On The Convergence Of Policy Iteration-Based Reinforcement Learning With Monte Carlo Policy Evaluation

Anna Winnicki
University of Illinois Urbana-Champaign

R. Srikant
University of Illinois Urbana-Champaign

Abstract

A common technique in reinforcement learning is to evaluate the value function from Monte Carlo simulations of a given policy, and use the estimated value function to obtain a new policy which is greedy with respect to the estimated value function. A well-known longstanding open problem in this context is to prove the convergence of such a scheme when the value function of a policy is estimated from data collected from a single sample path obtained from implementing the policy (see page 99 of [Sutton and Barto, 2018], page 8 of [Tsitsiklis, 2002]). We present a solution to the open problem by showing that a first-visit version of such a policy iteration scheme indeed converges to the optimal policy provided that the policy improvement step uses lookahead [Silver et al., 2016, Mnih et al., 2016, Silver et al., 2017b] rather than a simple greedy policy improvement. We provide results both for the original open problem in the tabular setting and also present extensions to the function approximation setting, where we show that the policy resulting from the algorithm performs close to the optimal policy within a function approximation error.

1 INTRODUCTION

In many applications of reinforcement learning, the underlying probability transition matrix is known but the size of the state space is large so that one uses approximate dynamic programming methods to obtain the optimal control policy. Examples of such applications include game-playing RL agents for playing games such as Chess and Go. Abstracting away the details, in essence what AlphaZero does is the following [Silver et al., 2017b]: it evaluates the current policy using a Monte Carlo rollout and obtains a new policy using the estimate of the value function of the old policy by using lookahead. We note that AlphaZero collects and uses Monte Carlo returns for all states in each rollout [Silver et al., 2017b]. Thus, effectively the algorithm performs policy iteration using Monte Carlo estimates of the value function. If one ignores the Monte Carlo aspect of policy evaluation but is interested in the tree search of aspects of rollout and lookahead, there are several recent works which quantify the impact of the depth of rollout and lookahead on the performance of algorithm [Efroni et al., 2019, Efroni et al., 2018a, Winnicki et al., 2021]. However, to the best of our knowledge, there is no analysis of Monte Carlo policy evaluation when the estimates of the value function are obtained from trajectories simulated from the policy. To the best of our knowledge, the only analysis of such algorithms assume that, at each iteration, either one estimates the value function starting from every single state of the underlying MDP [Tsitsiklis, 2002] or from a subset of fixed states [Winnicki and Srikant, 2022]. In fact, studying Monte Carlo policy evaluation using a single trajectory from each policy at each step of policy iteration is a known open problem [Sutton and Barto, 2018, Tsitsiklis, 2002, Sutton and Barto, 1998]. In this paper, we take a significant step in solving this problem: we prove that, with sufficient lookahead, policy iteration and Monte Carlo policy evaluation does indeed converge provided we use sufficient lookahead during the policy improvement step.

1.1 Main Contributions

Our paper has two main contributions.

Convergence of Monte Carlo ES We prove the convergence of Monte Carlo based policy iteration where a single trajectory corresponding to each policy is used at each iteration to generate returns, or empirical sums of costs that estimate the value function, for states visited by the trajectory. A formulation of this algorithm, which is called

Proceedings of the 26th International Conference on Artificial Intelligence and Statistics (AISTATS) 2023, Valencia, Spain. PMLR: Volume 206. Copyright 2023 by the author(s).

Monte Carlo with Exploring Starts (Monte Carlo ES), can be found on page 99 of [Sutton and Barto, 2018], and its convergence is "one of the most fundamental open theoretical questions in reinforcement learning" (page 99 of [Sutton and Barto, 2018]). See the Appendix for more on the connection of Monte Carlo ES to practice. The work of [Tsitsiklis, 2002] partially solves a variant of Monte Carlo ES, but the results assume a setting that is a hybrid of Monte Carlo sampling using a single trajectory and a generative model. Hence, the convergence of Monte Carlo ES, as well as related variants such as the "every visit" version [Singh and Sutton, 1996], remains an open problem. A major objective of this work is to solve the open problem.

Modern methods that use policy iteration based algorithms with Monte Carlo methods of policy evaluation have achieved spectacular empirical success in problems with very large state spaces [Mnih et al., 2016, Silver et al., 2017b, Silver et al., 2017a] using lookahead policies computed using Monte Carlo Tree Search (MCTS) as opposed to one-step greedy policies. The motivation behind using the lookahead is to significantly speed up the rate of convergence of the algorithms. The benefits of using MCTS to compute lookahead policies versus one step greedy policies far outweigh the additional computational overhead which is relatively small when the number of next states and actions is small, which is the case in many problems such as chess and Go. One of our main results shows that, with the use of lookahead, Monte Carlo ES converges asymptotically. We also provide finite-sample error bounds for the algorithms. Since the prior statement of the open problem is in the tabular setting, we present the results for that case. We then extend the results to the case where function approximation is use. Examples of such applications include game-playing RL agents for playing games such as Chess and Go.

Extension To Linear Function Approximation Beyond settling the open problem by using lookahead, we also extend the result to the case where one uses feature vectors to approximate the value function. We show that when the lookahead is sufficiently large, there is convergence to within a function approximation error. We also provide interpretable finite-sample convergence guarantees.

Then, we show that our techniques can be easily extended to incorporate other algorithms for policy evaluation with feature vectors that have recently been analyzed. For techniques where the mean square error is known such as TD learning with linear function approximation [Srikant and Ying, 2019, Bhandari et al., 2018], we show that the approximation error is approximately equal to the mean square error corresponding to the policy evaluation method with feature vectors. Analogously to the previous extension, we show that when the number of steps of TD learning is very large, the error primarily depends on the function approximation error due to the feature vectors.

When feature vectors are used, recent approximate policy iteration algorithms have a bound on the error in approximate policy iteration as a function of the discount factor α of $1/(1-\alpha)^2$ (see [Bertsekas, 2019, Winnicki et al., 2021, Lagoudakis and Parr, 2003]). When α is very close to 1, which is often the case in practice, reducing the bound by a factor of $1/(1-\alpha)$ significantly improves the performance of the algorithms. In our algorithms, our bounds are approximately of the order $1/(1-\alpha^{H-1})(1-\alpha)$, where H is the amount of lookahead.

1.2 Related Works

The connection between Monte Carlo methods and control methods based on policy iteration has been widely studied [Sutton and Barto, 2018, Singh and Sutton, 1996]. The work of [Tsitsiklis, 2002] studies Monte Carlo sampling with infinitely long trajectories beginning at all states or all states with regular frequencies to perform policy iteration. The works of [Chen, 2018, Liu, 2020] study a similar method in the setting of the stochastic shortest path problem. A related result has been obtained in [Wang et al., 2022, Lubars et al., 2021] under the strong assumption that for the optimal policy the transient states of the resulting Markov chain form an acyclic graph.

Monte Carlo methods with infinitely long trajectories and fixed starting states to perform approximate policy iteration with feature vectors for function approximation was studied in [Winnicki and Srikant, 2022]. The use of rollouts to produce an m-step return, where m is the partial evaluation parameter in the Monte Carlo simulation, as opposed to infinitely long trajectories, has been studied in [Puterman and Shin, 1978, Tsitsiklis and Van Roy, 1997, Efroni et al., 2019, Winnicki et al., 2021] (see Section 2 for definitions of return and rollout). More broadly speaking, these methods form a subset of approximate policy iteration algorithms that have been extensively studied; see [Bertsekas and Tsitsiklis, 1996, Bertsekas, 2019, Puterman and Shin, 1978] for results on dynamic programming and [Lesner and Scherrer, 2015, Efroni et al., 2020, Tomar et al., 2020, Efroni et al., 2018b, Deng et al., 2020] for applications to reinforcement learning.

The work of [Efroni et al., 2019] uses rollouts in the algorithms for policy evaluation along with multiple-step greedy policies, also known as lookahead policies, which have been featured in recent prominent implementations [Mnih et al., 2016, Silver et al., 2016, Silver et al., 2017b]. The work of [Winnicki et al., 2021] defines the necessity of depth of lookahead and amount of return required for approximate policy iteration as a function of the feature vectors and quanities bounds on the asympotic error. Here, we build upon the work of [Winnicki et al., 2021] and further strengthen the bounds using stochastic ap-

proximation as well as expand the setting of the problem to more carefully understand the role lookahead plays on an algorithm that requires only a single trajectory for each policy at each iteration for convergence. The work of [Winnicki and Srikant, 2022] provides a partial connection to the work of [Winnicki et al., 2021] and the present work as it incorporates stochastic approximation but only in a partially generative model setting, similar to the one in [Winnicki et al., 2021]. See [Bertsekas, 2011, Bertsekas, 2019] for more on feature vectors in approximate policy iteration. The works of [Bertsekas, 2011] and [Bertsekas, 2019] also study a variant of policy iteration wherein a greedy policy is evaluated approximately using feature vectors at each iteration.

When the model of the state space is not known, lookahead policies are computed using the Monte Carlo Tree Search (MCTS) algorithm, which has been studied in [Shah et al., 2020, Ma et al., 2019, Munos, 2014, Browne et al., 2012, Kocsis and Szepesvári, 2006, Efroni et al., 2018b, Powell, 2021]. For more on the use of tree search in RL algorithms, see [Bertsekas, 2019, Baxter et al., 1999, Veness et al., 2009, Lanctot et al., 2014]. Lookahead also bears much relationship to Model Predictive Control (MPC) [Bertsekas, 2022].

Our algorithms involve a general framework which allows for general methods of policy evaluation using feature vectors followed by policy improvement using lookahead. See [Srikant and Ying, 2019, Bhandari et al., 2018] for more on policy evaluation with feature vectors.

2 BACKGROUND ON REINFORCEMENT LEARNING

We consider a finite-state finite-action Markov decision process (MDP). The state space is $\mathcal{S}$ and has cardinality $|\mathcal{S}|$. The action space is $\mathcal{A}$ and has size $|\mathcal{A}|$. The probability of transitioning to state j from state i when action u is taken is $P_{ij}(u)$. The associated cost is $g(i,u)$. We assume $g(i,u) \in [0,1]\ \forall i, u$, with probability 1.

Policy $\mu : \mathcal{S} \to \mathcal{A}$ prescribes an action to take at state $i \in \mathcal{S}$. When a policy μ is fixed, we denote by $g_\mu \in \mathbb{R}^{|\mathcal{S}|}$ the vector of expected costs associated with policy μ. We call P_μ the probability transition matrix corresponding to the associated Markov chain. At time k, we call the state of the Markov chain x_k. Consider policy μ. The associated value function with discount factor $\alpha \in (0,1)$ is given by J^μ defined as follows:

$$J^\mu(i) := E[\sum_{k=0}^{\infty} \alpha^k g(x_k, \mu(x_k)) | x_0 = i] \quad \forall i \in \mathcal{S}.$$

Herein, we assume that $\alpha \in (0,1)$ for all discount factors α. It is well known that J^μ solves the associated Bellman equation:

$$J^\mu = g_\mu + \alpha P_\mu J^\mu.$$

The associated Bellman operator, $T_\mu : \mathcal{S} \to \mathcal{S}$, is defined as follows:

$$T_\mu J = g_\mu + \alpha P_\mu J.$$

When we apply the Bellman operator to vector J, the result is called $T_\mu J$, which has the following property:

$$\|T_\mu J - J^\mu\|_\infty \leq \alpha \|J - J^\mu\|_\infty.$$

If operator T_μ is applied m times to vector $J \in \mathbb{R}^{|\mathcal{S}|}$, then we say that we have performed an m-step rollout of the policy μ and the result $T_\mu^m J$ of the rollout is called the return. See [Winnicki et al., 2021] for more on rollout.

Our objective is to find a policy μ which minimizes the expected discounted cost:

$$E[\sum_{k=0}^{\infty} \alpha^k g(x_k, \mu(x_k)) | x_0 = i] \quad \forall i \in \mathcal{S}.$$

We call the associated value function J^*, or the optimal value function. In other words,

$$J^* := \min_\mu J^\mu.$$

In order to find J^* and a corresponding optimal policy, we define the Bellman optimality operator T. When there is no ambiguity, we call T the Bellman operator. We define the Bellman operator $T : \mathbb{R}^{|\mathcal{S}|} \to \mathbb{R}^{|\mathcal{S}|}$ as follows:

$$TJ = \min_\mu E[g_\mu + \alpha P_\mu J].$$

Component-wise, we have the following:

$$TJ(i) = \min_u \Big[g(i,u) + \alpha \sum_{j=1}^{|\mathcal{S}|} P_{ij}(u) J(j) \Big].$$

For any vector J, we say that the policy corresponding to TJ is the *greedy* policy. When we apply the Bellman operator H times to vector J, we denote the resulting operator, T^H, as the H-step "lookahead" corresponding to J. We call the greedy policy corresponding to $T^H J$ the H-step lookahead policy, or the lookahead policy, when H is understood. See [Winnicki et al., 2021] for more definitions on the lookahead policy. More succinctly, the lookahead policy μ corresponding to vector V is the following:

$$\mu \in \arg\min_\mu T^\mu T^{H-1} V.$$

We have that every time the Bellman operator is applied to vector J to obtain TJ,

$$\|TJ - J^*\|_\infty \leq \alpha \|J - J^*\|_\infty.$$

Thus, applying T to obtain TJ gives a better estimate of the value function than J, and hence, better lookahead policies than greedy policies.

The Bellman equations state that J^* is a solution to

$$J^* = TJ^*.$$

It is well known that every greedy policy with respect to the optimal value function J^* is optimal and vice versa [Bertsekas and Tsitsiklis, 1996].

3 CONVERGENCE OF FIRST-VISIT TRAJECTORY-BASED POLICY ITERATION

The convergence of the Monte Carlo with Exploring Starts (Monte Carlo ES) algorithm (page 99 of [Sutton and Barto, 2018]) is unknown and is "one of the most fundamental open theoretical questions in reinforcement learning" (page 99 of [Sutton and Barto, 2018]).

The episodic algorithm iteratively alternates between policy improvement and evaluation using a single trajectory corresponding to the policy in each episode. For states visited by the trajectory, sums of costs beginning at those states are available and constitute estimates of the value function at the states visited by the trajectory. Then, the estimates of the value function at the states visited by the trajectory are used to update components of a vector which stores estimates of the optimal value function of all states for the states visited by the trajectory. Then the next greedy policy is determined from the updated estimate of the optimal value function and the iterative process continues.

Algorithm: We consider a version of the Monte Carlo ES algorithm similar to the main algorithm in [Tsitsiklis, 2002], which provides a "partial solution" of the open problem. At each iteration, k, the algorithm stores an estimate of the optimal value function, $V_k \in \mathbb{R}^{|\mathcal{S}|}$. Using V_k, just as in policy iteration, the algorithm obtains a trajectory corresponding to the lookahead policy (see Section 2) corresponding to V_k, μ_{k+1}, where

$$\mu_{k+1} = \arg\min_{\mu} T^{\mu} T^{H-1} V_k.$$

We call the set of states visited by the trajectory $\mathcal{D}_k$. Note that as stated in [Winnicki et al., 2021, Efroni et al., 2019], the lookahead policy only needs to be computed for states in $\mathcal{D}_k$. Additionally, while the computation of $T^{H-1}V_k(i)$ for $i \in \mathcal{D}_k$ may be infeasible, in practice, techniques such as Monte Carlo Tree Search (MCTS) are employed [Silver et al., 2017b], which are particularly useful when the number of next states and actions is small.

The trajectory is then used to obtain estimates of $J^{\mu_{k+1}}(i)$ for $i \in \mathcal{D}_k$, which we call $\hat{J}^{\mu_{k+1}}(i)$. In order to obtain $\hat{J}^{\mu_{k+1}}(i)$ for $i \in \mathcal{D}_k$, we perform an m-step rollout (see Section 2) with policy μ_{k+1} by obtaining a discounted sum of m costs beginning at each i for $i \in \mathcal{D}_k$. The return gives us a noisy version of $T^m_{\mu_{k+1}} T^{H-1} V_k(i)$ for $i \in \mathcal{D}_k$. We call $w_k(i)$ the unbiased noise that arises, noting that $0 \leq w_k(i) \leq \frac{1}{1-\alpha}$. If a state is encountered more than once by the trajectory, we consider the rollout from the first visit to the state. Using $T^m_{\mu_{k+1}} T^{H-1} V_k(i) + w_k(i)$ for $i \in \mathcal{D}_k$, we obtain the next iterate as follows:

$$V_{k+1}(i) = \begin{cases} (1-\gamma_k)V_k + \gamma_k(T^m_{\mu_{k+1}} T^{H-1} V_k + w_k) & i \in \mathcal{D}_k \\ V_k(i) & i \notin \mathcal{D}_k. \end{cases}$$

where γ_k and is assumed to be square summable and sums to infinity is the stepsize or learning rate.

We can write our iterates as follows:

$$V_{k+1}(i) = \mathcal{I}_{i \in \mathcal{D}_k} \Big[(1-\gamma_k)V_k(i) + \gamma_k(T^m_{\mu_{k+1}} T^{H-1} V_k(i) + w_k(i)) \Big] + \mathcal{I}_{i \notin \mathcal{D}_k} \Big[V_k(i) \Big],$$

where $\mathcal{I}_{i \in \mathcal{D}_k}$ denotes the indicator function which equals one when state i is visited by the trajectory at iteration k and zero otherwise.

With some algebra, can we rewrite V_{k+1} as follows:

$$V_{k+1} = (I - \gamma_k P_{k,\mu_k})V_k + \gamma_k P_{k,\mu_k}(T^m_{\mu_{k+1}} T^{H-1} V_k + z_k), \tag{1}$$

where I denotes the $\mathcal{S} \times \mathcal{S}$ identity matrix, $p_{k,\mu_k}(i)$ is the probability that state i is ever visited by the trajectory under policy μ_k, P_{k,μ_k} is the diagonal matrix where diagonal entries of the matrix correspond to the values of $p_{k,\mu_k}(i)$ for all $i \in \mathcal{S}$, and z_k satisfies the same properties as w_k.

Our algorithm is described in Algorithm 1.

Remark: Note that we need not compute $\mu_{k+1}(i)$ for all states $i \in \mathcal{S}$ at instance $k+1$. We only need to compute $\mu_{k+1}(i)$ for states encountered in the rollout step of the algorithm.

Note the similarity of our algorithm and the algorithm in [Tsitsiklis, 2002]:

$$V_{k+1} = (1-\gamma_k)V_k + \gamma_k(J^{\tilde{\mu}_{k+1}} + w_k),$$

where $\tilde{\mu}_{k+1}$ denotes the greedy policy with respect to V_k (i.e., $H = 1$).

The proof of the main algorithm in [Tsitsiklis, 2002] is similar to the main steps of the proof of modified policy iteration [Puterman and Shin, 1978] and hinges on showing that

$$\limsup_{k \to \infty} TV_k - V_k \leq 0.$$

Algorithm 1 First-Visit Monte Carlo Policy Evaluation For Policy Iteration

Input: V_0, m, H.

1: Let $k = 0$.
2: Let μ_{k+1} be such that $T_{\mu_{k+1}} T^{H-1} V_k = T^H V_k$.
3: Obtain a trajectory using policy μ_{k+1} and obtain $T^m_{\mu_{k+1}} T^{H-1} V_k(i) + w_k(i)$ for $i \in \mathcal{D}_k$, where $\mathcal{D}_k$ is the set of states visited by the trajectory and $w_k(i)$ for $i \in \mathcal{D}_k$ is unbiased noise from sampling.
4: Obtain V_{k+1} as follows

$$V_{k+1}(i) = \begin{cases} (1-\gamma_k)V_k(i) & \\ +\gamma_k(T^m_{\mu_{k+1}} T^{H-1} V_k(i) + w_k(i)) & i \in \mathcal{D}_k \\ V_k(i) & i \notin \mathcal{D}_k. \end{cases}$$

5: Set $k \leftarrow k+1$. Go to 2.

To show that $\limsup_{k\to\infty} TV_k - V_k \leq 0$, the proof relies on the following steps:

$$\begin{aligned} TV_{k+1} &\leq T_{\tilde{\mu}_{k+1}} V_{k+1} \\ &= T_{\tilde{\mu}_{k+1}}((1-\gamma_k)V_k + \gamma_k(J^{\tilde{\mu}_{k+1}} + w_k)) \\ &= g_{\tilde{\mu}_{k+1}} + \alpha P_{\tilde{\mu}_{k+1}}((1-\gamma_k)V_k + \gamma_k(J^{\tilde{\mu}_{k+1}} + w_k)) \\ &= (1-\gamma_k)(g_{\tilde{\mu}_{k+1}} + \alpha P_{\tilde{\mu}_{k+1}} V_k) \\ &+ \gamma_k(g_{\tilde{\mu}_{k+1}} + \alpha P_{\tilde{\mu}_{k+1}}(J^{\tilde{\mu}_{k+1}} + w_k)) \\ &= (1-\gamma_k)(TV_k) + \gamma_k(J^{\tilde{\mu}_{k+1}} + \alpha P_{\tilde{\mu}_{k+1}} w_k), \end{aligned}$$

where $P_{\tilde{\mu}_{k+1}}$ is the probability transition matrix corresponding to the Markov chain induced by policy $\tilde{\mu}_{k+1}$. We can then subtract V_{k+1} from both sides and easily obtain the stochastic approximation paradigm that allows us to show that $\limsup_{k\to\infty} TV_k - V_k \leq 0$:

$$\begin{aligned} &\underbrace{TV_{k+1} - V_{k+1}}_{=X_{k+1}} \\ &\leq (1-\gamma_k)(\underbrace{TV_k - V_k}_{=X_k}) + \gamma_k(\underbrace{(\alpha P_{\tilde{\mu}_{k+1}} - I)w_k}_{=:v_k}), \end{aligned}$$

where I denotes the identity matrix and the noise v_k satisfies similar properties to w_k. The purpose of showing that $\limsup_{k\to\infty} TV_k - V_k \leq 0$ is that asymptotically, we can use monotonicity to show that $J^* \leq J^{\tilde{\mu}_{k+1}} \leq TV_k$. Since $J^{\tilde{\mu}_{k+1}}$ is upper and lower bounded by contractions with fixed point J^*, we can use stochastic approximation techniques to obtain convergence of our iterates.

In our algorithm given by its iterates in (1), it is clear that we cannot perform the steps of the above used in the proof of [Tsitsiklis, 2002].

We now give Theorem 1, which shows that with sufficiciently large lookahead, the iterates in equation (1) converge to the optimal value function.

Assumption 1 *(a) The starting state of the trajectory at each instance is drawn from a fixed distribution, p, where $p(i) > 0 \forall i \in \mathcal{S}$.*

(b) $\alpha^{H-1} + 2(1+\alpha^m)\frac{\alpha^{H-1}}{1-\alpha} < 1$.

(c) $\sum_{i=0}^{\infty} \gamma_i = \infty$. *Also,* $\sum_{i=0}^{\infty} \gamma_i^2 < \infty$.

We make several remarks on our assumptions:

(a) The first assusmption is what is denoted as "exploring starts" (see [Sutton and Barto, 2018]), and guarantees for all states to be selected infinitely many times. We note that it is straightforward to extend our results to any initial distribution as long as the probability of visiting any state is lower bounded by a constant. In particular, we do not require a fixed probability distribution for the initial state.

(b) We assume the lookahead is sufficiently large, see previous sections for more on lookahead.

(c) The stepsizes are square summable and sum to infinity, which allows for noise averaging.

Theorem 1 *Under Assumption 1, the iterates of Algorithm 1 given in equation* (1) *converge to J^*, the optimal value function, almost surely.*

The proof is given in the Appendix.

3.1 Proof Idea

The main idea in the proof is the following. With sufficiently large lookahead, we can show that

$$H(V_k) := T^m_{\mu_{k+1}} T^{H-1} V_k \tag{2}$$

is a contraction towards J^*, and hence we can apply stochastic approximation techniques to obtain convergence of $V_k \to J^*$. We note that in equation (1), we have written μ_{k+1} as a function of V_k since it is the lookahead policy with respect to V_k. The matrix P_{k,μ_k} is a diagonal matrix where each diagonal element indicates if the corresponding state is visited by the trajectory. If the matrix were a constant, one can use the techniques of [Tsitsiklis, 2002], but the key challenge for us is that the matrix is history dependent. The key to our proofs lies in the fact that, with sufficient lookahead, the operator $H(V)$ defined in (2) is a contraction. For clarify, we can alternatively rewrite T_μ as $T_{\mu(V)}$ when μ is a lookahead policy corresponding to vector V. Note that while the operator T is a contraction, when we consider the operator $T_{\mu(V)}$, μ depends on V because μ is the lookahead policy with respect to V. Therefore, it is not obvious if $||T^m_{\mu(V_1)} T^{H-1} V_1 - T^m_{\mu(V_2)} T^{H-1} V_2||_\infty$ is smaller than $||V_1 - V_2||_\infty$.

Our proof hinges on the following key Lemma:

Lemma 1

$$\|J^{\mu_{k+1}} - T^{H-1}V_k\|_\infty \leq \frac{\alpha^{H-1}}{1-\alpha}\|TV_k - V_k\|_\infty,$$

where $J^{\mu_{k+1}}$ is the value function corresponding to policy μ_{k+1} (see Section 2). We will prove Lemma 1 in the Appendix. Using Lemma 1, we can show that:

$$\|T^m_{\mu_{k+1}}T^{H-1}V_k - T^{H-1}V_k\|_\infty \leq (\frac{\alpha^{m+H-1}}{1-\alpha} + \frac{\alpha^{H-1}}{1-\alpha})\|TV_k - V_k\|_\infty.$$

We can now subtract J^* from both sides of the inequality and use the contraction property of the Bellman operator to get:

$$\|T^m_{\mu_{k+1}}T^{H-1}V_k - J^*\|_\infty \leq \Big(\alpha^{H-1} + 2(1+\alpha^m)\frac{\alpha^{H-1}}{1-\alpha}\Big)\|V_k - J^*\|_\infty.$$

3.2 Novelty of the Proof Technique

Contrasting with the proof technique of [Tsitsiklis, 2002], we can see that due to the contraction property that follows from the use of lookahead, we can evade the issues that arise when the proof of [Tsitsiklis, 2002] is extended to include trajectory based updates. Additionally, the contraction based property allows us reduce the asymptotic error that arises from feature vector representation using lookahead, which is the topic of the next section.

Remarks: In the special case where the Markov chains induced by all policies are irreducible and infinitely long trajectories are obtained, we recover the results of the main algorithm in [Tsitsiklis, 2002].

Furthermore, suppose we obtain $T^m_{\mu_{k+1}}T^{H-1}V_k(i) + w_k(i)$ for all $i \in \mathcal{S}$, for *any* m and H. Then, we can write the following iterative sequence:

$$V_{k+1} = (1-\gamma_k)T^{H-1}V_k + \gamma_k(T^m_{\mu_{k+1}}T^{H-1}V_k + w_k). \tag{3}$$

We will show in the Appendix that the iterates in (3) converge to J^* a.s. Hence, we obtain convergence of a generalized version of the main algorithm in [Tsitsiklis, 2002].

4 EXTENSIONS OF FIRST-VISIT SIMULATION-BASED POLICY ITERATION TO LINEAR FUNCTION APPROXIMATION

When the sizes of the state and action spaces are very large, we can assign a feature vector $\phi(s) \in \mathbb{R}^d$ to each state s of the state space $\mathcal{S}$, where $d << |\mathcal{S}|$, and at iteration k obtain an estimate of the value function corresponding to μ_{k+1}, $\phi(s)^\top\theta^{\mu_{k+1}}$, where $\theta^{\mu_{k+1}} \in \mathbb{R}^d$ and $\theta^{\mu_{k+1}}$ is estimated from the trajectory corresponding to μ_{k+1}. We define Φ to be a matrix whose rows are the feature vectors.

In this way, instead of storing vectors $V_k \in \mathbb{R}^{|\mathcal{S}|}$, we can instead update vectors $\theta_k \in \mathbb{R}^d$, where $d << |\mathcal{S}|$.

When a single trajectory corresponding to the lookahead policy is available, there are many ways to estimate $\theta^{\mu_{k+1}}$. We will formulate an algorithm that allows us to analyze general methods of obtaining $\theta^{\mu_{k+1}}$ and provide convergence guarantees and finite-time bounds for the algorithm.

Our main assumption on the method used to estimate $\theta^{\mu_{k+1}}$ is that there exists known κ and δ_{app} such that

$$\|E[\Phi\theta^{\mu_{k+1}}|\mathcal{F}_k] - T^{H-1}\Phi\theta_k\|_\infty \leq \kappa\|T^{H-1}\Phi\theta_k - J^{\mu_{k+1}}\|_\infty + \delta_{app}, \tag{4}$$

where $\delta_{app} > 0$ and $0 < \alpha^{H-1} + \kappa\frac{2\alpha^{H-1}}{1-\alpha} < 1$. We will later show how κ and δ_{app} can be obtained for different policy evaluation algorithms.

We present Algorithm 2 and convergence guarantees of the algorithm for general κ and δ_{app} where $\mathcal{F}_k$ denotes the filtration associated with the noise of the algorithm until instance k. Note that similarly to Algorithm 1, we only need to compute Step 2 of Algorithm 2 (computation of μ_{k+1}) for states visited by the trajectory.

Algorithm 2 Function Approximation Algorithm With Trajectory Based Samples and Lookahead

Input: θ_0, m, H feature vectors $\{\phi(i)\}_{i\in\mathcal{S}}, \phi(i) \in \mathbb{R}^d$.

1: Let $k = 0$.
2: Let μ_{k+1} be such that $T_{\mu_{k+1}}T^{H-1}\Phi\theta_k = T^H\Phi\theta_k$.
3: Obtain a trajectory using policy μ_{k+1} and obtain $\theta^{\mu_{k+1}}$ where

$$\|E[\Phi\theta^{\mu_{k+1}}|\mathcal{F}_k] - T^{H-1}\Phi\theta_k\|_\infty \leq \kappa\|T^{H-1}\Phi\theta_k - J^{\mu_{k+1}}\|_\infty + \delta_{app}$$

for some $\delta_{app} > 0$ and κ such that:

$$0 < \alpha^{H-1} + \kappa\frac{2\alpha^{H-1}}{1-\alpha} < 1.3$$

4:

$$\theta_{k+1} = (1-\gamma_k)\theta_k + \gamma_k(\theta^{\mu_{k+1}}). \tag{5}$$

5: Set $k \leftarrow k+1$. Go to 2.

Theorem 2 *Suppose the following conditions hold:*

- $\sum_{i=0}^\infty \gamma_i = \infty$, $\sum_{i=0}^\infty \gamma_i^2 < \infty$

- *there exist* $\delta_{app} > 0$, *and* $\kappa > 0$ *such that*

$$0 < \alpha^{H-1} + \kappa \frac{2\alpha^{H-1}}{1-\alpha} < 1,$$

and

$$\|E[\Phi\theta^{\mu_{k+1}}|\mathcal{F}_k] - T^{H-1}\Phi\theta_k\|_\infty \\ \leq \kappa\|T^{H-1}\Phi\theta_k - J^{\mu_{k+1}}\|_\infty + \delta_{app},$$

Then, almost surely, the following bound holds for iterates θ_k *of Algorithm 2:*

$$\limsup_{k\to\infty}\|\Phi\theta_k - J^*\|_\infty \leq \frac{\delta_{app}}{1-\alpha^{H-1}-\kappa\frac{2\alpha^{H-1}}{1-\alpha}}$$

almost surely and that the policies obtained almost surely have the following property:

$$\limsup_{k\to\infty}\|J^{\mu_k} - J^*\|_\infty \leq \frac{2\alpha^{H-1}}{1-\alpha}\Big[\frac{\delta_{app}}{1-\alpha^{H-1}-\kappa\frac{2\alpha^{H-1}}{1-\alpha}}\Big].$$

The proof of Theorem 2 can be found in the Appendix.

4.1 Proof Idea

Using Lemma 1, we will show that our $E[\Phi\theta^{\mu_{k+1}}|\mathcal{F}_k]$ is nearly a contraction with respect to J^*. To see this, notice that we can use Lemma 1 to further upper bound the inequality in (4) as follows:

$$\|E[\Phi\theta^{\mu_{k+1}}|\mathcal{F}_k] - T^{H-1}\Phi\theta_k\|_\infty \\ \leq \frac{\kappa\alpha^{H-1}}{1-\alpha}\|T\Phi\theta_k - \Phi\theta_k\|_\infty + \delta_{app}.$$

We can now subtract J^* from both sides of the inequality and use the contraction property of the Bellman operator to get:

$$\|E[\Phi\theta^{\mu_{k+1}}|\mathcal{F}_k] - J^*\|_\infty \\ \leq (\alpha^{H-1} + \frac{2\kappa\alpha^{H-1}}{1-\alpha})\|\Phi\theta_k - J^*\|_\infty + \delta_{app}.$$

Roughly speaking, the above inequality states that the result of the sampling, $\theta^{\mu_{k+1}}$, contracts towards J^* compared to the previous iterate θ_k. We can then use stochastic approximation techniques to obtain convergence of our iterates.

4.2 Finite-Time Bounds

Theorem 3 *Let* σ^2 *be an upper bound on the variance of* $(\Phi\theta^{\mu_{k+1}} - E[\Phi\theta^{\mu_{k+1}}|\mathcal{F}_k])$ *for all* k. *Then, we have the following finite-time error bound for Algorithm 2:*

$$E[\|\Phi\theta_k - J^*\|_\infty] \\ \leq \underbrace{\prod_{i=1}^{k-1} a_i\|\Phi\theta_0 - J^*\|_\infty}_{\text{initial condition error}} + \underbrace{\delta_{app}\sum_{j=1}^{k-1}\gamma_j\prod_{\ell=j+1}^{k-1} a_\ell}_{\text{error due to function approximation}} \\ + \underbrace{\sum_{j=1}^{k-1}\gamma_j(\sigma_{j+1}+\sigma_j)\prod_{\ell=j+1}^{k-1} a_\ell}_{\text{error due to noise}},$$

where $a_i := 1 - \gamma_i(1 - \alpha^{H-1} + \frac{2\kappa\alpha^{H-1}}{1-\alpha})$ *and* σ_j *is defined recursively as follows:*

$$\sigma_j = \sigma\sqrt{\sum_{i=1}^{j}\gamma_i^2\prod_{\ell=i+1}^{j}(1-\gamma_\ell)^2}. \tag{6}$$

Interpretation Of Finite-Time Bounds: We will now interpret the terms of the finite-time bounds:

- Initial condition error: This term goes to 0 as $k \to \infty$. To see this, notice that $0 < 1 - \alpha^{H-1} + \frac{2\kappa\alpha^{H-1}}{1-\alpha} < 1$ due to our assumptions in Theorem 2. Thus, since $\sum_{i=0}^{\infty}\gamma_i = \infty$ and $\sum_{i=0}^{\infty}\gamma_i^2 < \infty$, we have that $\prod_{i=1}^{k-1} a_i \to 0$.
- Error due to function approximation: δ_{app} can be interpreted as the error that arises from the use of feature vectors. In our later analysis of δ_{app} for various algorithms, it can be seen δ_{app} can be made small if the feature vectors are sufficiently expressive.
- Error due to noise: This error term is due to Monte Carlo sampling. Since the discounted infinite horizon reward is bounded by $1/(1-\alpha)$, almost sure convergence in the previous theorem implies that this term must also go to zero as $k \to \infty$. For more discussion, see the Appendix.

We now obtain κ and δ_{app} for several methods of computing $\theta^{\mu_{k+1}}$.

4.3 First Visit Monte-Carlo Policy Evaluation With Feature Vectors

We will now go back to Algorithm 1 and directly extend the results to include the use of feature vectors and θ_k instead of V_k.

Recall in the previous section that a single trajectory corresponding to the lookahead policy μ_{k+1} is obtained. We denote the states visited by the trajectory as $\mathcal{D}_k$. Just as in the previous section, for all states $i \in \mathcal{D}_k$, we obtain

$T^m_{\mu_{k+1}} T^{H-1}\Phi\theta_k(i) + w_k(i)$. Note that analogously to Section 3, we need only to compute the lookahead for states visited by the trajectory. Additionally, we do not need to compute $\Phi\theta_k$ - we only need to compute $\Phi\theta_k(i) = \phi(i)^\top\theta_k$ for states i visited by the trajectory or involved in the tree search.

Now, instead of updating $V_k(i)$ for $i \in \mathcal{D}_k$, the case in Section 3, we instead obtain $\theta^{\mu_{k+1}} \in \mathbb{R}^d$, which uses $T^m_{\mu_{k+1}} T^{H-1}\Phi\theta_k(i) + w_k(i)$ for $i \in \mathcal{D}_k$ to construct an estimate of $\theta^{\mu_{k+1}}$.

One way we obtain $\theta^{\mu_{k+1}}$ uses linear least squares to obtain the best fitting θ_{k+1}. We now compute $\theta^{\mu_{k+1}}$ with linear least squares using the term $\hat{J}^{\mu_{k+1}}$ which we will define in the next paragraph:

$$\begin{aligned}\theta^{\mu_{k+1}} &:= \arg\min_\theta \frac{1}{2}\|(\mathcal{P}_{1,k}\Phi)\theta - \mathcal{P}_{2,k}\hat{J}^{\mu_{k+1}}\|_2^2 \\ &= (\mathcal{P}_{1,k}\Phi)^+\mathcal{P}_{2,k}\hat{J}^{\mu_{k+1}}. \end{aligned} \tag{7}$$

We now explain the above terms:

- Φ is a matrix whose rows are the feature vectors
- $\mathcal{P}_{2,k}$ is a matrix whose elements are a subset of the elements of $\hat{J}^{\mu_{k+1}}$ corresponding to $\mathcal{D}_k$. Since the values of $\hat{J}^{\mu_{k+1}}(i)$ for $i \notin \mathcal{D}_k$ do not affect $\mathcal{P}_{2,k}\hat{J}^{\mu_{k+1}}$, we can define $\hat{J}^{\mu_{k+1}}$ as follows:

  $$\hat{J}^{\mu_{k+1}} := T^m_{\mu_{k+1}} T^{H-1}\Phi\theta_k + w_k,$$

  where $w_k := 0$ for states $i \notin \mathcal{D}_k$. Notice that since $E[w_k(i)|\mathcal{F}_k] = 0$ for $i \in \mathcal{D}_k$ we have that $E[w_k|\mathcal{F}_k] = 0$.
- $\mathcal{P}_{1,k}$ is a matrix of zeros and ones such that rows of $\mathcal{P}_{1,k}\Phi$ correspond to feature vectors associated with states in $\mathcal{D}_k$.
- $(\mathcal{P}_{1,k}\Phi)^+$ is the Moore-Penrose inverse of $\mathcal{P}_{1,k}\Phi$.

Using the previous paragraph, we rewrite our iterates in (7) as follows:

$$\theta^{\mu_{k+1}} = (\mathcal{P}_{1,k}\Phi)^+\mathcal{P}_{2,k}(T^m_{\mu_{k+1}} T^{H-1}\Phi\theta_k + w_k).$$

Our estimate of the value function at iteration k is thus given by:

$$\Phi\theta^{\mu_{k+1}} = \underbrace{\Phi(\mathcal{P}_{1,k}\Phi)^+\mathcal{P}_{2,k}}_{=:\mathcal{M}_k}(T^m_{\mu_{k+1}} T^{H-1}\Phi\theta_k + w_k),$$

where $\mathcal{M}_k$ is a projection matrix used to estimate the $J^{\mu_{k+1}}$ from samples of $\hat{J}^{\mu_{k+1}}(i) = T^m_{\mu_{k+1}} T^{H-1}\Phi\theta_k(i) + w_k(i)$ for $i \in \mathcal{D}_k$.

With $\theta^{\mu_{k+1}}$ defined as the above, we now obtain our corresponding δ_{app} and κ. See Appendix for proofs.

- $\kappa = 1 + \alpha^m\delta_{FV}$ where $\delta_{FV} := \sup_k\|\mathcal{M}_k\|_\infty$.
- δ_{app}

  $$= \sup_{k,\mu_k}\|E[\mathcal{M}_k(J^{\mu_{k+1}} + w_k) - (J^{\mu_{k+1}} + w_k)|\mathcal{F}_k]\|_\infty,$$

  where w_k is a martingale difference sequence, meaning that $E[w_k|\mathcal{F}_k] = 0$.

How to Interpret Terms In The Error?

- δ_{app} : Our error terms in the previous theorems mostly hinge on δ_{app}. Since $\mathcal{M}_k$ is a matrix which uses the feature vectors corresponding to states in $\mathcal{D}_k$ to construct an estimate of $\theta^{\mu_{k+1}}$ based on samples of $\hat{J}^{\mu_{k+1}}(i)$ for states $i \in \mathcal{D}_k$, it is easy to see that δ_{app} is a measure of the ability of the feature vectors to approximate the value functions corresponding to the lookahead policies. Hence, with increasingly expressive feature vectors, the error terms in Theorem 2 go to 0.
- κ : In the presence of sufficiently large lookahead, κ does not drastically alter the results in our theorems. However, we note that typically the quantity m denotes the length of the trajectory starting at each state in $\mathcal{D}_k$, so typically m is very large, and hence κ is close to 1.

Remarks: In order to compute $T^m_{\mu_{k+1}} T^{H-1}\Phi\theta_k(i) + w_k(i)$ for $i \in \mathcal{D}_k$, we do not need to compute $\Phi\theta_k$; we need only compute $\phi(i)^\top\theta_k$ for states $i \in \mathcal{D}_k$ and states i involved in the computation of the tree search at states visited by the trajectory. Recall from Section 3 that we only need to compute the lookahead and μ_{k+1} for states visited by the trajectory.

Suppose that $\mathcal{M}_k = I$, i.e. when we obtain an estimate of

$$T^m_{\mu_{k+1}} T^{H-1}V_k(i)$$

for all $i \in \mathcal{S}$. Then, $\Phi\theta_k \to J^*$ a.s. This matches the result of Theorem 1. Additionally, we can see that the error bounds mostly depend on the ability of the representative ability of the feature vectors instead of the sizes of the state and action spaces.

4.4 Extension To Gradient Descent

In order to speed up the rate of convergence of our iterates, we instead take several steps of gradient descent towards

$$\theta^{\mu_{k+1}} = \arg\min_\theta \frac{1}{2}\|(\mathcal{P}_{1,k}\Phi)\theta - \mathcal{P}_{2,k}\hat{J}^{\mu_{k+1}}\|_2^2,$$

where $\mathcal{P}_{1,k}$ and $\mathcal{P}_{2,k}$ are defined in the previous subsection. In other words, when η denotes the number of steps of gradient descent we take, for $\ell = 1, 2, \ldots, \eta$. We recursively

compute the following:

$$\theta_{k+1,\ell} = \theta_{k+1,\ell-1} - \xi \nabla_\theta c(\theta; \hat{J}^{\mu_{k+1}})|_{\theta_{k+1,\ell-1}}, \quad (8)$$

where $c(\theta; \hat{J}^{\mu_{k+1}}) := \frac{1}{2}\min_\theta \|(\mathcal{P}_{1,k}\Phi)\theta - \mathcal{P}_{2,k}\hat{J}^{\mu_{k+1}}\|_2^2$,

$$0 < \xi < \frac{1}{\sigma_{\mathcal{P}_{1,k}\Phi,\max}},$$

and $\sigma_{\mathcal{P}_{1,k}\Phi,\max}$ is the largest singular value squared of $\mathcal{P}_{1,k}\Phi$, and $\theta_{k+1,0} = 0$. We then set $\theta^{\mu_{k+1}} = \theta_{k+1,\eta}$. We obtain the following κ and δ_{app}:

- $\kappa = 1 + \alpha^m \delta_{FV}, \delta_{FV} := \sup_k \|(\mathcal{P}_{1,k}\Phi)^+ \mathcal{P}_{2,k}\|_\infty$.
- $\delta_{app} = \sup_{k,\mu_k} \|E[\mathcal{M}_k(J^{\mu_{k+1}} + w_k) - J^{\mu_{k+1}}|\mathcal{F}_k]\|_\infty$
$$+ (1 - \xi\sigma_{\mathcal{P}_{1,k}\Phi,\max})^\eta \|\Phi\|_\infty \|V_{k,1}\|_\infty \times \|\Sigma_{k,1}^{-1}\|_\infty \|U_k^\top \mathcal{P}_{2,k}\hat{J}^{\mu_{k+1}}\|_\infty,$$
where the singular value decomposition of $\mathcal{P}_{1,k}\Phi$ is:
$$\mathcal{P}_{1,k}\Phi = U_k \begin{bmatrix}\Sigma_{k,1} & 0\end{bmatrix} \begin{bmatrix} V_{k,1}^\top \\ V_{k,2}^\top \end{bmatrix} = U_k \Sigma_{k,1} V_{k,1}^\top.$$
where U_k is a unitary matrix, $\Sigma_{k,1}$ is a rectangular diagonal matrix, and V_k is a unitary matrix.

Our proof of the above is in the Appendix. Ultimately, the purpose of gradient descent is to improve the computational efficiency of the least squares algorithm in Subsection 4.3. The results show that as $\eta \to \infty$, the δ_{app} of the gradient descent algorithm equals to the δ_{app} from Subsection 4.3. The rate of convergence of δ_{app} of the gradient descent algorithm towards δ_{app} of Subsection 4.3 as a function of the number of steps of gradient descent is exponential.

4.5 Other Algorithms Including TD-Learning

Now consider a general mechanism of obtaining $\theta^{\mu_{k+1}}$ from a sample trajectory. We make the assumption that $\theta^{\mu_{k+1}}$ is bounded (which is always the case for methods with a fixed number of iterations for computing $\theta^{\mu_{k+1}}$). When we have a δ such that for all μ_{k+1} :

$$\|E[\Phi\theta^{\mu_{k+1}}] - J^{\mu_{k+1}}\|_\infty \le \delta,$$

i.e., there exists some δ which is an upper bound of the error of the method of estimating $J^{\mu_{k+1}}$, we can obtain a corresponding δ_{app} and κ as follows:

$$\begin{aligned}\|E[\Phi\theta^{\mu_{k+1}}] - T^{H-1}V_k\|_\infty &\le \|T^{H-1}V_k - J^{\mu_{k+1}}\|_\infty \\ &\quad + \|E[\Phi\theta^{\mu_{k+1}}] - J^{\mu_{k+1}}\|_\infty \\ &\le \|T^{H-1}V_k - J^{\mu_{k+1}}\|_\infty + \delta.\end{aligned}$$

Thus, when the mean square error is known, $\kappa = 1$ and $\delta_{app} = \delta$.

Recent studies including [Srikant and Ying, 2019, Bhandari et al., 2018] have obtained finite-time bounds for TD-learning with linear function approximation. The finite-time bounds in Theorem 3 of [Bhandari et al., 2018] are of the following form: for any μ_{k+1} where the output of the TD-learning algorithm is $\theta^{\mu_{k+1}}$, we have

$$E[\|\Phi\theta^{\mu_{k+1}*} - \Phi\theta^{\mu_{k+1}}\|_D^2] \le \delta_{T,\mu_{k+1}},$$

where $\delta_{T,\mu_{k+1}}$ depends on the number of iterations T of the TD-learning algorithm, $\|\cdot\|_D$ denotes the weighted 2-norm with weights corresponding to the stationary distribution of μ_{k+1}, and

$$\|\Phi\theta^{\mu_{k+1}*} - J^{\mu_{k+1}}\|_D \le \frac{1}{\sqrt{1-\alpha^2}} \min_\theta \|\Phi\theta - J^{\mu_{k+1}}\|_D,$$

meaning that $\theta^{\mu_{k+1}*}$ approximates $J^{\mu_{k+1}}$. From this, it can be shown that:

$$\begin{aligned}\|E[\Phi\theta^{\mu_{k+1}}] - J^{\mu_{k+1}}\|_\infty &\le \sup_{\mu_{k+1}} \|\Phi\theta^{\mu_{k+1}*} - J^{\mu_{k+1}}\|_\infty \\ &\quad + \frac{\sqrt{\delta_{T,\mu_{k+1}}}}{\pi_{\mu_{k+1},\min}},\end{aligned}$$

where $\pi_{\mu_{k+1},\min}$ denotes the minimum weight of the stationary distribution of μ_{k+1}, thus giving us a δ as desired. See Appendix for proofs with TD-learning.

5 CONCLUSION

We study Monte Carlo methods that estimate the value function corresponding to policies determined in the policy improvement step of Monte Carlo based policy iteration methods. We are concerned with trajectory based updates that involve obtaining estimates of the value function corresponding to the greedy policies from states that are visited by the trajectory. This is noted as an open problem in [Sutton and Barto, 2018] and [Tsitsiklis, 2002]. We show that when lookahead policies, which are commonly used in practice, are employed, we obtain convergence to the optimal value function. We further our analysis to include the use of feature vectors and also include analyses of general methods of policy evaluation in feature vector space that are computationally efficient such as TD learning.

Acknowledgements

The authors thank the reviewers for their helpful comments. The research presented here was supported by the following: NSF CCF 22-07547, NSF CNS 21-06801, NSF CCF 1934986, ONR N00014-19-1-2566, and ARO W911NF-19-1-0379.

References

[Baxter et al., 1999] Baxter, J., Tridgell, A., and Weaver, L. (1999). Tdleaf(lambda): Combining temporal

difference learning with game-tree search. *CoRR*, cs.LG/9901001.

[Bertsekas, 2011] Bertsekas, D. (2011). Approximate policy iteration: a survey and some new methods. *Journal of Control Theory and Applications*, 9:310–335.

[Bertsekas, 2022] Bertsekas, D. (2022). *Lessons from AlphaZero for Optimal, Model Predictive, and Adaptive Control*. Athena Scientific.

[Bertsekas and Tsitsiklis, 1996] Bertsekas, D. and Tsitsiklis, J. (1996). *Neuro-dynamic Programming*. Athena Scientific.

[Bertsekas, 2019] Bertsekas, D. P. (2019). *Reinforcement learning and optimal control*. Athena Scientific Belmont, MA.

[Bhandari et al., 2018] Bhandari, J., Russo, D., and Singal, R. (2018). A finite time analysis of temporal difference learning with linear function approximation. In *Conference on learning theory*, pages 1691–1692. PMLR.

[Browne et al., 2012] Browne, C., Powley, E., Whitehouse, D., Lucas, S., Cowling, P., Rohlfshagen, P., Tavener, S., Perez Liebana, D., Samothrakis, S., and Colton, S. (2012). A survey of monte carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in Games*, 4:1:1–43.

[Chen, 2018] Chen, Y. (2018). On the convergence of optimistic policy iteration for stochastic shortest path problem. *arXiv preprint arXiv:1808.08763*.

[Deng et al., 2020] Deng, H., Yin, S., Deng, X., and Li, S. (2020). Value-based algorithms optimization with discounted multiple-step learning method in deep reinforcement learning. In *2020 IEEE 22nd International Conference on High Performance Computing and Communications; IEEE 18th International Conference on Smart City; IEEE 6th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, pages 979–984.

[Efroni et al., 2018a] Efroni, Y., Dalal, G., Scherrer, B., and Mannor, S. (2018a). Beyond the one step greedy approach in reinforcement learning. *CoRR*, abs/1802.03654.

[Efroni et al., 2018b] Efroni, Y., Dalal, G., Scherrer, B., and Mannor, S. (2018b). Multiple-step greedy policies in online and approximate reinforcement learning. *arXiv preprint arXiv:1805.07956*.

[Efroni et al., 2019] Efroni, Y., Dalal, G., Scherrer, B., and Mannor, S. (2019). How to combine tree-search methods in reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3494–3501.

[Efroni et al., 2020] Efroni, Y., Ghavamzadeh, M., and Mannor, S. (2020). Online planning with lookahead policies. *Advances in Neural Information Processing Systems*, 33.

[Kocsis and Szepesvári, 2006] Kocsis, L. and Szepesvári, C. (2006). Bandit based monte-carlo planning. In *Machine Learning: ECML*, volume 2006, pages 282–293.

[Lagoudakis and Parr, 2003] Lagoudakis, M. G. and Parr, R. (2003). Least-squares policy iteration. *The Journal of Machine Learning Research*, 4:1107–1149.

[Lanctot et al., 2014] Lanctot, M., Winands, M. H., Pepels, T., and Sturtevant, N. R. (2014). Monte carlo tree search with heuristic evaluations using implicit minimax backups. In *2014 IEEE Conference on Computational Intelligence and Games*, pages 1–8. IEEE.

[Lesner and Scherrer, 2015] Lesner, B. and Scherrer, B. (2015). Non-stationary approximate modified policy iteration. In *ICML*.

[Liu, 2020] Liu, J. (2020). On the convergence of reinforcement learning with monte carlo exploring starts.

[Lubars et al., 2021] Lubars, J., Winnicki, A., Livesay, M., and Srikant, R. (2021). Optimistic policy iteration for MDPs with acyclic transient state structure. *CoRR*, abs/2102.00030.

[Ma et al., 2019] Ma, X., Driggs-Campbell, K., Zhang, Z., and Kochenderfer, M. J. (2019). Monte-carlo tree search for policy optimization. *arXiv preprint arXiv:1912.10648*.

[Mnih et al., 2016] Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T. P., Harley, T., Silver, D., and Kavukcuoglu, K. (2016). Asynchronous methods for deep reinforcement learning. *CoRR*, abs/1602.01783.

[Munos, 2014] Munos, R. (2014). From bandits to monte-carlo tree search: The optimistic principle applied to optimization and planning. *Foundations and Trends in Machine Learning*, 7.

[Powell, 2021] Powell, W. B. (2021). From reinforcement learning to optimal control: A unified framework for sequential decisions. In *Handbook of Reinforcement Learning and Control*, pages 29–74. Springer.

[Puterman and Shin, 1978] Puterman, M. and Shin, M. C. (1978). Modified policy iteration algorithms for discounted markov decision problems. *Management Science*, 24:1127–1137.

[Shah et al., 2020] Shah, D., Xie, Q., and Xu, Z. (2020). Non-asymptotic analysis of monte carlo tree search. In *Abstracts of the 2020 SIGMETRICS/Performance Joint International Conference on Measurement and Modeling of Computer Systems*, pages 31–32.

[Silver et al., 2016] Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al. (2016). Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489.

[Silver et al., 2017a] Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Kumaran, D., Graepel, T., Lillicrap, T. P., Simonyan, K., and Hassabis, D. (2017a). Mastering chess and shogi by self-play with a general reinforcement learning algorithm. *CoRR*, abs/1712.01815.

[Silver et al., 2017b] Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, A., et al. (2017b). Mastering the game of go without human knowledge. *Nature*, 550(7676):354–359.

[Singh and Sutton, 1996] Singh, S. P. and Sutton, R. S. (1996). Reinforcement learning with replacing eligibility traces. *Machine learning*, 22(1):123–158.

[Srikant and Ying, 2019] Srikant, R. and Ying, L. (2019). Finite-time error bounds for linear stochastic approximation andtd learning. In *Conference on Learning Theory*, pages 2803–2830. PMLR.

[Sutton and Barto, 1998] Sutton, R. S. and Barto, A. G. (1998). *Introduction to Reinforcement Learning*. MIT Press, Cambridge, MA, USA, 1st edition.

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

[Tomar et al., 2020] Tomar, M., Efroni, Y., and Ghavamzadeh, M. (2020). Multi-step greedy reinforcement learning algorithms. In *International Conference on Machine Learning*, pages 9504–9513. PMLR.

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

[Tsitsiklis, 2002] Tsitsiklis, J. N. (2002). On the convergence of optimistic policy iteration. *Journal of Machine Learning Research*, 3(Jul):59–72.

[Veness et al., 2009] Veness, J., Silver, D., Blair, A., and Uther, W. (2009). Bootstrapping from game tree search. In Bengio, Y., Schuurmans, D., Lafferty, J., Williams, C., and Culotta, A., editors, *Advances in Neural Information Processing Systems*, volume 22. Curran Associates, Inc.

[Wang et al., 2022] Wang, C., Yuan, S., Shao, K., and Ross, K. W. (2022). On the convergence of the monte carlo exploring starts algorithm for reinforcement learning. In *International Conference on Learning Representations*.

[Winnicki et al., 2021] Winnicki, A., Lubars, J., Livesay, M., and Srikant, R. (2021). The role of lookahead and approximate policy evaluation in policy iteration with linear value function approximation. *CoRR*, abs/2109.13419.

[Winnicki and Srikant, 2022] Winnicki, A. and Srikant, R. (2022). Reinforcement learning with unbiased policy evaluation and linear function approximation. In *2022 IEEE 61st Conference on Decision and Control (CDC)*, pages 801–806.

A PROOF OF CONVERGENCE OF ITERATES IN EQUATION (2) OF SECTION 3

We write our iterates as follows:

$$V_{k+1} = (1-\gamma_k)T^{H-1}V_k + \gamma_k(T^m_{\mu_{k+1}}T^{H-1}V_k + w_k).$$

We break the proof up into steps as follows.

Step 1:

$$\limsup_{k\to\infty} TV_k - V_k \leq 0.$$

Proof of Step 1: We will show that for every $\varepsilon > 0$, there exists sufficiently large $k(\varepsilon)$ such that the following holds:

$$(1-\gamma_k)T^{H-1}V_k + \gamma_k T^m_{\mu_{k+1}}T^{H-1}V_k - \varepsilon e \leq V_{k+1} \leq (1-\gamma_k)T^{H-1}V_k + \gamma_k T^m_{\mu_{k+1}}T^{H-1}V_k + \varepsilon e, \tag{9}$$

where e is the vector of all 1s.

To do this, we define a sequence of random variables, Y_k as follows:

$$Y_{k+1} = (1-\gamma_k)Y_k + \gamma_k w_k, Y_0 = 0.$$

It is clear that $Y_k \to 0$ a.s. by standard stochastic approximation theory. Then, we subtract Y_{k+1} from both sides of the iterates as follows:

$$V_{k+1} - Y_{k+1} = (1-\gamma_k)(T^{H-1}V_k - Y_k) + \gamma_k(T^m_{\mu_{k+1}}T^{H-1}V_k).$$

Rearranging terms, we have:

$$V_{k+1} = (1-\gamma_k)(T^{H-1}V_k) + \gamma_k(T^m_{\mu_{k+1}}T^{H-1}V_k) + Y_{k+1} - (1-\gamma_k)Y_k.$$

Since $Y_k \to 0$ a.s., we have that for every $\varepsilon > 0$, there exists $k(\varepsilon)$ such that for all $k > k(\varepsilon)$ we have the right side of the inequality in (9). The left side follows accordingly.

Using the inequality in (9), we have that:

$$\begin{aligned} TV_{k+1} &\leq T_{\mu_{k+1}}\Big[(1-\gamma_k)T^{H-1}V_k + \gamma_k T^m_{\mu_{k+1}}T^{H-1}V_k + \varepsilon e\Big] \\ &= (1-\gamma_k)T^H V_k + \gamma_k T^{m+1}_{\mu_{k+1}}T^{H-1}V_k + \alpha\varepsilon e. \end{aligned}$$

Furthermore, using the inequality in (9),

$$TV_{k+1} - V_{k+1} \leq (1-\gamma_k)(T^H V_k - T^{H-1}V_k) + \gamma_k(T^{m+1}_{\mu_{k+1}}T^{H-1}V_k - T^m_{\mu_{k+1}}T^{H-1}V_k) + (1+\alpha)\varepsilon e.$$

We recursively define δ_k such that:

$$TV_k - V_k \leq \delta_k e.$$

For $k(\varepsilon)$, we have that:

$$\delta_{k(\varepsilon)} := \|TV_k - V_k\|_\infty.$$

For $k > k(\varepsilon)$, we define δ_k as follows:

$$\delta_k = \delta_{k-1}(\alpha^{H-1} + \alpha^{m+H-1}) + (1+\alpha)\varepsilon.$$

It is clear that $TV_k - V_k \leq \delta_k$ since

$$\begin{aligned}
&TV_{k-1} - V_{k-1} \leq \delta_{k-1} e \\
&\implies T^H V_{k-1} - T^{H-1} V_{k-1} \leq \alpha^{H-1} \delta_{k-1} e \\
&\implies T^m_{\mu_k} T^H V_{k-1} - T^m_{\mu_k} T^{H-1} V_{k-1} \leq \alpha^{m+H-1} \delta_{k-1} e.
\end{aligned}$$

Thus,

$$\begin{aligned}
TV_{k+1} - V_{k+1} &\leq (1-\gamma_k)\alpha^{H-1}\delta_{k-1} e + \gamma_k[\alpha^{m+H-1}\delta_{k-1} e] + (1+\alpha)\varepsilon e \\
&\leq (1-\gamma_k)\delta_{k-1}(\alpha^{H-1} e + \alpha^{m+H-1}) + (1+\alpha)\varepsilon e \\
&= \delta_k e.
\end{aligned}$$

Thus, we have that

$$\limsup_{k\to\infty} TV_k - V_k \leq \lim_{k\to\infty} \delta_k e.$$

We now calculate $\lim_{k\to\infty} \delta_k$ as follows:

$$\lim_{k\to\infty} \delta_k = \frac{1+\alpha}{\alpha^{H-1} + \alpha^{m+H-1}} \varepsilon.$$

Since ε can be any value greater than 0, we have that $\lim_{k\to\infty} TV_k - V_k \leq 0$.

Step 2:

For all $\varepsilon, \tilde{\varepsilon} > 0$,

$$V_{k+1} \leq T^{H-1} V_k + \frac{\alpha^{H-1}}{1-\alpha} \tilde{\varepsilon} e + \varepsilon e. \tag{10}$$

Proof of Step 2:

Hence, for any $\tilde{\varepsilon} > 0$, there exists $k(\tilde{\varepsilon})$ such that for any $k > k(\tilde{\varepsilon})$, $TV_k - V_k \leq \tilde{\varepsilon} e$.

Thus:

$$\begin{aligned}
&TV_k - V_k \leq \tilde{\varepsilon} e \\
&\implies TV_k \leq V_k + \tilde{\varepsilon} e \\
&\implies T^H V_k \leq T^{H-1} V_k + \alpha^{H-1} \tilde{\varepsilon} e \\
&\implies T^m_{\mu_{k+1}} T^{H-1} V_k \leq T^{H-1} V_k + \frac{\alpha^{H-1}}{1-\alpha} \tilde{\varepsilon} e.
\end{aligned}$$

Thus, we have for $k > k(\varepsilon) + k(\tilde{\varepsilon})$:

$$\begin{aligned}
V_{k+1} &\leq (1-\gamma_k) T^{H-1} V_k + \gamma_k (T^{H-1} V_k + \frac{\alpha^{H-1}}{1-\alpha} \tilde{\varepsilon} e) + \varepsilon e \\
&\leq T^{H-1} V_k + \gamma_k \frac{\alpha^{H-1}}{1-\alpha} \tilde{\varepsilon} e + \varepsilon e \\
&\leq T^{H-1} V_k + \frac{\alpha^{H-1}}{1-\alpha} \tilde{\varepsilon} e + \varepsilon e.
\end{aligned}$$

Step 3:

For all $\varepsilon, \tilde{\varepsilon} > 0$,

$$V_{k+1} \geq T^{m+H-1}V_k - \frac{\alpha^{H-1}}{1-\alpha}\tilde{\varepsilon}e - \varepsilon e.$$

Proof of Step 3: Furthermore, since $TV_k \leq V_k + \tilde{\varepsilon}$ for all $k > k(\tilde{\varepsilon})$, we have that

$$T^{H-1}V_k \geq T^{m+H-1}V_k - \frac{\alpha^{H-1}}{1-\alpha}\tilde{\varepsilon}e.$$

Thus:

$$\begin{aligned}
V_{k+1} &\geq (1-\gamma_k)(T^{m+H-1}V_k - \frac{\alpha^{H-1}}{1-\alpha}\tilde{\varepsilon}e) + \gamma_k(T^m_{\mu_{k+1}}T^{H-1}V_k) - \varepsilon e \\
&\geq (1-\gamma_k)(T^{m+H-1}V_k - \frac{\alpha^{H-1}}{1-\alpha}\tilde{\varepsilon}e) + \gamma_k(T^{m+H-1}V_k) - \varepsilon e \\
&= T^{m+H-1}V_k - (1-\gamma_k)\frac{\alpha^{H-1}}{1-\alpha}\tilde{\varepsilon}e - \varepsilon e \\
&\geq T^{m+H-1}V_k - \frac{\alpha^{H-1}}{1-\alpha}\tilde{\varepsilon}e - \varepsilon e.
\end{aligned}$$

Step 4:

$$\|V_{k+1} - J^*\|_\infty \leq \alpha^{H-1}\|V_k - J^*\|_\infty + \frac{\alpha^{H-1}}{1-\alpha}\tilde{\varepsilon} + \varepsilon.$$

Proof of Step 4:

Putting the above together, we have:

$$T^{m+H-1}V_k - \frac{\alpha^{H-1}}{1-\alpha}\tilde{\varepsilon}e - \varepsilon e \leq V_{k+1} \leq T^{H-1}V_k + \frac{\alpha^{H-1}}{1-\alpha}\tilde{\varepsilon}e + \varepsilon e.$$

Subtracting J^* and using the contraction property of the Bellman operator, we have:

$$\begin{aligned}
-\alpha^{m+H-1}\|V_k - J^*\|_\infty e - \frac{\alpha^{H-1}}{1-\alpha}\tilde{\varepsilon}e - \varepsilon e &\leq T^{m+H-1}V_k - J^* - \frac{\alpha^{H-1}}{1-\alpha}\tilde{\varepsilon}e - \varepsilon e \\
&\leq V_{k+1} - J^* \\
&\leq T^{H-1}V_k - J^* + \frac{\alpha^{H-1}}{1-\alpha}\tilde{\varepsilon}e + \varepsilon e \\
&\leq \alpha^{H-1}\|V_k - J^*\|_\infty e + \frac{\alpha^{H-1}}{1-\alpha}\tilde{\varepsilon}e + \varepsilon e.
\end{aligned}$$

Thus,

$$\|V_{k+1} - J^*\|_\infty \leq \alpha^{H-1}\|V_k - J^*\|_\infty + \frac{\alpha^{H-1}}{1-\alpha}\tilde{\varepsilon} + \varepsilon.$$

The above implies that:

$$\limsup_{k\to\infty}\|V_k - J^*\|_\infty \leq \frac{\frac{\alpha^{H-1}}{1-\alpha}\tilde{\varepsilon} + \varepsilon}{\alpha^{m+H-1} + \alpha^{H-1}}$$

Since the above holds for all $\varepsilon > 0$ and all $\tilde{\varepsilon} > 0$, we have that:

$$V_k \to J^* a.s.$$

B PROOF OF LEMMA 1

The following holds:

$$\begin{aligned}
&T_{\mu_{k+1}} T^{H-1} V_k - T^H V_k = 0 \\
&\implies T_{\mu_{k+1}} T^{H-1} V_k - T^H V_k + T^{H-1} V_k - T^{H-1} V_k = 0 \\
&\implies T_{\mu_{k+1}} T^{H-1} V_k + \|T^H V_k - T^{H-1} V_k\|_\infty e - T^{H-1} V_k \geq 0 \\
&\implies T_{\mu_{k+1}} T^{H-1} V_k + \alpha^{H-1} \|TV_k - V_k\|_\infty e - T^{H-1} V_k \geq 0 \\
&\implies J^{\mu_{k+1}} - T^{H-1} V_k \geq -\frac{\alpha^{H-1}}{1-\alpha} \|TV_k - V_k\|_\infty e,
\end{aligned}$$

where the last line follows from iteratively applying $T_{\mu_{k+1}}$ to both sides and using a telescoping sum and e is the vector of all 1s.

We also have:

$$\begin{aligned}
&T_{\mu_{k+1}} T^{H-1} V_k = T^H V_k \\
&\implies T_{\mu_{k+1}} T^{H-1} V_k - T^H V_k + T^{H-1} V_k - T^{H-1} V_k = 0 \\
&\implies T_{\mu_{k+1}} T^{H-1} V_k - \|T^H V_k - T^{H-1} V_k\|_\infty e - T^{H-1} V_k \leq 0 \\
&\implies T_{\mu_{k+1}} T^{H-1} V_k - \alpha^{H-1} \|TV_k - V_k\|_\infty e - T^{H-1} V_k \leq 0 \\
&\implies J^{\mu_{k+1}} - T^{H-1} V_k \leq \frac{\alpha^{H-1}}{1-\alpha} \|TV_k - V_k\|_\infty e.
\end{aligned}$$

Putting the above two together, we get the following:

$$\|J^{\mu_{k+1}} - T^{H-1} V_k\|_\infty \leq \frac{\alpha^{H-1}}{1-\alpha} \|TV_k - V_k\|_\infty.$$

C PROOF OF THEOREM 1

We break the proof of Theorem 1 up into steps.

Step 1:

$$\|J^{\mu_{k+1}} - T^{H-1} V_k\|_\infty \leq \frac{\alpha^{H-1}}{1-\alpha} \|TV_k - V_k\|_\infty.$$

Proof of Step 1: Step 1 is a restatement of Lemma 1 which is proved in Appendix B.

Step 2:

$$\|H(V_k) - T^{H-1} V_k\|_\infty \leq (\frac{\alpha^{m+H-1}}{1-\alpha} + \frac{\alpha^{H-1}}{1-\alpha}) \|TV_k - V_k\|_\infty$$

Proof of Step 2: We have:

$$\begin{aligned}
&\|H(V_k) - T^{H-1} V_k\|_\infty - \|T^{H-1} V_k - J^{\mu_{k+1}}\|_\infty \\
&\leq \|H(V_k) - T^{H-1} V_k + T^{H-1} V_k - J^{\mu_{k+1}}\|_\infty \\
&= \|H(V_k) - J^{\mu_{k+1}}\|_\infty \\
&\leq \alpha^m \|T^{H-1} V_k - J^{\mu_{k+1}}\|_\infty,
\end{aligned}$$

Which implies that

$$\|H(V_k) - T^{H-1} V_k\|_\infty \leq (1 + \alpha^m) \|T^{H-1} V_k - J^{\mu_{k+1}}\|_\infty.$$

Plugging in the results of Step 2, we have:

$$\|H(V_k) - T^{H-1}V_k\|_\infty \leq (1+\alpha^m)\frac{\alpha^{H-1}}{1-\alpha}\|TV_k - V_k\|_\infty.$$

Step 3:

$$\|H(V_k) - J^*\|_\infty \leq \underbrace{\left(\alpha^{H-1} + (1+\alpha^m)\frac{\alpha^{H-1}}{1-\alpha}(1+\alpha)\right)}_{=:\beta}\|V_k - J^*\|_\infty.$$

Proof of Step 3: We have

$$\begin{aligned}
&T^{H-1}V_k - (1+\alpha^m)\frac{\alpha^{H-1}}{1-\alpha}\|TV_k - V_k\|_\infty e \leq H(V_k) \\
&\leq T^{H-1}V_k + (1+\alpha^m)\frac{\alpha^{H-1}}{1-\alpha}\|TV_k - V_k\|_\infty e \\
&\implies \|H(V_k) - T^{H-1}V_k\|_\infty \leq (1+\alpha^m)\frac{\alpha^{H-1}}{1-\alpha}\|TV_k - V_k\|_\infty \\
&\implies \|H(V_k) - J^*\|_\infty - \|T^{H-1}V_k - J^*\|_\infty \leq (1+\alpha^m)\frac{\alpha^{H-1}}{1-\alpha}\|TV_k - V_k\|_\infty \\
&\implies \|H(V_k) - J^*\|_\infty \leq \|T^{H-1}V_k - J^*\|_\infty + (1+\alpha^m)\frac{\alpha^{H-1}}{1-\alpha}\|TV_k - V_k\|_\infty \\
&\implies \|H(V_k) - J^*\|_\infty \leq \alpha^{H-1}\|V_k - J^*\|_\infty \\
&+ (1+\alpha^m)\frac{\alpha^{H-1}}{1-\alpha}(1+\alpha)\|V_k - J^*\|_\infty \\
&\implies \|H(V_k) - J^*\|_\infty \leq \underbrace{\left(\alpha^{H-1} + (1+\alpha^m)\frac{\alpha^{H-1}}{1-\alpha}(1+\alpha)\right)}_{=:\beta}\|V_k - J^*\|_\infty.
\end{aligned}$$

Note that above, e is a vector of all 1s.

Step 4:

$$V_k \to J^*.$$

Proof of Step 4:

So far, we have the following rewrite of our iterates:

$$V_{k+1}(i) = (1 - \gamma_k p_{k,\mu_k}(i))V_k(i) + \gamma_k p_{k,\mu_k}(i)(H(V_k)(i) + z_k(i)),$$

where

$$\|H(V_k) - J^*\|_\infty \leq \beta\|V_k - J^*\|_\infty.$$

We define $\Delta_k := V_k - J^*$. Using Δ_k, the following holds:

$$\Delta_{k+1}(i) = (1 - \gamma_k p_{k,\mu_k}(i))\Delta_k(i) + \gamma_k p_{k,\mu_k}(i)(H(V_k) - J^* + z_k)(i).$$

Letting Y_k be a sequence defined recursively as follows:

$$Y_{k+1}(i) = (1 - \gamma_k p_{k,\mu_k}(i))Y_k(i) + \gamma_k p_{k,\mu_k}(i)z_k(i),$$

where $Y_0 = 0$. Since w_k is bounded for all k, $Y_k \to 0$ a.s.

We now define the following sequence X_k as follows: $X_k := \Delta_k - Y_k$.

Thus,

$$X_{k+1}(i) = (1 - \gamma_k p_{k,\mu_k}(i))X_k(i) + \gamma_k p_{k,\mu_k}(i)(H(V_k) - J^*)(i).$$

Taking absolute values on both sides we have:

$$\begin{aligned}
|X_{k+1}(i)| &= (1 - \gamma_k p_{k,\mu_k}(i))|X_k(i)| + \gamma_k p_{k,\mu_k}(i)|(H(V_k) - J^*)(i)| \\
&\leq (1 - \gamma_k p_{k,\mu_k}(i))\|X_k\|_\infty + \gamma_k p_{k,\mu_k}(i)\|H(V_k) - J^*\|_\infty \\
&\leq (1 - \gamma_k p_{k,\mu_k}(i))\|X_k\|_\infty + \gamma_k p_{k,\mu_k}(i)\beta\|V_k - J^*\|_\infty \\
&\leq (1 - \gamma_k p_{k,\mu_k}(i))\|X_k\|_\infty + \gamma_k p_{k,\mu_k}(i)\beta\|\Delta_k\|_\infty \\
&\leq (1 - \gamma_k p_{k,\mu_k}(i))\|X_k\|_\infty + \gamma_k p_{k,\mu_k}(i)\beta\|X_k\|_\infty + \beta\|Y_k\|_\infty \\
&\leq \max_i \Big[(1 - \gamma_k p_{k,\mu_k}(i))\|X_k\|_\infty + \gamma_k p_{k,\mu_k}(i)\beta\|X_k\|_\infty + \beta\|Y_k\|_\infty\Big].
\end{aligned}$$

We denote by $\tilde{\gamma}_k$ the $\gamma_k p_{k,\mu_k}(i)$ corresponding to a maximizing i in the above expression. Thus,

$$|X_{k+1}(i)| \leq (1 - \tilde{\gamma}_k)\|X_k\|_\infty + \tilde{\gamma}_k\beta\|X_k\|_\infty + \beta\|Y_k\|_\infty,$$

and since the right hand side of the inequality does not depend on i, we have that:

$$\|X_{k+1}\|_\infty \leq (1 - \tilde{\gamma}_k)\|X_k\|_\infty + \tilde{\gamma}_k\beta\|X_k\|_\infty + \beta\|Y_k\|_\infty.$$

Since $Y_k \to 0$ a.s., we conclude there must exist for all $\varepsilon > 0$ some $k(\varepsilon)$ such that for all $k > k(\varepsilon)$:

$$\|Y_k\|_\infty \leq \varepsilon.$$

So, for $k > k(\varepsilon)$, the following holds:

$$\|X_{k+1}\|_\infty \leq (1 - \tilde{\gamma}_k)\|X_k\|_\infty + \tilde{\gamma}_k\Big[\beta\|X_k\|_\infty + \beta\varepsilon\Big].$$

Rearranging terms, we have:

$$\begin{aligned}
&\|X_{k+1}\|_\infty \\
&\leq (1 - \tilde{\gamma}_k(1-\beta))\|X_k\|_\infty + \tilde{\gamma}_k(\beta\varepsilon) \\
&= (1 - \underbrace{\tilde{\gamma}_k(1-\beta)}_{=:\gamma'_k})\|X_k\|_\infty + \tilde{\gamma}_k(1-\beta)\Big[\frac{\beta\varepsilon}{1-\beta}\Big].
\end{aligned}$$

Now, consider any positive integer N. We define a sequence of random variables $\overline{X}_k^N$ for $k \geq N$, by setting $\overline{X}_N^N = \|X_N\|_\infty$ and

$$\overline{X}_{k+1}^N = (1 - \gamma'_k)\overline{X}_k^N + \gamma'_k\Big[\frac{\beta\varepsilon}{1-\beta}\Big]\forall k > N.$$

We will carry out a comparison of the sequence $\|X_k\|_\infty$ with the sequence $\overline{X}_k^N$. Consider the event that $k(\varepsilon) = N$, which we denote by A_N. We can use an easy inductive argument to show that for any N, for any sample path in A_N, and for all $k \geq N$, that $\|X_k\|_\infty \leq \overline{X}_k^N$. It is evident from the assumptions that $\sum_{k=0}^\infty \gamma'_k = \infty$ and hence $\overline{X}_k^N \to \frac{\beta\varepsilon}{1-\beta}e$ as $k \to \infty$. To see this, observe that when the terms of $\overline{X}_k^N$ are written out, we have:

$$\overline{X}_k^N = \prod_{\ell=N+1}^{k}(1 - \gamma'_\ell)\overline{X}_N^N + (1 - \prod_{\ell=N+1}^{k}(1 - \gamma'_\ell))\frac{\beta\varepsilon}{1-\beta}e$$

for $k > N$. Since γ'_k sums to infinity, we have that $\lim_{k\to\infty}\prod_{\ell=N}^{k}(1-\gamma'_\ell) = 0$, hence $\overline{X}_k^N \to \frac{\beta\varepsilon}{1-\beta}e$ as $k \to \infty$. Since ε can be chosen to be arbitrarily close to 0, for all sample paths in A_N, we have that:

$$\limsup_{k\to\infty} \overline{X}_k^N \leq 0.$$

Since the union of the events A_N is the entire sample space, we have:

$$\limsup_{k\to\infty} \overline{X}_k \leq 0.$$

From the definition of Δ_k and the fact that $Y_k \to 0$ a.s., we conclude that:

$$\limsup_{k\to\infty}\|\Delta_k\|_\infty = \limsup_{k\to\infty}\|V_k - J^*\|_\infty \leq 0 a.s,$$

and hence $V_k \to J^* a.s.$

D PROOF OF THEOREM 2

We define $V_k := \Phi\theta_k$ and write the sequence of iterates $\{V_k\}_{k=0}^{\infty}$ as follows:

$$V_{k+1} = (1-\gamma_k)V_k + \gamma_k(H(V_k) + z_k),$$

where $H(V_k) = E[\Phi\theta^{\mu_{k+1}}|\mathcal{F}_k]$ and $z_k := \Phi\theta^{\mu_{k+1}} - E[\Phi\theta^{\mu_{k+1}}|\mathcal{F}_k]$.

Proof Outline: We can use our assumption in Theorem 2 to show that:

$$\|H(V_k) - T^{H-1}V_k\|_\infty \leq \kappa\frac{\alpha^{H-1}}{1-\alpha}\|TV_k - V_k\|_\infty + \delta_{app},$$

for some κ and δ_{app}, which implies that:

$$\|H(V_k) - J^*\|_\infty \leq \underbrace{\left(\alpha^{H-1} + \kappa\frac{2\alpha^{H-1}}{1-\alpha}\right)}_{=:\beta}\|V_k - J^*\|_\infty + \delta_{app}.$$

Thus, $H(V_k)$ becomes almost a contraction with an error term, δ_{app}. We can then apply stochastic approximation techniques to show that:

$$\limsup_{k\to\infty}\|V_k - J^*\|_\infty \leq \frac{\delta_{app}}{1-\beta}.$$

To see this, suppose that there is no noise and so our iterates do not involve the noise averaging, i.e.,

$$V_{k+1} = \underbrace{\hat{J}^{\mu_{k+1}}}_{=:H(V_k)},$$

where $\|\hat{J}^{\mu_{k+1}} - J^{\mu_{k+1}}\|_\infty \leq \delta$. Then, we can trace the steps of the above, defining κ and δ_{app} as we did above and we

have the following:

$$
\begin{aligned}
&T^{H-1}V_k - \kappa\frac{\alpha^{H-1}}{1-\alpha}\|TV_k - V_k\|_\infty - \delta_{app} \leq V_{k+1} \leq T^{H-1}V_k + \kappa\frac{\alpha^{H-1}}{1-\alpha}\|TV_k - V_k\|_\infty + \delta_{app} \\
&\implies \|V_{k+1} - T^{H-1}V_k\|_\infty \leq \kappa\frac{\alpha^{H-1}}{1-\alpha}\|TV_k - V_k\|_\infty + \delta_{app} \\
&\implies \|V_{k+1} - J^*\|_\infty - \|T^{H-1}V_k - J^*\|_\infty \leq \kappa\frac{\alpha^{H-1}}{1-\alpha}\|TV_k - V_k\|_\infty + \delta_{app} \\
&\implies \|V_{k+1} - J^*\|_\infty \leq \|T^{H-1}V_k - J^*\|_\infty + \kappa\frac{\alpha^{H-1}}{1-\alpha}\|TV_k - V_k\|_\infty + \delta_{app} \\
&\implies \|V_{k+1} - J^*\|_\infty \leq \alpha^{H-1}\|V_k - J^*\|_\infty + \kappa\frac{\alpha^{H-1}}{1-\alpha}(1+\alpha)\|V_k - J^*\|_\infty + \delta_{app} \\
&\implies \|V_{k+1} - J^*\|_\infty \leq \left(\alpha^{H-1} + \kappa\frac{\alpha^{H-1}}{1-\alpha}(1+\alpha)\right)\|V_k - J^*\|_\infty + \delta_{app} \\
&\implies \|V_k - J^*\| \leq \left(\alpha^{H-1} + \kappa\frac{\alpha^{H-1}}{1-\alpha}(1+\alpha)\right)^k\|V_0 - J^*\| + \delta_{app}\sum_{i=0}^{k-1}\left(\alpha^{H-1} + \kappa\frac{\alpha^{H-1}}{1-\alpha}(1+\alpha)\right)^i
\end{aligned}
$$

Taking limits, we get the following:

$$
\begin{aligned}
\limsup_{k\to\infty}\|V_k - J^*\|_\infty &\leq \frac{\delta_{app}}{1 - \alpha^{H-1} - \kappa\frac{2\alpha^{H-1}}{1-\alpha}} \\
&= \frac{\delta_{app}}{1-\beta}.
\end{aligned}
$$

We will now prove our Theorem. We break the proof up into steps.

Step 1: We first obtain an upper bound for $\|H(V_k) - T^{H-1}V_k\|_\infty$ as follows:

$$\|H(V_k) - T^{H-1}V_k\|_\infty \leq \kappa\|T^{H-1}V_k - J^{\mu_{k+1}}\|_\infty + \delta_{app}$$

Proof of Step 1: We assume the existence of κ and δ_{app} in the statement of Theorem 2.

Step 2:

$$\|J^{\mu_{k+1}} - T^{H-1}V_k\|_\infty \leq \frac{\alpha^{H-1}}{1-\alpha}\|TV_k - V_k\|_\infty.$$

Proof of Step 2: Step 2 is a restatement of Lemma 1, which is proved in Appendix B.

Step 3:

$$\|H(V_k) - T^{H-1}V_k\|_\infty \leq (1+\kappa)\frac{\alpha^{H-1}}{1-\alpha}\|TV_k - V_k\|_\infty + \delta_{app}$$

Proof of Step 3:

We have from Step 1:

$$\|H(V_k) - T^{H-1}V_k\|_\infty \leq \kappa\|T^{H-1}V_k - J^{\mu_{k+1}}\|_\infty + \delta_{app}.$$

Plugging in the result of Step 2, we have:

$$\|H(V_k) - T^{H-1}V_k\|_\infty \leq \kappa\frac{\alpha^{H-1}}{1-\alpha}\|TV_k - V_k\|_\infty + \delta_{app}.$$

Step 4:

$$\|H(V_k) - J^*\|_\infty \leq \underbrace{\left(\alpha^{H-1} + \kappa\frac{2\alpha^{H-1}}{1-\alpha}\right)}_{=:\beta}\|V_k - J^*\|_\infty + \delta_{app}.$$

Proof of Step 4: We have

$$
\begin{aligned}
&T^{H-1}V_k - \kappa\frac{\alpha^{H-1}}{1-\alpha}\|TV_k - V_k\|_\infty e - \delta_{app}e \leq H(V_k)\\
&\leq T^{H-1}V_k + \kappa\frac{\alpha^{H-1}}{1-\alpha}\|TV_k - V_k\|_\infty e + \delta_{app}e\\
&\implies \|H(V_k) - T^{H-1}V_k\|_\infty \leq \kappa\frac{\alpha^{H-1}}{1-\alpha}\|TV_k - V_k\|_\infty + \delta_{app}\\
&\implies \|H(V_k) - J^*\|_\infty - \|T^{H-1}V_k - J^*\|_\infty \leq \kappa\frac{\alpha^{H-1}}{1-\alpha}\|TV_k - V_k\|_\infty\\
&+ \delta_{app}\\
&\implies \|H(V_k) - J^*\|_\infty \leq \|T^{H-1}V_k - J^*\|_\infty + \kappa\frac{\alpha^{H-1}}{1-\alpha}\|TV_k - V_k\|_\infty\\
&+ \delta_{app}\\
&\implies \|H(V_k) - J^*\|_\infty \leq \alpha^{H-1}\|V_k - J^*\|_\infty\\
&+ \kappa\frac{\alpha^{H-1}}{1-\alpha}(1+\alpha)\|V_k - J^*\|_\infty + \delta_{app}\\
&\implies \|H(V_k) - J^*\|_\infty \leq \underbrace{\left(\alpha^{H-1} + \kappa\frac{2\alpha^{H-1}}{1-\alpha}\right)}_{=:\beta}\|V_k - J^*\|_\infty\\
&+ \delta_{app}.
\end{aligned}
$$

Step 5:

$$\limsup_{k\to\infty}\|V_k - J^*\|_\infty \leq \frac{\delta_{app}}{1-\beta}.$$

Proof of Step 5:

So far, we have the following rewrite of our iterates:

$$V_{k+1} = (1-\gamma_k)V_k + \gamma_k(H(V_k) + z_k),$$

where

$$\|H(V_k) - J^*\|_\infty \leq \beta\|V_k - J^*\|_\infty + \delta_{app}.$$

We define $\Delta_k := V_k - J^*$. Using Δ_k, the following holds:

$$\Delta_{k+1} = (1-\gamma_k)\Delta_k + \gamma_k(H(V_k) - J^* + w_k).$$

Letting Y_k be a sequence defined recursively as follows:

$$Y_{k+1} = (1-\gamma_k)Y_k + \gamma_k w_k,$$

where $Y_0 = 0$. Since w_k is bounded for all k, $Y_k \to 0$ a.s.

We now define the following sequence X_k as follows:

$$X_k := \Delta_k - Y_k. \tag{11}$$

Thus,

$$X_{k+1} = (1-\gamma_k)X_k + \gamma_k(H(V_k) - J^*).$$

Taking norms on both sides gives:

$$\begin{aligned}\|X_{k+1}\|_\infty &\leq (1-\gamma_k)\|X_k\|_\infty + \gamma_k\|H(V_k) - H(J^*)\|_\infty \\ &\leq (1-\gamma_k)\|X_k\|_\infty + \gamma_k\Big[\beta\|V_k - J^*\|_\infty + \delta_{app}\Big] \\ &=\leq (1-\gamma_k)\|X_k\|_\infty + \gamma_k\Big[\beta\|\Delta_k\|_\infty + \delta_{app}\Big] \\ &\leq (1-\gamma_k)\|X_k\|_\infty + \gamma_k\Big[\beta\|X_k\|_\infty + \beta\|Y_k\|_\infty + \delta_{app}\Big].\end{aligned}$$

Since $Y_k \to 0$ a.s., we conclude there must exist for all $\varepsilon > 0$ some $k(\varepsilon)$ such that for all $k > k(\varepsilon)$:

$$\|Y_k\|_\infty \leq \varepsilon.$$

So, for $k > k(\varepsilon)$, the following holds:

$$\|X_{k+1}\|_\infty \leq (1-\gamma_k)\|X_k\|_\infty + \gamma_k\Big[\beta\|X_k\|_\infty + \beta\varepsilon + \delta_{app}\Big].$$

Rearranging terms, we have:

$$\begin{aligned}&\|X_{k+1}\|_\infty \\ &\leq (1-\gamma_k(1-\beta))\|X_k\|_\infty + \gamma_k(\beta\varepsilon + \delta_{app}) \\ &= (1-\underbrace{\gamma_k(1-\beta)}_{=:\gamma'_k})\|X_k\|_\infty + \gamma_k(1-\beta)\Big[\frac{\beta\varepsilon + \delta_{app}}{1-\beta}\Big].\end{aligned}$$

Now, consider any positive integer N. We define a sequence of random variables $\overline{X}_k^N$ for $k \geq N$, by setting $\overline{X}_N^N = \|X_N\|_\infty$ and

$$\overline{X}_{k+1}^N = (1-\gamma'_k)\overline{X}_k^N + \gamma'_k\Big[\frac{\beta\varepsilon + \delta_{app}}{1-\beta}\Big]\forall k > N.$$

We will carry out a comparison of the sequence $\|X_k\|_\infty$ with the sequence $\overline{X}_k^N$. Consider the event that $k(\varepsilon) = N$, which we denote by A_N. We can use an easy inductive argument to show that for any N, for any sample path in A_N, and for all $k \geq N$, that $\|X_k\|_\infty \leq \overline{X}_k^N$. It is evident from the assumption that $\sum_{k=0}^\infty \gamma_k = \infty$ and thus $\sum_{k=0}^\infty \gamma'_k = \infty$ that $\overline{X}_k^N \to \frac{\beta\varepsilon+\delta_{app}}{1-\beta}$ as $k \to \infty$. Since ε can be chosen to be arbitrarily close to 0, for all sample paths in A_N, we have that

$$\limsup_{k\to\infty} \overline{X}_k^N \leq \frac{\delta_{app}}{1-\beta}.$$

Since the union of the events A_N is the entire sample space, we have:

$$\limsup_{k\to\infty} \overline{X}_k \leq \frac{\delta_{app}}{1-\beta}.$$

From the definition of Δ_k and the fact that $Y_k \to 0$ a.s., we conclude that

$$\limsup_{k\to\infty}\|\Delta_k\|_\infty = \limsup_{k\to\infty}\|V_k - J^*\|_\infty \leq \frac{\delta_{app}}{1-\beta}.$$

Furthermore, since $V_k = \Phi\theta_k$, we have that

$$\limsup_{k\to\infty}\|\Phi\theta_k - J^*\|_\infty \leq \frac{\delta_{app}}{1-\beta}.$$

Step 6:

$$\limsup_{k\to\infty}\|J^{\mu_k}-J^*\|_\infty \leq \frac{\delta_{app}}{1-\beta}.$$

Proof of Step 6: Choose any $\varepsilon > 0$. Then, there exists $k(\varepsilon)$ such that the following holds for all $k > k(\varepsilon)$:

$$\|V_k - J^*\|_\infty \leq \Delta + \varepsilon. \tag{12}$$

Using (12), we can see that:

$$\begin{aligned}
&\|V_k - TV_k\|_\infty - \|TV_k - J^*\|_\infty \leq \|V_k - TV_k + TV_k - J^*\|_\infty \|V_k - J^*\|_\infty \leq \Delta + \varepsilon \\
&\implies \|V_k - TV_k\|_\infty \leq \|TV_k - J^*\|_\infty + \Delta + \varepsilon \\
&\implies \|V_k - TV_k\|_\infty \leq \alpha\|V_k - J^*\|_\infty + \Delta + \varepsilon \\
&\implies \|V_k - TV_k\|_\infty \leq \alpha(\Delta + \varepsilon) + \Delta + \varepsilon \\
&\implies \|V_k - TV_k\|_\infty \leq (1+\alpha)(\Delta + \varepsilon).
\end{aligned}$$

Thus,

$$\begin{aligned}
&-TV_k \leq -V_k + (1+\alpha)(\Delta+\varepsilon)e \\
\implies &-T^H V_k \leq -T^{H-1}V_k + \alpha^{H-1}(1+\alpha)(\Delta+\varepsilon)e \\
\implies &-T_{\mu_{k+1}}T^{H-1}V_k \leq -T^{H-1}V_k + \alpha^{H-1}(1+\alpha)(\Delta+\varepsilon)e.
\end{aligned}$$

Suppose that we apply the $T_{\mu_{k+1}}$ operator $\ell - 1$ times. Then, due to monotonicity and the fact that $T_\mu(J + ce) = T_\mu(J) + \alpha ce$, for any policy μ, we have the following:

$$-T^\ell_{\mu_{k+1}}T^{H-1}V_k \leq -T^{\ell-1}_{\mu_{k+1}}T^{H-1}V_k + \alpha^{\ell-1}\alpha^{H-1}(1+\alpha)(\Delta+\varepsilon)e.$$

Using a telescoping sum, we get the following inequality:

$$-T^j_{\mu_{k+1}}T^{H-1}V_k + T^{H-1}V_k \leq -\sum_{\ell=1}^{j}\alpha^{\ell-1}\alpha^{H-1}(\alpha+1)(\Delta+\varepsilon)e.$$

Taking the limit as $j \to \infty$ on both sides, we have the following:

$$-J^{\mu_{k+1}} + T^{H-1}V_k \leq -\frac{\alpha^{H-1}(\alpha+1)(\Delta+\varepsilon)}{1-\alpha}e.$$

Rearranging terms and subtracting J^* from both sides, we get the following:

$$\begin{aligned}
&-J^{\mu_{k+1}} + T^{H-1}V_k \leq -\frac{\alpha^{H-1}(\alpha+1)(\Delta+\varepsilon)}{1-\alpha}e \\
\implies &J^* - J^{\mu_{k+1}} \leq J^* - T^{H-1}V_k - \frac{\alpha^{H-1}(\alpha+1)(\Delta+\varepsilon)}{1-\alpha}e
\end{aligned}$$

Since $J^\mu \leq J^*$ for all policies μ, the above line implies that:

$$\begin{aligned}
\|J^* - J^{\mu_{k+1}}\|_\infty &\leq \|J^* - T^{H-1}V_k\|_\infty + \frac{\alpha^{H-1}(\alpha+1)(\Delta+\varepsilon)}{1-\alpha} \\
&\leq \alpha^{H-1}\|J^* - V_k\|_\infty + \frac{\alpha^{H-1}(\alpha+1)(\Delta+\varepsilon)}{1-\alpha} \\
&\leq \alpha^{H-1}(\Delta+\varepsilon) + \frac{\alpha^{H-1}(\alpha+1)(\Delta+\varepsilon)}{1-\alpha} \\
&= \frac{2\alpha^{H-1}(\Delta+\varepsilon)}{1-\alpha}.
\end{aligned}$$

Since the above holds for all $\varepsilon > 0$, we have the following conclusion:

$$\limsup_{k\to\infty}\|J^{\mu_{k+1}} - J^*\|_\infty \leq \frac{2\alpha^{H-1}\Delta}{1-\alpha}.$$

E PROOF OF THEOREM 3 AND EXPLANATION

Our iterates are:

$$V_{k+1} = (1-\gamma_k)V_k + \gamma_k(H(V_k) + z_k).$$

We have

$$\begin{aligned}
&X_{k+1} = (1-\gamma_k)X_k + \gamma_k(H(V_k)) \\
&\implies \|X_{k+1} - J^*\|_\infty = (1-\gamma_k)\|X_k - J^*\|_\infty + \gamma_k\|H(V_k) - J^*\|_\infty \\
&\implies \|X_{k+1} - J^*\|_\infty \leq (1-\gamma_k)\|X_k - J^*\|_\infty + \gamma_k((\alpha^{H-1} + \frac{2\kappa\alpha^{H-1}}{1-\alpha})\|V_k - J^*\|_\infty \\
&+ \delta_{app}) \\
&\implies \|X_{k+1} - J^*\|_\infty \leq (1-\gamma_k(1-\alpha^{H-1} + \frac{2\kappa\alpha^{H-1}}{1-\alpha}))\|X_k - J^*\|_\infty + \gamma_k\delta_{app} \\
&\implies E[\|X_{k+1} - J^*\|_\infty] \leq (1-\gamma_k(1-\alpha^{H-1} + \frac{2\kappa\alpha^{H-1}}{1-\alpha}))E[\|X_k - J^*\|_\infty] + \gamma_k\delta_{app} \\
&\implies E[\|V_{k+1} - J^*\|_\infty] \leq (1-\gamma_k(1-\alpha^{H-1} + \frac{2\kappa\alpha^{H-1}}{1-\alpha}))E[\|V_k - J^*\|_\infty] + \gamma_k(E[\|Y_{k+1}\|_\infty] + E[\|Y_k\|_\infty] + \delta_{app}),
\end{aligned}$$

where the last line follows from using the triangle inequality and the definition in (11).

Iterating, we have:

$$E[\|V_k - J^*\|_\infty] \leq \underbrace{\prod_{i=1}^{k-1} a_i \|V_0 - J^*\|_\infty}_{\text{initial condition error}} + \underbrace{\delta_{app}\sum_{j=1}^{k-1}\gamma_j \prod_{\ell=j+1}^{k-1} a_\ell}_{\text{error due to function approximation}} + \underbrace{\sum_{j=1}^{k-1}\gamma_j(E[\|Y_{j+1}\|_\infty] + E[\|Y_j\|_\infty]) \prod_{\ell=j+1}^{k-1} a_\ell}_{\text{error due to noise}}.$$

We note that since γ_k is square summable and sums to infinity, $Y_k \to 0$ a.s. and hence the error due to noise decreases over time. Additionally, since γ_k is square summable and sums to infinity, we have that $\prod_{i=1}^{k-1} a_i \to 0$, hence only the function approximation error remains.

We now obtain an upper bound for the $\|Y_j\|_\infty$ as follows. From the definition of Y_k in Appendix D, we have the following:

$$E(\|Y_k + 1\|^2) \leq (1-\gamma_k)^2 E(\|Y_k\|^2) + \gamma_k\sigma^2.$$

Furthermore, since $\|Y_0\| = 0$, we can iterate over k to get σ_j in Section 4.2.

F SECTION 4.3 - PROOFS

Recall that from the equation in (6), we rewrite our iterates as follows:

$$\theta^{\mu_{k+1}} = (\mathcal{P}_{1,k}\Phi)^+\mathcal{P}_{2,k}(T^m_{\mu_{k+1}}T^{H-1}V_k + w_k),$$

and thus,

$$\Phi\theta^{\mu_{k+1}} = \underbrace{\Phi(\mathcal{P}_{1,k}\Phi)^+\mathcal{P}_{2,k}}_{=:\mathcal{M}_k}(T^m_{\mu_{k+1}}T^{H-1}V_k + w_k)$$

We have:

$$
\begin{aligned}
&\|H(V_k) - J^{\mu_{k+1}}\|_\infty \\
&= \|E[\mathcal{M}_k(T^m_{\mu_{k+1}} T^{H-1} V_k + w_k) - J^{\mu_{k+1}} | \mathcal{F}_k]\|_\infty \\
&= \|E[\mathcal{M}_k(T^m_{\mu_{k+1}} T^{H-1} V_k + w_k) - \mathcal{M}_k(J^{\mu_{k+1}} + w_k) + \mathcal{M}_k(J^{\mu_{k+1}} + w_k) - J^{\mu_{k+1}} | \mathcal{F}_k]\|_\infty \\
&\leq \|E[\mathcal{M}_k(T^m_{\mu_{k+1}} T^{H-1} V_k + w_k) - \mathcal{M}_k(J^{\mu_{k+1}} + w_k) | \mathcal{F}_k]\|_\infty + \|E[\mathcal{M}_k(J^{\mu_{k+1}} + w_k) - J^{\mu_{k+1}} | \mathcal{F}_k]\|_\infty \\
&\leq \|E[\mathcal{M}_k(T^m_{\mu_{k+1}} T^{H-1} V_k + w_k) - \mathcal{M}_k(J^{\mu_{k+1}} + w_k) | \mathcal{F}_k]\|_\infty \\
&+ \underbrace{\sup_{k,\mu_k} \|E[\mathcal{M}_k(J^{\mu_{k+1}} + w_k) - J^{\mu_{k+1}} | \mathcal{F}_k]\|_\infty}_{=:\delta_{app}} \\
&= \|E[\mathcal{M}_k(T^m_{\mu_{k+1}} T^{H-1} V_k) - \mathcal{M}_k(J^{\mu_{k+1}}) | \mathcal{F}_k]\|_\infty + \delta_{app} \\
&= E[\|\mathcal{M}_k(T^m_{\mu_{k+1}} T^{H-1} V_k) - \mathcal{M}_k(J^{\mu_{k+1}})\|_\infty | \mathcal{F}_k] + \delta_{app} \\
&\leq E[\sup_k \|\mathcal{M}_k\|_\infty \|T^m_{\mu_{k+1}} T^{H-1} V_k - J^{\mu_{k+1}}\|_\infty | \mathcal{F}_k] + \delta_{app} \\
&= \underbrace{\sup_k \|\mathcal{M}_k\|_\infty}_{=:\delta_{FV}} \|T^m_{\mu_{k+1}} T^{H-1} V_k - J^{\mu_{k+1}}\|_\infty + \delta_{app} \\
&\leq \alpha^m \delta_{FV} \|T^{H-1} V_k - J^{\mu_{k+1}}\|_\infty + \delta_{app}.
\end{aligned}
$$

Using the above, we furthermore have that

$$
\|H(V_k) - T^{H-1} V_k\|_\infty \leq \underbrace{(1 + \alpha^m \delta_{FV})}_{=:\kappa} \|T^{H-1} V_k - J^{\mu_{k+1}}\|_\infty + \delta_{app}.
$$

G SECTION 4.4 - PROOFS

First, we will show that the gradient descent converges to

$$
{\theta_{k+1}}^* := \min_\theta \underbrace{\frac{1}{2} \|\underbrace{(\mathcal{P}_{1,k}\Phi)}_{=:A_k} \theta - \underbrace{\mathcal{P}_{2,k} \hat{J}^{\mu_{k+1}}}_{=:b_k}\|_2^2}_{=:f_k(\theta)} = (\mathcal{P}_{1,k}\Phi)^+ \mathcal{P}_{2,k}(T^m_{\mu_{k+1}} T^{H-1} V_k + w_k). \tag{13}
$$

To do so, we will show that

$$
\|\theta_{k+1,\eta} - \theta^*_{k+1}\|_\infty \leq (1 - \xi \sigma_{\mathcal{P}_{1,k}\Phi,\max})^\eta \|\Phi\|_\infty \|V_{k,1}\|_\infty \|\Sigma^{-1}_{k,1}\|_\infty \|U_k^\top \mathcal{P}_{2,k} \hat{J}^{\mu_{k+1}}\|_\infty,
$$

where the singular value decomposition of A_k is:

$$
A_k = U_k \begin{bmatrix} \Sigma_{k,1} & 0 \end{bmatrix} \begin{bmatrix} V_{k,1}^\top \\ V_{k,2}^\top \end{bmatrix} = U_k \Sigma_{k,1} V_{k,1}^\top.
$$

where U_k is a unitary matrix, $\Sigma_{k,1}$ is a rectangular diagonal matrix, and V_k is a unitary matrix.

Note that using the singular value decomposition of A_k, we can rewrite θ^*_{k+1} as follows:

$$
\theta^*_{k+1} = V_{k,1} \Sigma^{-1}_{k,1} U_k^\top b_k.
$$

The gradient of $f_k(\theta)$ is:

$$
\nabla f_k(\theta) = A_k^\top (A_k \theta - b_k).
$$

Using gradient descent with step size $\xi > 0$, our iterates of gradient descent are given by:

$$\begin{aligned}\theta_{k+1,\ell} &= \theta_{k+1,\ell-1} - \xi \nabla f_k(\theta_{k+1,\ell-1}) \\ &= (I - \xi A_k^\top A_k)\theta_{k+1,\ell-1} + \xi A_k^\top b.\end{aligned}$$

Hence,

$$\theta_{k+1,\ell} = \xi \sum_{\ell=0}^{\ell-1} (I - \xi A_k^\top A_k)^\ell A_k^\top b.$$

From the singular value decomposition of A_k, we have that

$$(I - \xi A_k^\top A_k)^\ell = V_k (I - \xi \Sigma_k^2)^\ell V_k^\top,$$

we can rewrite $\theta_{k+1,\ell}$ as follows:

$$\begin{aligned}\theta_{k+1,\ell} &= \xi \sum_{\ell=0}^{\ell-1} V_k (I - \xi \Sigma_k^2)^\ell V_k^\top A_k^\top b_k \\ &= \xi \sum_{\ell=0}^{\ell-1} V_k (I - \xi \Sigma_k^2)^\ell \Sigma_k U_k^\top b_k \\ &= \xi \sum_{\ell=0}^{\ell-1} V_k \begin{bmatrix} (I - \xi \Sigma_{k,1}^2)^\ell & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \Sigma_{k,1} \\ 0 \end{bmatrix} U_k^\top b_k \\ &= \xi \sum_{\ell=0}^{\ell-1} V_{k,1} (I - \xi \Sigma_{k,1}^2)^\ell \Sigma_{k,1} U_k^\top b_k.\end{aligned}$$

Since

$$\Sigma_{k,1}^{-1} = \xi (I - I + \xi \Sigma_{k,1}^2)^{-1} \Sigma_{k,1} = \xi \sum_{\ell=0}^{\infty} (I - \xi \Sigma_{k,1}^2)^\ell \Sigma_{k,1},$$

we further rewrite θ_{k+1}^* as follows:

$$\theta_{k+1}^* = \sum_{\ell=0}^{\infty} \xi V_{k,1} (I - \xi \Sigma_{k,1}^2)^\ell \Sigma_{k,1} U_k^\top b_k.$$

We now compute

$$
\begin{aligned}
&\|\theta^{\mu_{k+1}} - \theta^*_{k+1}\|_\infty \\
&\leq \|\theta_{k+1,\eta} - \theta^*_{k+1}\|_\infty \\
&= \|\xi \sum_{\ell=0}^{\eta} V_{k,1}(I - \xi\Sigma_{k,1}^2)^\ell \Sigma_{k,1} U_k^\top b_k - \xi \sum_{\ell=0}^{\infty} V_{k,1}(I - \xi\Sigma_{k,1}^2)^\ell \Sigma_{k,1} U_k^\top b_k\|_\infty \\
&= \|\xi \sum_{\ell=\eta}^{\infty} V_{k,1}(I - \xi\Sigma_{k,1}^2)^\ell \Sigma_{k,1} U_k^\top b_k\|_\infty \\
&\leq \|V_{k,1}\|_\infty \|\xi \sum_{\ell=\eta}^{\infty} (I - \xi\Sigma_{k,1}^2)^\ell \Sigma_{k,1} U_k^\top b_k\|_\infty \\
&= \|V_{k,1}\|_\infty \|\xi (I - \xi\Sigma_{k,1}^2)^\eta \sum_{\ell=0}^{\infty} (I - \xi\Sigma_{k,1}^2)^\ell \Sigma_{k,1} U_k^\top b_k\|_\infty \\
&= \|V_{k,1}\|_\infty \|(I - \xi\Sigma_{k,1}^2)^\eta\|_\infty \|\xi \sum_{\ell=0}^{\infty} (I - \xi\Sigma_{k,1}^2)^\ell \Sigma_{k,1} U_k^\top b_k\|_\infty \\
&\leq \|V_{k,1}\|_\infty \|I - \xi\Sigma_{k,1}^2\|_\infty^\eta \|\xi \sum_{\ell=0}^{\infty} (I - \xi\Sigma_{k,1}^2)^\ell \Sigma_{k,1} U_k^\top b_k\|_\infty \\
&= \|V_{k,1}\|_\infty \|I - \xi\Sigma_{k,1}^2\|_\infty^\eta \|\Sigma_{k,1}^{-1} U_k^\top b_k\|_\infty \\
&\leq (1 - \xi\sigma_{\mathcal{P}_{1,k}\Phi,\max})^\eta \|V_{k,1}\|_\infty \|\Sigma_{k,1}^{-1}\|_\infty \|U_k^\top b_k\|_\infty \\
&\leq (1 - \xi\sigma_{\mathcal{P}_{1,k}\Phi,\max})^\eta \|V_{k,1}\|_\infty \|\Sigma_{k,1}^{-1}\|_\infty \|U_k^\top \mathcal{P}_{2,k} \hat{J}^{\mu_{k+1}}\|_\infty,
\end{aligned}
$$

where $\sigma_{\mathcal{P}_{1,k}\Phi,\max}$ is the largest singular value squared of $\mathcal{P}_{1,k}\Phi$.

Note that the above implies that in order to obtain convergence of $\theta_{k+1,\eta}$ to θ^*_{k+1} as a function of η, we must have that $0 < \xi\sigma_{\mathcal{P}_{1,k}\Phi,\max} < 1$.

Thus, we have:

$$
\|H(V_k) - \Phi\theta^*_{k+1}\|_\infty \leq (1 - \xi\sigma_{\mathcal{P}_{1,k}\Phi,\max})^\eta \|\Phi\|_\infty \|V_{k,1}\|_\infty \|\Sigma_{k,1}^{-1}\|_\infty \|U_k^\top \mathcal{P}_{2,k} \hat{J}^{\mu_{k+1}}\|_\infty.
$$

Defining $\varepsilon := (1 - \xi\sigma_{\mathcal{P}_{1,k}\Phi,\max})^\eta \|\Phi\|_\infty \|V_{k,1}\|_\infty \|\Sigma_{k,1}^{-1}\|_\infty \|U_k^\top \mathcal{P}_{2,k} \hat{J}^{\mu_{k+1}}\|_\infty$ and using $\mathcal{M}_k$ and δ_{app} as defined in Appendix E, we obtain:

$$
\begin{aligned}
&\|H(V_k) - J^{\mu_{k+1}}\|_\infty \\
&= \|E[\mathcal{M}_k(T^m_{\mu_{k+1}} T^{H-1} V_k + w_k) - J^{\mu_{k+1}} | \mathcal{F}_k]\|_\infty + \varepsilon \\
&= \|E[\mathcal{M}_k(T^m_{\mu_{k+1}} T^{H-1} V_k + w_k) - \mathcal{M}_k(J^{\mu_{k+1}} + w_k) + \mathcal{M}_k(J^{\mu_{k+1}} + w_k) - J^{\mu_{k+1}} | \mathcal{F}_k]\|_\infty + \varepsilon \\
&\leq \|E[\mathcal{M}_k(T^m_{\mu_{k+1}} T^{H-1} V_k + w_k) - \mathcal{M}_k(J^{\mu_{k+1}} + w_k) | \mathcal{F}_k]\|_\infty + \|E[\mathcal{M}_k(J^{\mu_{k+1}} + w_k) - J^{\mu_{k+1}} | \mathcal{F}_k]\|_\infty + \varepsilon \\
&\leq \|E[\mathcal{M}_k(T^m_{\mu_{k+1}} T^{H-1} V_k + w_k) - \mathcal{M}_k(J^{\mu_{k+1}} + w_k) | \mathcal{F}_k]\|_\infty \\
&+ \underbrace{\sup_{k,\mu_k} \|E[\mathcal{M}_k(J^{\mu_{k+1}} + w_k) - J^{\mu_{k+1}} | \mathcal{F}_k]\|_\infty}_{=:\delta_{app}} + \varepsilon \\
&= \|E[\mathcal{M}_k(T^m_{\mu_{k+1}} T^{H-1} V_k) - \mathcal{M}_k(J^{\mu_{k+1}}) | \mathcal{F}_k]\|_\infty + \delta_{app} + \varepsilon \\
&= E[\|\mathcal{M}_k(T^m_{\mu_{k+1}} T^{H-1} V_k) - \mathcal{M}_k(J^{\mu_{k+1}})\|_\infty | \mathcal{F}_k] + \delta_{app} + \varepsilon \\
&\leq E[\sup_k \|\mathcal{M}_k\|_\infty \|T^m_{\mu_{k+1}} T^{H-1} V_k J^{\mu_{k+1}}\|_\infty | \mathcal{F}_k] + \delta_{app} + \varepsilon \\
&= \underbrace{\sup_k \|\mathcal{M}_k\|_\infty}_{=:\delta_{FV}} \|T^m_{\mu_{k+1}} T^{H-1} V_k J^{\mu_{k+1}}\|_\infty + \delta_{app} + \varepsilon \\
&\leq \alpha^m \delta_{FV} \|T^{H-1} V_k - J^{\mu_{k+1}}\|_\infty + \delta_{app} + \varepsilon.
\end{aligned}
$$

Using the above, we furthermore have that

$$\|H(V_k) - T^{H-1}V_k\|_\infty \leq \underbrace{(1+\alpha^m\delta_{FV})}_{=:\kappa}\|T^{H-1}V_k - J^{\mu_{k+1}}\|_\infty + \delta_{app}$$
$$+ (1-\xi\sigma_{\mathcal{P}_{1,k}\Phi,\max})^\eta \|\Phi\|_\infty \|V_{k,1}\|_\infty \|\Sigma_{k,1}^{-1}\|_\infty \|U_k^\top \mathcal{P}_{2,k}\hat{J}^{\mu_{k+1}}\|_\infty.$$

The new κ and δ_{app} are apparent from the above.

H SECTION 4.5 - PROOFS

$$(\pi_{\mu_{k+1},\min})^2 E[\|\Phi\theta^{\mu_{k+1}*} - \Phi\theta^{\mu_{k+1}}\|_\infty^2] \leq (\pi_{\mu_{k+1},\min})^2 E[\|\Phi\theta^{\mu_{k+1}*} - \Phi\theta^{\mu_{k+1}}\|_2^2] \leq E[\|\Phi\theta^{\mu_{k+1}*} - \Phi\theta^{\mu_{k+1}}\|_D^2] \leq \delta_{T,\mu_{k+1}}.$$

Using Jensen's inequality, we have:

$$(\pi_{\mu_{k+1},\min})^2 (E[\|\Phi\theta^{\mu_{k+1}*} - \Phi\theta^{\mu_{k+1}}\|_\infty])^2 \leq (\pi_{\mu_{k+1},\min})^2 E[\|\Phi\theta^{\mu_{k+1}*} - \Phi\theta^{\mu_{k+1}}\|_\infty^2] \leq \delta_{T,\mu_{k+1}}$$
$$\implies \pi_{\mu_{k+1},\min} E[\|\Phi\theta^{\mu_{k+1}*} - \Phi\theta^{\mu_{k+1}}\|_\infty] \leq \sqrt{\delta_{T,\mu_{k+1}}}$$
$$\implies E[\|\Phi\theta^{\mu_{k+1}*} - \Phi\theta^{\mu_{k+1}}\|_\infty] \leq \frac{\sqrt{\delta_{T,\mu_{k+1}}}}{\pi_{\mu_{k+1},\min}}$$
$$\implies E[\|\Phi\theta^{\mu_{k+1}*} - J^{\mu_{k+1}} + J^{\mu_{k+1}} - \Phi\theta^{\mu_{k+1}}\|_\infty] \leq \frac{\sqrt{\delta_{T,\mu_{k+1}}}}{\pi_{\mu_{k+1},\min}}$$
$$\implies E[\|J^{\mu_{k+1}} - \Phi\theta^{\mu_{k+1}}\|_\infty] \leq \sup_{\mu_{k+1}} \|\Phi\theta^{\mu_{k+1}*} - J^{\mu_{k+1}}\|_\infty + \frac{\sqrt{\delta_{T,\mu_{k+1}}}}{\pi_{\mu_{k+1},\min}},$$

where the last inequality follows from applying the reverse triangle inequality and then taking the supremum over all policies μ_{k+1}.

Finally, we use Jensen's inequality again to obtain the following:

$$\|E[\Phi\theta^{\mu_{k+1}}] - J^{\mu_{k+1}}\|_\infty = \|E[J^{\mu_{k+1}} - \Phi\theta^{\mu_{k+1}}]\|_\infty \leq \sup_{\mu_{k+1}} \|\Phi\theta^{\mu_{k+1}*} - J^{\mu_{k+1}}\|_\infty + \frac{\sqrt{\delta_{T,\mu_{k+1}}}}{\pi_{\mu_{k+1},\min}}.$$

Thus, we can combine the $\delta_{T,\mu_{k+1}}$ in (Bhandari et al., 2018) with the above terms to obtain a δ_{app} and our calculations in Section 4.5 give $\kappa = 1$.

I CONNECTION OF MONTE CARLO ES TO PRACTICE

We make several remarks regarding the connection of Monte Carlo ES to practice. While AlphaZero [Silver et al., 2017b] uses techniques such as function approximation and lookahead through planning algorithms in the form of Monte Carlo Tree Search (MCTS), Monte Carlo ES is nonetheless a Monte Carlo algorithm since it uses full trajectories and their returns to estimate loss functions. Additionally, the AlphaZero algorithm uses returns from all states visited by the trajectories to make updates instead of only the first state.