini-batches (used to estimate the gradient) are sampled from the modified transition kernel:

$$\tilde{P}(\cdot|s,a) = \gamma P(\cdot|s,a) + (1-\gamma)\mu_0(\cdot),$$

which causes the encountered states to be distributed, at steady-state, according to the discounted state distribution of the policy. Thus, ξ_{appr} is a convenient representation of the approximation error of the critic. However,

[24] And the conditions adapted from Corollary 3.1 needed to make $\mathbb{E}[\|\omega_t - \omega_{J_t}^*\|_2^2] \leq \frac{\epsilon}{360}$ (instead of just ϵ) for all $t = 0, \ldots, T-1$.

the mini-batches used for training the critic are sampled from original kernel, and the encountered states are thus distributed, at steady-state, according to the ordinary stationary distribution of the policy. And thus, the TD fixed point is naturally characterized using that distribution (see Section 4.1). The issue then is that the ξ_{appr} not only conceals the approximation error incurred due to using TD learning with linear function approximation, but also conceals some notion of divergence between the ordinary stationary distribution of a policy and its discounted state distribution.

A simple way to deal with this issue is to adopt, for the critic, the same sampling scheme that we use for the actor (see Section 3.5), which also ensures that the next state used in the TD-error is still sampled correctly from the original transition kernel. One would then characterize the TD fixed point (i.e., the definitions of A and $b(\hat{J})$) using the discounted state distribution. Fortunately, the analysis of the critic is still applicable after this modification. This is because the uniform Ergodicity assumption (Assumption 3, which is adapted from (Xu et al., 2020)) readily considers the case when the modified kernel is used (at which case the steady-state distribution is the discounted state distribution), which was already needed for the analysis of the actor. The incurred cost, much like in the actor's case, is that the number of sampled interactions per mini-batch is now, on average, $(2-\gamma)M$ since two states are sampled whenever we restart, which can happen at any step with probability $1-\gamma$. Note that this additional cost diminishes as γ approaches 1.

B Analysis of the Factored Method

B.1 Deriving The Factored Method Formula

We can start by recalling the definition of the transformed value function:

$$V_\pi^\lambda(s) := \mathop{\mathbb{E}}_{\substack{a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}}\left[\sum_{t=0}^{\infty}\gamma^t\Big(R(s_t,a_t)-\lambda\big(R(s_t,a_t)-J_\pi\big)^2\Big)\middle| s_0=s\right].$$

By expanding the squared term and using the linearity of the expected value, we get:

$$V_\pi^\lambda(s') = \mathop{\mathbb{E}}_{\substack{a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}}\left[\sum_{t=0}^{\infty}\gamma^t R(s_t,a_t)\middle| s_0=s\right] - \lambda \mathop{\mathbb{E}}_{\substack{a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}}\left[\sum_{t=0}^{\infty}\gamma^t R^2(s_t,a_t)\middle| s_0=s\right] + 2\lambda J_\pi \mathop{\mathbb{E}}_{\substack{a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}}\left[\sum_{t=0}^{\infty}\gamma^t R(s_t,a_t)\middle| s_0=s\right] - \frac{\lambda J_\pi^2}{1-\gamma}.$$

Then, by the definitions of V^π and M^π (see (7)), we get that:

$$V_\pi^\lambda(s) = (1+2\lambda J_\pi)V^\pi(s) - \lambda M^\pi(s) - \frac{\lambda}{1-\gamma}J_\pi^2.$$

B.2 Analyzing MVAC With a Factored Mini-Batch TD Critic

The goal in this section is to derive a finite time bound for Algorithm 2 when the critic is trained using the factored method where V^π and M^π are learned using Algorithm 1. We have remarked before that the most natural and efficient way to do this is to learn both V^π and M^π in parallel using the same sample path. As the scheme in Algorithm 1 does not allow us to represent this case, we state the desired scheme explicitly in Algorithm 4. For a given policy, after estimating V^π and M^π using Algorithm 4, we can obtain $\hat{J}$ using Algorithm 3 as usual, and then set ($\forall s \in \mathcal{S}$):

$$\hat{V}^\lambda(s) = (1+2\lambda\hat{J})\phi(s)^\intercal\omega^v - \lambda\phi(s)^\intercal\omega^m - \frac{\lambda}{1-\gamma}\hat{J}^2,$$

as the critic to be used in Algorithm 2 for estimating the mean-volatility gradient at the given policy. Contrary to the direct case, we do not need extra effort for analyzing the critic, thanks to the (almost complete) separation between the estimators in the factored method. We will directly leverage the bound in (Xu et al., 2020) for V^π, and a small variant of it for M^π, along with the results in Propositions 1 and 2. To do this, we state the following assumption, which is analogous to Assumption 4.

Algorithm 4 Mini-batch Factored Mean-Volatility TD (Mini-batch F-MVTD)

Input: $s_{\text{ini}}, \theta, \phi, \gamma, \beta, T_c, M$
Initialize: ω_0^v, ω_0^m
Set $s_{-1,M} = s_{\text{ini}}$
for $k = 0, \dots, T_c - 1$ **do**
  $s_{k,0} = s_{k-1,M}$
  for $j = 0, \dots, M-1$ **do**
    $a_{k,j} \sim \pi_\theta(s_{k,j})$, $s_{k,j+1} \sim P(.|s_{k,j}, a_{k,j})$
    $\delta^v_{\omega_k}(s_{k,j}, a_{k,j}, s_{k,j+1}) = R(s_{k,j}, a_{k,j}) + \gamma\phi(s_{k,j+1})^\intercal\omega_k^v - \phi(s_{k,j})^\intercal\omega_k^v$
    $\delta^m_{\omega_k}(s_{k,j}, a_{k,j}, s_{k,j+1}) = R^2(s_{k,j}, a_{k,j}) + \gamma\phi(s_{k,j+1})^\intercal\omega_k^m - \phi(s_{k,j})^\intercal\omega_k^m$
  end for
  $\omega^v_{k+1} = \omega^v_k + \beta\frac{1}{M}\sum_{j=0}^{M-1}\delta^v_{\omega_k}(s_{k,j}, a_{k,j}, s_{k,j+1})\phi(s_{k,j})$
  $\omega^m_{k+1} = \omega^m_k + \beta\frac{1}{M}\sum_{j=0}^{M-1}\delta^m_{\omega_k}(s_{k,j}, a_{k,j}, s_{k,j+1})\phi(s_{k,j})$
end for
Output: $\omega^v_{T_c}, \omega^m_{T_c}, s_{k,M}$

Assumption 6. *For any triple $(s_{i,t}, a_{i,t}, s_{i,t+1}) \in \mathcal{S}\times\mathcal{A}\times\mathcal{S}$, there exists real constants $C_{v,b}$ and $C_{m,b}$ such that $\|\phi(s_{i,t})R(s_{i,t}, a_{i,t})\|_2 \le C_{v,b}$ and $\|\phi(s_{i,t})R^2(s_{i,t}, a_{i,t})\|_2 \le C_{m,b}$.*

Again, while stated as an assumption, the previous statement is justified since the reward function and the norm of the feature vectors are bounded. Analogous to C_ω in the direct case, and with a similar justification, we use $C_{v,\omega}$ and $C_{m,\omega}$ such that $\|\omega^{v*}\|_2 \le C_{v,\omega} \coloneqq \frac{C_{v,b}}{\bar{\sigma}}$ and $\|\omega^{m*}\|_2 \le C_{m,\omega} \coloneqq \frac{C_{m,b}}{\bar{\sigma}}$, where ω^{v*} and ω^{m*} are the TD fixed points of V^π and M^π respectively. Also, like we did in the direct case (see Section A.3), we define policy-independent versions of $C_{v,\omega}$ and $C_{m,\omega}$ to be used in the actor's bound. Namely, we define $\tilde{C}_{v,\omega} \coloneqq \frac{C_{v,b}}{\tilde{\sigma}}$ and $\tilde{C}_{m,\omega} \coloneqq \frac{C_{m,b}}{\tilde{\sigma}}$.

We can then use the results in (Xu et al., 2020) to state that, for $M \ge \left(\frac{2}{\chi_A} + 2\beta\right)\frac{192C_A^2[1+(\kappa-1)\rho]}{(1-\rho)\chi_A}$ and $\beta \le \min\left\{\frac{\chi_A}{8C_A^2}, \frac{4}{\chi_A}\right\}$,

$$\mathbb{E}[\|\omega^v_{T_c} - \omega^{v*}\|_2^2] \le \left(1 - \frac{\chi_A}{8}\beta\right)^{T_c}\|\omega_0^v - \omega^{v*}\|_2^2 + \left(\frac{2}{\chi_A} + 2\beta\right)\frac{192(C_A^2C_{v,\omega}^2 + C_{v,b}^2)[1+(\kappa-1)\rho]}{(1-\rho)\chi_A M},$$

and

$$\mathbb{E}[\|\omega^m_{T_c} - \omega^{m*}\|_2^2] \le \left(1 - \frac{\chi_A}{8}\beta\right)^{T_c}\|\omega_0^m - \omega^{m*}\|_2^2 + \left(\frac{2}{\chi_A} + 2\beta\right)\frac{192(C_A^2C_{m,\omega}^2 + C_{m,b}^2)[1+(\kappa-1)\rho]}{(1-\rho)\chi_A M}.$$

Also, we will need to express the notion of approximation error for both V^π and M^π. Thus, we define the two following quantities:

$$\xi^v_{appr} \coloneqq \sup_{\theta\in\mathbb{R}^{d_\theta}} \mathbb{E}_{s\sim d_{\mu_0,\pi_\theta}(\cdot)}\left[|V^{\pi_\theta}(s) - \phi(s)^\intercal\omega_\theta^{v*}|^2\right],$$

$$\xi^m_{appr} \coloneqq \sup_{\theta\in\mathbb{R}^{d_\theta}} \mathbb{E}_{s\sim d_{\mu_0,\pi_\theta}(\cdot)}\left[|M^{\pi_\theta}(s) - \phi(s)^\intercal\omega_\theta^{m*}|^2\right].$$

Armed with these bounds and definitions, we proceed with the analysis of the actor. Naturally, the analysis will be similar to what we did in Section A.5, so we will omit any repetitions. We will use the definitions stated at the beginning of that section, but we will modify the first two. In particular, define:

- the TD-error $\delta_\omega(s, a, s') = R^\lambda(s, a, J) + \gamma\hat{V}^\lambda(s', \omega, J) - \hat{V}^\lambda(s, \omega, J)$,
- the *approximated* TD-error $\hat{\delta}_\omega(s, a, s') = R^\lambda(s, a, \hat{J}) + \gamma\hat{V}^\lambda(s', \omega, \hat{J}) - \hat{V}^\lambda(s, \omega, \hat{J})$,

where, for compactness, we use $\omega \coloneqq [\omega^v, \omega^m]$, and accordingly define

$$\hat{V}^\lambda(s, \omega, \hat{J}) = (1 + 2\lambda\hat{J})\phi(s)^\intercal\omega^v - \lambda\phi(s)^\intercal\omega^m - \frac{\lambda}{1-\gamma}\hat{J}^2.$$

Proposition 3 is still applicable in this case, so we start with the following proposition as an analogue to Proposition 4.

Proposition 5. *Suppose Assumptions 1 to 6 hold, then the following holds for Algorithm 2 (when using the factored method utilizing Algorithm 4 to learn V^π and M^π), where $\mathcal{F}_t$ is the filtration on the samples up to iteration t:*

$$\begin{aligned}\mathbb{E}\left[\|\hat{v}_t(\omega_t,\theta_t)-\nabla\eta(\theta_t)\|_2^2|\mathcal{F}_t\right] \le\ & 24\frac{(R_{\lambda,\max}+2C_{fac})^2(1+(k-1)\rho)}{B(1-\rho)}\\ &+36\lambda^2(R_{\max}+2\tilde{C}_{v,\omega})^2\,\mathbb{E}\left[|J_t-\hat{J}_t|^2|\mathcal{F}_t\right]\\ &+36(1+2\lambda R_{\max})^2\|\omega_t^{v*}-\omega_t^v\|_2^2+36\lambda^2\|\omega_t^{m*}-\omega_t^m\|_2^2\\ &+24(1+2\lambda R_{\max})^2\xi_{appr}^v+24\lambda^2\xi_{appr}^m.\end{aligned}$$

where $C_{fac}\coloneqq(1+2\lambda R_{\max})\tilde{C}_{v,\omega}+\lambda\tilde{C}_{m,\omega}+\frac{\lambda}{1-\gamma}R_{\max}^2$, J_t is short for $J_{\pi_{\theta_t}}$, ω_t^{v*} and ω_t^{m*} are the TD fixed points of V^π and M^π respectively for policy π_{θ_t}, and ω_t^v and ω_t^m are their learned estimates.

Proof. In the following we use the compact notation: $\omega_t\coloneqq[\omega_t^v,\omega_t^m]$ and $\omega_t^*\coloneqq[\omega_t^{v*},\omega_t^{m*}]$.

Consider $\|\hat{v}_t(\omega_t,\theta_t)-\nabla\eta(\theta_t)\|_2^2$, we can decompose it in the following way (followed by an application of Lemma 2.ii):

$$\begin{aligned}\|\hat{v}_t(\omega_t,\theta_t)-\nabla\eta(\theta_t)\|_2^2 &= \|\hat{v}_t(\omega_t,\theta_t)\pm v_t(\omega_t^*,\theta_t)\pm g(\omega_t^*,\theta_t)-\nabla\eta(\theta_t)\|_2^2\\ &\le 3\underbrace{\|\hat{v}_t(\omega_t,\theta_t)-v_t(\omega_t^*,\theta_t)\|_2^2}_{(a)}+3\|v_t(\omega_t^*,\theta_t)-g(\omega_t^*,\theta_t)\|_2^2+3\underbrace{\|g(\omega_t^*,\theta_t)-\nabla\eta(\theta_t)\|_2^2}_{(b)}. \quad (15)\end{aligned}$$

We now focus on (a):

$$\begin{aligned}\|\hat{v}_t(\omega_t,\theta_t)-v_t(\omega_t^*,\theta_t)\|_2^2 =& \left\|\frac{1}{B}\sum_{i=0}^{B-1}\psi_{\theta_t}(s_{t,i},a_{t,i})\left[\hat{\delta}_{\omega_t}(s_{t,i},a_{t,i},s'_{t,i+1})-\delta_{\omega_t^*}(s_{t,i},a_{t,i},s'_{t,i+1})\right]\right\|_2^2\\ \le&\frac{1}{B}\sum_{i=0}^{B-1}\underbrace{\|\psi_{\theta_t}(s_{t,i},a_{t,i})\|_2^2}_{\le C_\psi=1}\left|\hat{\delta}_{\omega_t}(s_{t,i},a_{t,i},s'_{t,i+1})-\delta_{\omega_t^*}(s_{t,i},a_{t,i},s'_{t,i+1})\right|^2\\ \le&\frac{1}{B}\sum_{i=0}^{B-1}\Big|R^\lambda(s_{t,i},a_{t,i},\hat{J}_t)-R^\lambda(s_{t,i},a_{t,i},J_t)+\gamma\left(\hat{V}^\lambda(s'_{t,i+1},\omega_t,\hat{J}_t)-\hat{V}^\lambda(s'_{t,i+1},\omega_t^*,J_t)\right)\\ &\qquad-\left(\hat{V}^\lambda(s_{t,i},\omega_t,\hat{J}_t)-\hat{V}^\lambda(s_{t,i},\omega_t^*,J_t)\right)\Big|\\ \le&\frac{1}{B}\sum_{i=0}^{B-1}\Big|R^\lambda(s_{t,i},a_{t,i},\hat{J}_t)-R^\lambda(s_{t,i},a_{t,i},J_t)\\ &\qquad+\gamma\left(\phi(s'_{t,i+1})^\intercal\omega_t^v+2\lambda\hat{J}_t\phi(s'_{t,i+1})^\intercal\omega_t^v-\lambda\phi(s'_{t,i+1})^\intercal\omega_t^m-\frac{\lambda}{1-\gamma}\hat{J}_t^2\right)\\ &\qquad-\gamma\left(\phi(s'_{t,i+1})^\intercal\omega_t^{v*}+2\lambda J_t\phi(s'_{t,i+1})^\intercal\omega_t^{v*}-\lambda\phi(s'_{t,i+1})^\intercal\omega_t^{m*}-\frac{\lambda}{1-\gamma}J_t^2\right)\\ &\qquad+\phi(s_{t,i})^\intercal\omega_t^{v*}+2\lambda J_t\phi(s_{t,i})^\intercal\omega_t^{v*}-\lambda\phi(s_{t,i})^\intercal\omega_t^{m*}-\frac{\lambda}{1-\gamma}J_t^2\\ &\qquad-\phi(s_{t,i})^\intercal\omega_t^v-2\lambda\hat{J}_t\phi(s_{t,i})^\intercal\omega_t^v+\lambda\phi(s_{t,i})^\intercal\omega_t^m+\frac{\lambda}{1-\gamma}\hat{J}_t^2\Big|^2\\ =&\frac{1}{B}\sum_{i=0}^{B-1}\Big|R^\lambda(s_{t,i},a_{t,i},\hat{J}_t)-R^\lambda(s_{t,i},a_{t,i},J_t)\\ &\qquad+(\phi(s_{t,i})^\intercal-\gamma\phi(s'_{t,i+1})^\intercal)(\omega_t^{v*}-\omega_t^v)\end{aligned}$$

$$
\begin{aligned}
&\quad + 2\lambda(J_t\phi(s_{t,i})^\intercal\omega_t^{v*} \mp \hat{J}_t\phi(s_{t,i})^\intercal\omega_t^{v*} - \hat{J}_t\phi(s_{t,i})^\intercal\omega_t^{v}) \\
&\quad + 2\lambda\gamma(\hat{J}_t\phi(s'_{t,i+1})^\intercal\omega_t^{v} \mp \hat{J}_t\phi(s'_{t,i+1})^\intercal\omega_t^{v*} - J_t\phi(s'_{t,i+1})^\intercal\omega_t^{v*}) \\
&\quad + \lambda(\gamma\phi(s'_{t,i+1})^\intercal - \phi(s_{t,i})^\intercal)(\omega_t^{m*} - \omega_t^{m}) + \lambda(\hat{J}_t^2 - J_t^2)\Big|^2 \\
=&\frac{1}{B}\sum_{i=0}^{B-1}\Big|2\lambda R(s_{t,i},a_{t,i})(\hat{J}_t - J_t) + \lambda(J_t^2 - \hat{J}_t^2) \\
&\quad + (1+2\lambda\hat{J}_t)(\phi(s_{t,i})^\intercal - \gamma\phi(s'_{t,i+1})^\intercal)(\omega_t^{v*} - \omega_t^{v}) \\
&\quad + 2\lambda(\phi(s_{t,i})^\intercal - \gamma\phi(s'_{t,i+1})^\intercal)\omega_t^{v*}(J_t - \hat{J}_t) \\
&\quad + \lambda(\gamma\phi(s'_{t,i+1})^\intercal - \phi(s_{t,i})^\intercal)(\omega_t^{m*} - \omega_t^{m}) + \lambda(\hat{J}_t^2 - J_t^2)\Big|^2 \\
=&\frac{1}{B}\sum_{i=0}^{B-1}\Big|2\lambda[R(s_{t,i},a_{t,i}) + (\gamma\phi(s'_{t,i+1})^\intercal - \phi(s_{t,i})^\intercal)\omega_t^{v*}](\hat{J}_t - J_t) \\
&\quad + (1+2\lambda\hat{J}_t)(\phi(s_{t,i})^\intercal - \gamma\phi(s'_{t,i+1})^\intercal)(\omega_t^{v*} - \omega_t^{v}) \\
&\quad + \lambda(\gamma\phi(s'_{t,i+1})^\intercal - \phi(s_{t,i})^\intercal)(\omega_t^{m*} - \omega_t^{m})\Big|^2 \\
\le&\frac{1}{B}\sum_{i=0}^{B-1} 3\Big|2\lambda[R(s_{t,i},a_{t,i}) + (\gamma\phi(s'_{t,i+1})^\intercal - \phi(s_{t,i})^\intercal)\omega_t^{v*}](\hat{J}_t - J_t)\Big|^2 \\
&\quad + 3\Big|(1+2\lambda\hat{J}_t)(\phi(s_{t,i})^\intercal - \gamma\phi(s'_{t,i+1})^\intercal)(\omega_t^{v*} - \omega_t^{v})\Big|^2 \\
&\quad + 3\Big|\lambda(\gamma\phi(s'_{t,i+1})^\intercal - \phi(s_{t,i})^\intercal)(\omega_t^{m*} - \omega_t^{m})\Big|^2 \\
\le&12\lambda^2(R_{\max} + 2\tilde{C}_{v,\omega})^2|\hat{J}_t - J_t|^2 + 12(1+2\lambda R_{\max})^2\|\omega_t^{v*} - \omega_t^{v}\|_2^2 + 12\lambda^2\|\omega_t^{m*} - \omega_t^{m}\|_2^2,
\end{aligned}
$$

where the penultimate inequality is an application of Lemma 2.ii, and the last inequality uses Cauchy-Schwarz, and Assumptions 1.i, 2, and 5.

As for (b), we proceed as follows:

$$
\begin{aligned}
&\|g(\omega_t^*,\theta_t) - \nabla\eta(\theta_t)\|_2^2 \\
&= \Big\| \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu_0,\pi_\theta}(\cdot)\\ a\sim\pi_\theta(\cdot|s)}}[A_{\omega_t^*}(s,a)\psi_{\theta_t}(s,a)] - \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu_0,\pi_\theta}(\cdot)\\ a\sim\pi_\theta(\cdot|s)}}\big[A^\lambda_{\theta_t}(s,a)\psi_{\theta_t}(s,a)\big]\Big\|_2^2 \\
&= \Big\| \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu_0,\pi_\theta}(\cdot)\\ a\sim\pi_\theta(\cdot|s)}}\big[\big(A_{\omega_t^*}(s,a) - A^\lambda_{\theta_t}(s,a)\big)\psi_{\theta_t}(s,a)\big]\Big\|_2^2 \\
&\le \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu_0,\pi_\theta}(\cdot)\\ a\sim\pi_\theta(\cdot|s)}}\Big[\big\|\big(A_{\omega_t^*}(s,a) - A^\lambda_{\theta_t}(s,a)\big)\psi_{\theta_t}(s,a)\big\|_2^2\Big] \\
&= \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu_0,\pi_\theta}(\cdot)\\ a\sim\pi_\theta(\cdot|s)}}\Big[\big|A_{\omega_t^*}(s,a) - A^\lambda_{\theta_t}(s,a)\big|^2 \underbrace{\|\psi_{\theta_t}(s,a)\|_2^2}_{\le C_\psi = 1}\Big] \\
&\le \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu_0,\pi_\theta}(\cdot)\\ a\sim\pi_\theta(\cdot|s)}}\Big[\Big|\mathop{\mathbb{E}}_{s'\sim P(\cdot|s,a)}\Big[R^\lambda(s,a,J_t) + \gamma\hat{V}^\lambda(s',\omega_t^*,J_t) - \hat{V}^\lambda(s,\omega_t^*,J_t)\Big] \\
&\qquad - \mathop{\mathbb{E}}_{s'\sim P(\cdot|s,a)}\big[R^\lambda(s,a,J_t) + \gamma V_t^\lambda(s') - V_t^\lambda(s)\big]\Big|^2\Big]
\end{aligned}
$$

$$
\begin{aligned}
&\leq \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu_0,\pi_\theta}(\cdot)\\ a\sim\pi_\theta(\cdot|s)}}\left[\left|V_t^\lambda(s)-\hat{V}^\lambda(s,\omega_t^*,J_t)+\gamma\mathop{\mathbb{E}}_{s'\sim P(\cdot|s,a)}\left[\hat{V}^\lambda(s',\omega_t^*,J_t)-V_t^\lambda(s')\right]\right|^2\right]\\
&\leq \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu_0,\pi_\theta}(\cdot)\\ a\sim\pi_\theta(\cdot|s)}}\left[\left|(1+2\lambda J_t)(V_t(s)-\phi(s)^\intercal\omega_t^{v*})+\lambda(\phi(s)^\intercal\omega_t^{m*}-M_t(s))\right.\right.\\
&\qquad\qquad\left.\left.+\gamma\mathop{\mathbb{E}}_{s'\sim P(\cdot|s,a)}\left[(1+2\lambda J_t)(\phi(s')^\intercal\omega_t^{v*}-V_t(s'))+\lambda(M_t(s')-\phi(s')^\intercal\omega_t^{m*})\right]\right|^2\right]\\
&\overset{(1)}{\leq} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu_0,\pi_\theta}(\cdot)\\ a\sim\pi_\theta(\cdot|s)}}\left[4|1+2\lambda J_t|^2|V_t(s)-\phi(s)^\intercal\omega_t^{v*}|^2+4\lambda^2|\phi(s)^\intercal\omega_t^{m*}-M_t(s)|^2\right.\\
&\qquad\qquad+4\gamma^2|1+2\lambda J_t|^2\mathop{\mathbb{E}}_{s'\sim P(\cdot|s,a)}\left[|\phi(s')^\intercal\omega_t^{v*}-V_t(s')|^2\right]\\
&\qquad\qquad\left.+4\gamma^2\lambda^2\mathop{\mathbb{E}}_{s'\sim P(\cdot|s,a)}\left[|M_t(s')-\phi(s')^\intercal\omega_t^{m*}|^2\right]\right]\\
&\leq 4(1+2\lambda R_{\max})^2\mathop{\mathbb{E}}_{\substack{s\sim d_{\mu_0,\pi_\theta}(\cdot)\\ a\sim\pi_\theta(\cdot|s)}}\left[|V_t(s)-\phi(s)^\intercal\omega_t^{v*}|^2\right]+4\lambda^2\mathop{\mathbb{E}}_{\substack{s\sim d_{\mu_0,\pi_\theta}(\cdot)\\ a\sim\pi_\theta(\cdot|s)}}\left[|\phi(s)^\intercal\omega_t^{m*}-M_t(s)|^2\right]\\
&\quad+4\gamma(1+2\lambda R_{\max})^2\mathop{\mathbb{E}}_{\substack{s\sim d_{\mu_0,\pi_\theta}(\cdot)\\ a\sim\pi_\theta(\cdot|s)}}\left[\gamma\mathop{\mathbb{E}}_{s'\sim P(\cdot|s,a)}\left[|\phi(s')^\intercal\omega_t^{v*}-V_t(s')|^2\right]\right]\\
&\quad+4\gamma\lambda^2\mathop{\mathbb{E}}_{\substack{s\sim d_{\mu_0,\pi_\theta}(\cdot)\\ a\sim\pi_\theta(\cdot|s)}}\left[\gamma\mathop{\mathbb{E}}_{s'\sim P(\cdot|s,a)}\left[|M_t(s')-\phi(s')^\intercal\omega_t^{m*}|^2\right]\right].\\
&\leq 4(1+2\lambda R_{\max})^2\mathop{\mathbb{E}}_{\substack{s\sim d_{\mu_0,\pi_\theta}(\cdot)\\ a\sim\pi_\theta(\cdot|s)}}\left[|V_t(s)-\phi(s)^\intercal\omega_t^{v*}|^2\right]+4\lambda^2\mathop{\mathbb{E}}_{\substack{s\sim d_{\mu_0,\pi_\theta}(\cdot)\\ a\sim\pi_\theta(\cdot|s)}}\left[|\phi(s)^\intercal\omega_t^{m*}-M_t(s)|^2\right]\\
&\quad+4\gamma(1+2\lambda R_{\max})^2\mathop{\mathbb{E}}_{\substack{s\sim d_{\mu_0,\pi_\theta}(\cdot)\\ a\sim\pi_\theta(\cdot|s)}}\left[\gamma\mathop{\mathbb{E}}_{s'\sim P(\cdot|s,a)}\left[|\phi(s')^\intercal\omega_t^{v*}-V_t(s')|^2\right]+(1-\gamma)\mathop{\mathbb{E}}_{s'\sim\mu_0(\cdot)}\left[|\phi(s')^\intercal\omega_t^{v*}-V_t(s')|^2\right]\right]\\
&\quad+4\gamma\lambda^2\mathop{\mathbb{E}}_{\substack{s\sim d_{\mu_0,\pi_\theta}(\cdot)\\ a\sim\pi_\theta(\cdot|s)}}\left[\gamma\mathop{\mathbb{E}}_{s'\sim P(\cdot|s,a)}\left[|M_t(s')-\phi(s')^\intercal\omega_t^{m*}|^2\right]+(1-\gamma)\mathop{\mathbb{E}}_{s'\sim\mu_0(\cdot)}\left[|M_t(s')-\phi(s')^\intercal\omega_t^{m*}|^2\right]\right].\\
&\overset{(2)}{=} 4(1+2\lambda R_{\max})^2\mathop{\mathbb{E}}_{\substack{s\sim d_{\mu_0,\pi_\theta}(\cdot)\\ a\sim\pi_\theta(\cdot|s)}}\left[|V_t(s)-\phi(s)^\intercal\omega_t^{v*}|^2\right]+4\lambda^2\mathop{\mathbb{E}}_{\substack{s\sim d_{\mu_0,\pi_\theta}(\cdot)\\ a\sim\pi_\theta(\cdot|s)}}\left[|\phi(s)^\intercal\omega_t^{m*}-M_t(s)|^2\right]\\
&\quad+4\gamma(1+2\lambda R_{\max})^2\mathop{\mathbb{E}}_{\substack{s\sim d_{\mu_0,\pi_\theta}(\cdot)\\ a\sim\pi_\theta(\cdot|s)}}\left[\mathop{\mathbb{E}}_{s'\sim\tilde{P}(\cdot|s,a)}\left[|\phi(s')^\intercal\omega_t^{v*}-V_t(s')|^2\right]\right]\\
&\quad+4\gamma\lambda^2\mathop{\mathbb{E}}_{\substack{s\sim d_{\mu_0,\pi_\theta}(\cdot)\\ a\sim\pi_\theta(\cdot|s)}}\left[\mathop{\mathbb{E}}_{s'\sim\tilde{P}(\cdot|s,a)}\left[|M_t(s')-\phi(s')^\intercal\omega_t^{m*}|^2\right]\right].\\
&\overset{(3)}{\leq} 8(1+2\lambda R_{\max})^2\xi_{appr}^v+8\lambda^2\xi_{appr}^m,
\end{aligned}
$$

where (1) is an application of Lemma 2.ii and Jensen's inequality, (2) holds by the definition of the modified transition kernel $\tilde{P}$, and (3) holds by the definitions of ξ_{appr}^v and ξ_{appr}^m, and the fact that d_{μ_0,π_θ} is the stationary distribution of the Markov chain with the modified transition kernel $\tilde{P}$ when acting with policy π_θ (Xu et al., 2020).

Substituting back to inequality (15) and taking the expectation w.r.t. the filtration $\mathcal{F}_t$, we get:

$$\begin{aligned}\mathbb{E}\left[\|\hat{\nu}_t(\omega_t,\theta_t)-\nabla\eta(\theta_t)\|_2^2|\mathcal{F}_t\right] \leq\ & 3\,\mathbb{E}\left[\|v_t(\omega_t^*,\theta_t)-g(\omega_t^*,\theta_t)\|_2^2|\mathcal{F}_t\right] \\ & + 36\lambda^2(R_{\max}+2\tilde{C}_{v,\omega})^2\,\mathbb{E}\left[|J_t-\hat{J}_t|^2|\mathcal{F}_t\right] \\ & + 36(1+2\lambda R_{\max})^2\|\omega_t^{v*}-\omega_t^v\|_2^2+36\lambda^2\|\omega_t^{m*}-\omega_t^m\|_2^2 \\ & + 24(1+2\lambda R_{\max})^2\xi_{appr}^v+24\lambda^2\xi_{appr}^m.\end{aligned}$$

To bound the first term on the RHS, we follow the related passage in the proof of Theorem 5 in Xu et al. (2020) to have:

$$\mathbb{E}\left[\|v_t(\omega_t^*,\theta_t)-g(\omega_t^*,\theta_t)\|_2^2|\mathcal{F}_t\right]\leq\frac{8(R_{\lambda,\max}+2C_{fac})^2(1+(k-1)\rho)}{B(1-\rho)},$$

where $C_{fac}\coloneqq(1+2\lambda R_{\max})\tilde{C}_{v,\omega}+\lambda\tilde{C}_{m,\omega}+\frac{\lambda}{1-\gamma}R_{\max}^2$, which serves as an upper bound on $\hat{V}^\lambda(s,\omega_t^*,J_t)$. □

We then conclude with the following theorem.

Theorem 5 (Bound for MVAC with Factored Critic)**.** *Suppose Assumptions 1 to 6 hold, and suppose we run Algorithm 2 for T iterations while using the factored method and utilizing Algorithm 4 to learn V^π and M^π, then if $\alpha=\frac{1}{8L_\eta}$, we have:*

$$\begin{aligned}\mathbb{E}\left[\left\|\nabla\eta(\theta_{\hat{T}})\right\|_2^2\right]\leq\ & \frac{64L_\eta R_{\lambda,\max}}{T}+108(1+2\lambda R_{\max})^2\frac{\sum_{t=0}^{T-1}\mathbb{E}[\|\omega_t^{v*}-\omega_t^v\|_2^2]}{T} \\ & +108\lambda^2\frac{\sum_{t=0}^{T-1}\mathbb{E}[\|\omega_t^{m*}-\omega_t^m\|_2^2]}{T}+\xi_{distr}^{fac}+\xi_J^{fac}+\xi_{appr}^{fac},\end{aligned}$$

where ω_t^{v*} and ω_t^{m*} are the TD fixed points of V^π and M^π respectively for policy π_{θ_t}, and ω_t^v and ω_t^m are their learned estimates. Furthermore:

$$\xi_{distr}^{fac}\coloneqq 72\frac{(R_{\lambda,\max}+2C_{fac})^2(1+(k-1)\rho)}{B(1-\rho)},$$

$$\xi_{appr}^{fac}\coloneqq 72(1+2\lambda R_{\max})^2\xi_{appr}^v+72\lambda^2\xi_{appr}^m,$$

$$\xi_J^{fac}\coloneqq 108\lambda^2(R_{\max}+2\tilde{C}_{v,\omega})^2\left(\gamma^{2T_J}R_{\max}^2+\frac{R_{\max}^2}{L}\right).$$

Proof. The result can be obtained by following the same steps of the proof of Theorem 4, but using the results of Proposition 5 instead of Proposition 4. □

From the result of the last theorem, one can obtain the same sample complexity (i.e., $\mathcal{O}\left(\epsilon^{-2}\log\left(\epsilon^{-1}\right)\right)$) as that shown in Corollary 4.1 for MVAC when using the direct method.

C Reusing Samples

Our results showed that we can achieve, with our modified objective, the same sample complexity as the risk neutral case. However, estimating the expected return still requires an extra batch of samples per iteration. Towards alleviating this extra burden, we briefly discuss two possible approaches for future work, and highlight their challenges.

C.1 Estimating the Expected Return and the Critic Using the Same Data

We can consider a variant of the algorithm in which we use the trajectories collected to estimate J as the mini-batches for training the critic. One negative consequence of the naivety of this approach is that the derived sample complexity of the algorithm will be worsened since we are forcing the number of steps in the trajectories to be the same as the mini-batch size of the critic. Nonetheless, it is a simple approach that allows us to see the main challenges of reusing samples in our algorithm. To this end, we will only focus in this section on the

direct mini-batch TD algorithm. More specifically, it suffices to discuss the issues that arise when attempting to bound the following quantity (at the t^{th} iteration of the critic, which uses as mini-batch the t^{th} trajectory used for estimating $\hat{J}$):

$$\mathbb{E}\left[\left\|\hat{b}_t(\hat{J}) - b(\hat{J})\right\|_2^2\right]. \tag{16}$$

We remind the reader that $b(\hat{J}) = \mathbb{E}_{\substack{s\sim\mu_{\theta_t}(\cdot)\\ a\sim\pi_{\theta_t}(\cdot|s)}}\left[\phi(s)R^\lambda(s,a,\hat{J})\right]$, where ϕ is a feature mapping for the states, μ_θ is the stationary state distribution of policy π_θ, and $R^\lambda(s,a,\hat{J}) = R(s,a) - \lambda(R(s,a) - \hat{J})^2$. Furthermore, $\hat{b}_t(\hat{J}) = \frac{1}{T_J}\sum_{i=0}^{T_J-1} b_{t,i}(\hat{J})$ is the average (over the mini-batch) of $b_{t,i}(\hat{J}) = \phi(s_{t,i})R^\lambda(s_{t,i},a_{t,i},\hat{J})$, where $(s_{t,i},a_{t,i})$ is the state-action pair in the i^{th} step of the t^{th} mini-batch (or, in this case, the t^{th} trajectory). The risk-neutral equivalent of (16) (i.e., with $R(s,a)$ instead of $R^\lambda(s,a,\hat{J})$, and thus no dependence on $\hat{J}$) is bounded in Lemma 2 in (Xu et al., 2020) in terms of the mini-batch size and the mixing properties of the MDP. However, since the expectation is also over $\hat{J}$, the same procedure cannot be applied in our case. Even if we condition on $\hat{J}$ (like we do at some point in the proof of Theorem 3) and attempt to mirror the derivation in (Xu et al., 2020), we will not reach a form that will allow us to leverage the mixing assumption. This is because we have conditioned on $\hat{J}$, which conveys extra information about the distribution of the states encountered in the trajectory as it is one of the trajectories used in the calculation of $\hat{J}$ in the first place. We thus need to take a different route.

One such route is an information-theoretical one, which is inspired by the approach adopted in (Bhandari et al., 2018) to analyze TD learning with Markovian sampling. Denote the t^{th} trajectory by $\tau_t = (s_{t,0}, a_{t,0}, s_{t,1}, a_{t,1}, \dots, s_{t,T_J})$. Note that the random variables $\hat{J}$ and τ_t (for any $t \in \{1,\dots,N\}$) are not independent. Define

$$v(\tau_t, \hat{J}) \coloneqq \left\|\hat{b}_t(\hat{J}) - b(\hat{J})\right\|_2^2,$$

which is a deterministic function of τ_t and $\hat{J}$. We can then rewrite (16) as $\mathbb{E}[v(\tau_t,\hat{J})]$. Let τ_t' and $\hat{J}'$ be two independent copies of τ_t and $\hat{J}$. That is:

$$P(\tau_t' = \cdot, \hat{J}' = \cdot) = P(\tau_t = \cdot)P(\hat{J} = \cdot).$$

By adding and subtracting $\mathbb{E}[v(\tau_t',\hat{J}')]$, we can rewrite (16) as:

$$\mathbb{E}[v(\tau_t,\hat{J})] - \mathbb{E}[v(\tau_t',\hat{J}')] + \mathbb{E}[v(\tau_t',\hat{J}')].$$

Due to the Independence of τ_t' and $\hat{J}'$, we can use Lemma 2 in (Xu et al., 2020) to bound the last term (after conditioning on $\hat{J}'$). As for the first two terms, we can leverage the following variational representation of the total variation distance (Bhandari et al. (2018) or Theorem 6.3. in Ajjanagadde et al. (2017)):

$$D_{\text{TV}}(P\|Q) = \frac{1}{2}\sup_{\|f\|_\infty\le 1}\left|\mathop{\mathbb{E}}_{P}[f(x)] - \mathop{\mathbb{E}}_{Q}[f(x)]\right|.$$

And thus,

$$\mathbb{E}[v(\tau_t,\hat{J})] - \mathbb{E}[v(\tau_t',\hat{J}')] \le 2\|v\|_\infty D_{\text{TV}}(p_{(\tau_t,\hat{J})}\|p_{\tau_t}p_{\hat{J}}).$$

The $\|v\|_\infty$ coefficient can be easily bounded using some constants that we defined in the analysis of the original algorithm. The problematic part is obviously the total variation distance. One way to deal with it is to relate it to the KL-divergence. This can be either done using Pinsker's inequality:

$$D_{\text{TV}}(P\|Q) \le \sqrt{\frac{1}{2}D_{\text{KL}}(P\|Q)},$$

or by this inequality (Inequality (2.25) in Tsybakov (2008)):

$$D_{\text{TV}}(P\|Q) \le 1 - \frac{1}{2}e^{-D_{\text{KL}}(P\|Q)}.$$

The latter is upper bounded by 1, so it provides a non-vacuous bound for the total variation even if the KL-divergence is large (unlike Pinsker's inequality). However, it is lower bounded by $\frac{1}{2}$, which makes it unsuitable if we are to show that the bound can be reduced arbitrarily by using enough samples. In any case, our focus now

becomes bounding $D_{\mathrm{KL}}(p_{(\tau_t,\hat{J})}\|p_{\tau_t}p_{\hat{J}})$, which is the same as the mutual information between τ_t and $\hat{J}$: $\mathrm{I}(\tau_t;\hat{J})$. To simplify the problem, we can construct the following Markov chain[25]:

$$\tau_t \to G_t \to \sum_{i=1}^{N} G_i \to \hat{J}.$$

We can then use the data processing inequality to deduce that

$$\mathrm{I}(\tau_t;\hat{J}) \leq \mathrm{I}\left(G_t;\sum_{i=1}^{N} G_i\right).$$

That is, we have reduced our task to bounding the mutual information between a sum of N i.i.d. random variables and one of these random variables. Note that choosing to work with the sum instead of the average is just for convenience; both the sum and the average provide the same information about the individual samples. The ideal goal would be to show that this mutual information term can be reduced arbitrarily by making N large enough.

Before proceeding, we can frame the problem in more generic terms. Let $Y = X_0 + X_1 + \ldots + X_{N-1}$, where $X_0, \ldots, X_{N-1}$ are (continuous) i.i.d. random variables. Furthermore, let $Z = X_1 + \ldots + X_{N-1}$ such that $Y = X_0 + Z$. The goal is to bound $\mathrm{I}(X_0;Y)$. We start by the following decomposition (Madiman, 2008):

$$\begin{aligned}\mathrm{I}(X_0;Y) &= H(Y) - H(Y|X_0)\\ &= H(Y) - H(Z),\end{aligned} \tag{17}$$

where $H(\cdot)$ is the differential entropy. The next step is to bound $H(Y)$, but first we highlight some relevant terminology adapted from (Madiman, 2008). Let $[N]$ be a shorthand for $\{1,\ldots,N\}$. Moreover, for a collection C of subsets of $[N]$, define:

- A fractional covering $\alpha : C \to \mathbb{R}_+$ is a function that associates to every set in C, a number in $\mathbb{R}_+$ such that $\forall i \in [N] : \sum_{s\in C: i\in s} \alpha_s \geq 1$.
- For any $i \in [N]$, its degree is defined as $r(i) = |\{s \in C : i \in s\}|$, i.e. the number of sets that i belongs to.
- For a set $s \in C$, $r_-(s) = \min_{i\in s} r(i)$ (the minimum degree among the elements of s). One way to obtain a simple fractional covering function is to set $\alpha_s = \frac{1}{r_-(s)}$ (Madiman, 2008).

For any collection C of subsets of $[N-1]$ and any associated fractional covering α, we have the following bound (Theorem 3 in Madiman (2008)):

$$H(Y) = H\left(X_0 + \sum_{i\in[N-1]} X_i\right) \leq \sum_{s\in C} \alpha_s H\left(X_0 + \sum_{i\in s} X_i\right) - \left(\sum_{s\in C}\alpha_s - 1\right)H(X_0).$$

In particular, we define C to be the collection of all subsets of $[N-1]$ with $N-2$ elements. There are $N-1$ such sets, and each element occurs in $N-2$ sets. Thus, if we set $\alpha_s = \frac{1}{r_-(s)}$, then $\alpha_s = \frac{1}{N-2}$ for every $s \in C$, and $\sum_{s\in C}\alpha_s = \frac{N-1}{N-2}$. We can then write

$$H(Y) \leq \sum_{s\in C} \frac{1}{N-2} H\left(X_0 + \sum_{i\in s} X_i\right) - \left(\frac{N-1}{N-2} - 1\right)H(X_0),$$

for $N > 2$. Since $X_0, \ldots, X_{N-1}$ are i.i.d., the specific labels in the first entropy term on the R.H.S. are irrelevant; all that matters is that we have $N-1$ *different* i.i.d. random variables. We can then simplify the bound in the following way:

$$H(Y) \leq \frac{N-1}{N-2}H(Z) - \left(\frac{N-1}{N-2} - 1\right)H(X_0).$$

[25] In the following, we use N instead of L to denote the number of trajectories. This way, the notation is closer to that in (Madiman, 2008), from which we will use some results.

Therefore, from (17), we have that

$$\begin{aligned}\mathrm{I}(X_0;Y) &\leq \left(\frac{N-1}{N-2}-1\right)H(Z) - \left(\frac{N-1}{N-2}-1\right)H(X_0)\\ &= \frac{1}{N-2}(H(Z)-H(X_0)),\end{aligned}$$

for $N > 2$. We note the following property of differential entropy (first statement in Theorem 1.6. in Ajjanagadde et al. (2017)): if W is a random variable taking values only in the interval $[-M, +M]$, then

$$H(W) \leq \log(2M).$$

Then, if we assume that our random variables $X_0, \ldots, X_{N-1}$ take values only in the interval $[-M, +M]$, we can then write:

$$\begin{aligned}\mathrm{I}(X_0;Y) &\leq \frac{1}{N-2}(\log(2(N-1)M) - H(X_0))\\ &= \frac{1}{N-2}(\log(2(N-1)) + \log(M) - H(X_0)),\end{aligned}$$

for $N > 2$. Another property of differential entropy (second statement in Theorem 1.6. in Ajjanagadde et al. (2017)) is that $H(aW) = H(W) + \log|a|$, where W is a random variable and a is a scalar. Denote by $\tilde{X}_0$ a normalized version of X_0 taking value between $[-1, 1]$, then

$$H(X_0) = H(M\tilde{X}_0) = H(\tilde{X}_0) + \log(M).$$

Then we have that:

$$\begin{aligned}\mathrm{I}(X_0;Y) &\leq \frac{1}{N-2}\Big(\log(2(N-1)) + \log(M) - H(\tilde{X}_0) - \log(M)\Big)\\ &= \frac{\log(2(N-1)) - H(\tilde{X}_0)}{N-2},\end{aligned}$$

for $N > 2$. Note that differential entropy can be negative, so we cannot eliminate the negative entropy term from the numerator. We can see from the last expression that, as long as $H(\tilde{X}_0)$ is finite, the mutual information can be reduced arbitrarily by making N large enough.

Returning to our original problem, we now have that

$$\mathrm{I}(\tau_t; \hat{J}) \leq \frac{\log(2(N-1)) - H(\tilde{G}_t)}{N-2}.$$

And consequently

$$\mathbb{E}[v(\tau_t, \hat{J})] - \mathbb{E}[v(\tau'_t, \hat{J}')] \leq 2\|v\|_\infty \sqrt{\frac{1}{2}\frac{\log(2(N-1)) - H(\tilde{G}_t)}{N-2}},$$

for $N > 2$, which is added as an extra term to the bound that we can obtain from Lemma 2 in (Xu et al., 2020).

The main issue with this result is the presence of the negative differential entropy term, which is likely an artifact of the adopted methodology. More specifically, we need to understand what properties of the MDP and/or the policies affect this term. At the very least, we need to impose sufficient assumptions on the MDP and/or the policies to ensure that this term is finite, lest we end up with a vacuous bound. Another side effect is that the derived sample complexity of the algorithm is worsened yet again due to the square root in Pinsker's equality. One has to wonder if this is necessary, or again an artifact of the performed analysis.

C.2 Leveraging Previous Expected Return Estimates

Another possible approach consists in still reusing the trajectories collected for estimating J as mini-batches for the critic, but at each iteration of the algorithm, employing the estimated J of the previous iteration, instead of the current one. This can help mitigate some of the issues highlighted in the previous section. However, doing so

introduces an extra source of error, which depends on how much the expected return changes between successive iterations. This difference, in turn, is proportional to the difference between the parameters of the two successive policies, thanks to the Lipschitz continuity assumption. In practice, to control this extra error, we can try to guarantee that the policy updates diminish with the number of iterations. One way to achieve this is to use a decreasing step-size, which might, however, negatively affect the sample complexity of our algorithm.

As an alternative (only concerned with the estimation of J, without necessarily reusing samples) one could use less trajectories for estimating J, but employ an incremental averaging scheme across iterations, controlled with a step-size parameter. This can, in effect, allow us to take advantage of the sampled trajectories from previous iterations as well. Still, to understand the effect of this approach, one needs to understand how much the expected return of the policies changes between successive iterations.

Adopting a schedule of step-sizes which allows to take advantage of these ideas, while minimizing the negative effects, is an interesting direction for future work.

D Details of the Experiments

D.1 Description of the Environment

We now provide the exact description of the dynamics of the Point-Reacher environment. As mentioned before, the agent controls a point mass that moves along the real line in the interval $[-10, 10]$, by taking (continuous) actions in $[-2, 2]$, denoting the size and the direction of the desired step. If the agent is in state s and takes action a, the new state[26] is $s' \sim \mathcal{N}(s + a, \sqrt[4]{|a| + 0.01})$ and the immediate reward is $\mathrm{r} = -[\sqrt[4]{0.1|s'|} + 0.25p]$, where $p \sim \mathcal{N}(0, |a|^3)$. The initial state is drawn uniformly in $[-5, -4.9] \cup [4.9, 5]$. The version we considered is of the continuing type. One can see that taking larger steps can, on average, take the agent faster towards the 0 point (around which the agent can, on average, collect higher rewards). However, taking larger steps also leads to higher variance of the immediate reward and the reached next state (which also affects the immediate reward).

D.2 Implementation Details

We detail here some of the implementation details of MVAC. As mentioned in the main paper, we considered Gaussian policies, where the mean and standard deviation are linear functions of the states. The features we used for the states are Gaussian radial basis functions with 4 means spread uniformly over the state space, and a width of 8. The critic also used these same features, but with a width of 10. The following is a summary of the used parameters:

- Discount factor: $\gamma = 0.9$.
- Number of trajectories for estimating J: $L = 100$.
- Length of the trajectories for estimating J: $T_J = 50$.
- Critic batch size: $M = 10$.
- Number of critic iterations: $T_c = 60$.
- Critic step-size: $\beta = 0.1$.
- Actor batch size: $B = 400$.
- Number of actor iterations: $T = 500$.
- Actor step-size: $\alpha = 0.03$.

One enhancement that was adopted for the actor is the usage of *root mean square propagation* (RMSprop) for smoother and faster learning.

[26] The new state is clipped back into $[-10, 10]$ if it gets outside this interval.

The approximated Pareto frontiers in plot (f) of Figure 1 were obtained by plotting the expected return and the reward volatility (or return variance in the case of plot (i)) corresponding to the points with the highest mean-volatility (on the averaged curves) for each of the six values of λ chosen uniformly between 0 and 1.2.