ReVar: Strengthening Policy Evaluation via Reduced Variance Sampling

Subhojyoti Mukherjee[1] **Josiah P. Hanna**[2] **Robert Nowak**[1]

[1]Department of Electrical and Computer Engineering, University of Wisconsin-Madison, USA
[2]Computer Sciences Department, University of Wisconsin-Madison, USA

Abstract

This paper studies the problem of data collection for policy evaluation in Markov decision processes (MDPs). In policy evaluation, we are given a *target* policy and asked to estimate the expected cumulative reward it will obtain in an environment formalized as an MDP. We develop theory for optimal data collection within the class of tree-structured MDPs by first deriving an oracle data collection strategy that uses knowledge of the variance of the reward distributions. We then introduce the **Re**duced **Var**iance Sampling (ReVar) algorithm that approximates the oracle strategy when the reward variances are unknown a priori and bound its sub-optimality compared to the oracle strategy. Finally, we empirically validate that ReVar leads to policy evaluation with mean squared error comparable to the oracle strategy and significantly lower than simply running the target policy.

1 INTRODUCTION

In reinforcement learning (RL) applications, there is often a need for policy evaluation to determine (or estimate) the expected return (future cumulative reward) of a given policy. Policy evaluation is also required in other sequential decision-making settings outside of RL. For example, testing an autonomous vehicle stack or ad-serving system can be seen as policy evaluation applications. Accurate and data efficient policy evaluation is critical for safe and trust-worthy deployment of autonomous systems.

This paper studies data collection for low mean squared error (MSE) policy evaluation in sequential decision-making tasks formalized as Markov decision processes (MDPs). The objective of policy evaluation is to estimate the expected return that will be obtained by running a *target policy* which is a given probabilistic mapping from states to actions.

To evaluate the target policy, we require data from the environment in which it will be deployed. Collecting data requires running a (possibly non-stationary) *behavior* policy to generate state-action-reward trajectories. Our goal is to find a behavior policy that leads to a minimum MSE evaluation of the target policy.

The most natural choice is *on-policy sampling* in which we use the target policy as the behavior policy. However, we show that in some cases this choice is far from optimal (e.g., Figure 2 in our empirical analysis) as it fails to actively take actions from which the expected return is uncertain. Instead, an optimal behavior policy should take actions in any given state to reduce uncertainty in the current estimate of the expected return from that state.

Our paper makes the following main contributions. We first derive an optimal "oracle" behavior policy for finite tree-structured MDPs *assuming oracle access to the MDP transition probabilities and variances of the reward distributions*. Sampling trajectories according to the oracle behavior policy minimizes the MSE of the estimator of the target policy's expected. As a special case (depth 1 tree MDPs), we recover the optimal behavior policy for multi-armed bandits Carpentier et al. [2015].

We then introduce a practical algorithm, **Re**duced **Var**iance Sampling (ReVar), that adaptively learns the optimal behavior policy by observing rewards and adjusting the policy to select actions that reduce the MSE of the estimator. The main idea of ReVar is to plug-in upper-confidence bounds on the reward distribution variances to approximate the oracle behavior policy. We define a notion of policy evaluation regret compared to the oracle behavior policy, and bound the regret of ReVar. The regret converges rapidly to 0 as the number of sampled episodes grows, theoretically guaranteeing that ReVar quickly matches the performance of the oracle policy. Finally, we implement ReVar and show it leads to low MSE policy evaluation in both a tree-structured and a general finite-horizon MDP. Taken together, our contributions provide a theoretical foundation towards optimal

Proceedings of the 38th Conference on Uncertainty in Artificial Intelligence (UAI 2022), PMLR 180:1413–1422.

data collection for policy evaluation in MDPs.

The remainder of the paper is organized as follows. In Section 3 we reformulate our problem in the bandit setting and discuss related bandit works. In Section 4 we extend the bandit formulation to the tree MDP. Finally we introduce the more general Directed Acyclic Graph (DAG) MDP in Section 5 and discuss some limitations of our sampling behavior. We show numerical experiments in Section 6 and conclude in Section 7.

2 BACKGROUND

In this section, we introduce notation, define the policy evaluation problem, and discuss the prior literature.

2.1 NOTATION

A finite-horizon Markov Decision Process, $\mathbf{M}$, is the tuple $(\mathcal{S}, \mathcal{A}, P, R, \gamma, d_0, L)$, where $\mathcal{S}$ is a finite set of states, $\mathcal{A}$ is a finite set of actions, $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is a state transition function, R is the reward distribution (formalized below), $\gamma \in [0, 1)$ is the discount factor, d_0 is the starting state distribution, and L is the maximum episode length. A (stationary) policy, $\pi : \mathcal{S} \times \mathcal{A} \to [0, 1]$, is a probability distribution over actions conditioned on a given state. We assume data can only be collected through episodic interaction: an agent begins in state $S_0 \sim d_0$ and then at each step t takes an action $A_t \sim \pi(\cdot|S_t)$ and proceeds to state $S_{t+1} \sim P(\cdot|S_t, A_t)$. Interaction terminates in at most L steps. Each time the agent takes action a_t in state s_t it observes a reward $R_t \sim R(s_t, a_t)$. We assume $R(s, a) = \mathcal{P}(\mu(s, a), \sigma^2(s, a))$, where $\mathcal{P}$ denotes a parametric distribution with mean $\mu(s, a)$ and variance $\sigma^2(s, a)$. The entire interaction produces a trajectory $H \coloneqq \{(S_t, A_t, R_t)\}_{t=1}^{L}$. We assume d_0 is known but P and the reward distributions are unknown. We define the value of a policy as: $v(\pi) \coloneqq \mathbb{E}_\pi[\sum_{t=1}^{L} \gamma^{t-1} R_t]$, where $\mathbb{E}_\pi$ is the expectation w.r.t. trajectories sampled by following π.

We will make use of the fact that the value of a policy can be written as: $v(\pi) = \mathbb{E}[v_0^\pi(S_0)|S_0 \sim d_0]$ where,

$$v_t^\pi(s) \coloneqq \sum_a \pi(a|s)\mu(s, a) + \gamma \sum_{s'} P(s'|s, a) v_{t+1}^\pi(s')$$

for $t \leq L$ and $v_t^\pi(s) = 0$ for $t > L$.

2.2 POLICY EVALUATION

We now formally define our objective. We are given a target policy, π, for which we want to estimate $v(\pi)$. To estimate $v(\pi)$ we will generate a set of K trajectories where each trajectory is generated by following some policy. Let $H^k \coloneqq \{s_t^k, a_t^k, R_t^k(s_t^k, a_t^k)\}_{t=1}^{L}$ be the trajectory collected in episode k and let b^k be the policy ran to produce H^k. The entire set of collected data is given as $\mathcal{D} \coloneqq \{H^k, b^k\}_{k=1}^{K}$.

Once $\mathcal{D}$ is collected, we estimate $v(\pi)$ with a certainty-equivalence estimate Sutton [1988]. Suppose $\mathcal{D}$ consists of $n = KL$ state-action transitions. We define the random variable representing the estimated future reward from state s at time-step t as:

$$Y_n(s, t) \coloneqq \sum_a \pi(a|s)\widehat{\mu}(s, a) + \gamma \sum_{s'} \widehat{P}(s'|s, a) Y_n(s', t+1),$$

where $Y_n(s, t+1) \coloneqq 0$ if $t \geq L$, $\widehat{\mu}(s, a)$ is an estimate of $\mu(s, a)$ and $\widehat{P}(s'|s, a)$ is an estimate of $P(s'|s, a)$, both computed from $\mathcal{D}$. Finally, the estimate of $v(\pi)$ is computed as $Y_n \coloneqq \sum_s d_0(s) Y_n(s, 0)$. In the policy evaluation literature, the certainty-equivalence estimator is also known as the direct method Jiang and Li [2016] and, in tabular settings, can be shown to be equivalent to batch temporal-difference estimators Sutton [1988], Pavse et al. [2020]. Thus, it is representative of two types of policy evaluation estimators that often give strong empirical performance Voloshin et al. [2019].

Our objective is to determine the sequence of behavior policies that minimize error in estimation of $v(\pi)$. Formally, we seek to minimize mean squared error which is defined as: $\mathbb{E}_{\mathcal{D}}\left[\left(Y_n - v(\pi)\right)^2\right]$ where the expectation is over the collected data set $\mathcal{D}$.

2.3 RELATED WORK

Our paper builds upon work in the bandit literature for optimal data collection for estimating a weighted sum of the mean reward associated with each arm. Antos et al. [2008] study estimating the mean reward of each arm equally well and show that the optimal solution is to pull each arm proportional to the variance of its reward distribution. Since the variances are unknown a priori, they introduce an algorithm that pulls arms in proportion to the empirical variance of each reward distribution. Carpentier et al. [2015] extend this work by introducing a weighting on each arm that is equivalent to the target policy action probabilities in our work. They show that the optimal solution is then to pull each arm proportional to the product of the standard deviation of the reward distribution and the arm weighting. Instead of using the empirical standard deviations, they introduce an upper confidence bound on the standard deviation and use it to select actions. Our work is different from these earlier works in that we consider more general tree-structured MDPs of which bandits are a special case.

In RL and MDPs, exploration is widely studied with the objective of finding the optimal policy. Prior work attempts to balance exploration to reduce uncertainty with exploitation to converge to the optimal policy. Common approaches are based on reducing uncertainty [Osband et al., 2016,

O'Donoghue et al., 2018] or incentivizing visitation of novel states [Barto, 2013, Pathak et al., 2017, Burda et al., 2018]. These works differ from our work in that we focus on evaluating a fixed policy rather than finding the optimal policy. In our problem, the trade-off becomes balancing taking actions to reduce uncertainty with taking actions that the target policy is likely to take.

Our work is similar in spirit to work on adaptive importance sampling [Rubinstein and Kroese, 2013] which aims to lower the variance of Monte Carlo estimators by adapting the data collection distribution. Adaptive importance sampling was used by Hanna et al. [2017] to lower the variance of policy evaluation in MDPs. It has also been used to lower the variance of policy gradient RL algorithms [Bouchard et al., 2016, Ciosek and Whiteson, 2017]. AIS methods attempt to find a single optimal sampling distribution whereas our approach attempts to reduce uncertainty in the estimated mean rewards. In a similar spirit, Talebi and Maillard [2019] adapt the behavior policy to minimize error in estimating the transition model P.

3 OPTIMAL DATA COLLECTION IN MULTI-ARMED BANDITS

Before we address optimal data collection for policy evaluation in MDPs, we first revisit the problem in the bandit setting as addressed by earlier work Carpentier et al. [2015]. The bandit setting provides intuition for how a good data collection strategy should select actions, though it falls short of an entire solution for MDPs.

Observe that the policy value in a bandit problem is defined as $v(\pi) \coloneqq \sum_{a=1}^{A} \pi(a)\mu(a)$ where the bandit consist of a single state s and A actions indexed as $a = 1, 2, \ldots, A$. In this setting, the horizon $L = 1$ so we return to the same state after taking an action a at time t. Hence, we drop the state s from our standard notation.

Suppose we have a budget of n samples to divide between the arms and let $T_n(1), T_n(2), \ldots, T_n(A)$ be the number of samples allocated to actions $1, 2, \ldots, A$ at the end of n rounds. We define the estimate:

$$Y_n \coloneqq \sum_{a=1}^{A} \frac{\pi(a)}{T_n(a)} \sum_{h=1}^{T_n(a)} R_h(a) = \sum_{a=1}^{A} \pi(a)\widehat{\mu}(a). \qquad (1)$$

where, $R_h(a)$ is the h^{th} reward received after taking action a. Note that, once all actions where $\pi(a) > 0$ have been tried, Y_n is an unbiased estimator of $v(\pi)$ since $\widehat{\mu}(a)$ is an unbiased estimator of $\mu(a)$. Thus, reducing MSE requires allocating the n samples to reduce variance. As shown by Carpentier et al. [2015], the minimal-variance allocation is given by pulling each arm with the proportion $b^\star(a) \propto \pi(a)\sigma(a)$. Though this result was previously shown, we prove it for completeness in Proposition 1 in Appendix A. Intuitively, there is more uncertainty about the mean reward for actions with higher variance reward distributions. Selecting these actions more often is needed to offset higher variance. The optimal proportion also takes π into account as a high variance mean reward estimate for one action can be acceptable if π would rarely take that action.

Note that sampling according to eq. (1) introduces unnecessary variance compared to deterministically selecting actions to match the optimal proportion. Since the variances are typically unknown, a number of works in the bandit community propose different approaches to estimate the variances for both basic bandits and several related extensions [Antos et al., 2008, Carpentier and Munos, 2011, 2012, Carpentier et al., 2015, Neufeld et al., 2014]. Finally, note that incorporating variance aware techniques has been studied in multi-armed bandits [Audibert et al., 2009, Mukherjee et al., 2018]. However, these works tend to focus on regret minimization, whereas we focus on MSE reduction. However, none of of these works address the fundamental challenge that MDPs bring – action selection must account for both immediate variance reduction in the current state as well as variance reduction in future states visited. In the next section, we begin to address this challenge by deriving minimal-variance action proportions for tree-structured MDPs.

4 OPTIMAL DATA COLLECTION IN TREE MDPS

In this section, we derive the optimal action proportions for tree-structured MDPs assuming the variances of the reward distributions are known, introduce an algorithm that approximates the optimal allocation when the variances are unknown, and bound the finite-sample MSE of this algorithm. Tree MDPs are a straightforward extension of the multi-armed bandit model to capture the fact that the optimal allocation for each action in a given state must consider the future states that could arise from taking that action.

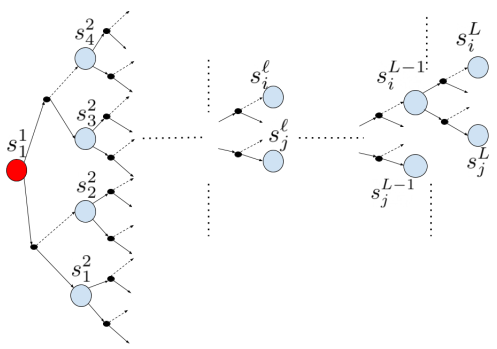

Figure 1: An L-depth tree with 2 actions at each state.

We first define a discrete tree MDP as follows:

Definition 1. (Tree MDP) An MDP is a discrete tree MDP

$\mathbf{T} \subset \mathbf{M}$ (see Figure 1) if the following holds:

(1) There are L levels indexed by ℓ where $\ell = 1, 2, \ldots, L$.

(2) Every state is represented as s_i^ℓ where ℓ is the level of the state s indexed by i.

(3) The transition probabilities are such that one can only transition from a state in level ℓ to one in level $\ell+1$ and each non-initial state can only be reached through one other state and only one action in that state. Formally, $\forall s'$, $P(s'|s,a) \neq 0$ for only one state-action pair s, a and if s' is in level $\ell+1$ then s is in level ℓ. Finally, $P(s_j^{L+1}|s_i^L, a) = 0, \forall a$.

(4) For simplicity, we assume that there is a single starting state s_1^1 (called the root). It is easy to extend our results to multiple starting states with a starting state distribution, d_0, by assuming that there is only one action available in the root that leads to each possible start state, s, with probability $d_0(s)$. The leaf states are denoted as s_i^L.

(5) The interaction stops after L steps in state s_i^L after taking an action a and observing the reward $R_L(s_i^L, a)$.

Note that, because we assume a single initial state, s_1^1, we have that estimating $v(\pi)$ is equivalent to estimating $v(s_1^1)$. A similar Tree MDP model has been previously used in theoretical analysis by Jiang and Li [2016]; our model is slightly more general as we consider per-step stochastic rewards whereas Jiang and Li [2016] only consider deterministic rewards at the end of trajectories.

4.1 ORACLE DATA COLLECTION

We first consider an oracle data collection strategy which knows the variance of all reward distributions and knows the state transition probabilities. After observing n state-action-reward tuples, the oracle computes the following estimate of $v^\pi(s_1^1)$ (or equivalently $v(\pi)$):

$$
\begin{aligned}
Y_n(s_1^1) \coloneqq & \sum_{a=1}^{A} \pi(a|s_1^1) \Bigg(\frac{1}{T_n(s_1^1, a)} \sum_{h=1}^{T_n(s_1^1,a)} R_h(s_1^1, a) \\
& + \gamma \sum_{s_j^{\ell+1}} P(s_j^{\ell+1}|s_1^1, a) Y_n(s_j^2) \Bigg) \\
= & \sum_{a=1}^{A} \pi(a|s_1^1) \Big(\widehat{\mu}(s_1^1, a) + \gamma \sum_{s_j^{\ell+1}} P(s_j^{\ell+1}|s_1^1, a) Y_n(s_j^2) \Big) \quad (2)
\end{aligned}
$$

where $T_n(s,a)$ denotes the number of times that the oracle took action a in state s. Note that in Section 2 we define $Y_n(s,t)$ but now we use $Y_n(s)$ as timestep is implicit in the layer of the tree. Also (2) differs from the estimator defined in Section 2.2 as it uses the true transition probabilities, P, instead of their empirical estimate, $\widehat{P}$. The MSE of Y_n is:

$$
\begin{aligned}
& \mathbb{E}_{\mathcal{D}}[\big(Y_n(s_1^1) - v^\pi(s_1^1)\big)^2] \\
& \quad = \mathbf{Var}(Y_n(s_1^1)) + \text{bias}^2(Y_n(s_1^1)). \qquad (3)
\end{aligned}
$$

The bias of this estimator becomes zero once all (s,a)-pairs with $\pi(a|s) > 0$ have been visited a single time, thus we focus on reducing $\mathbf{Var}(Y_n(s_1^1))$. Before defining the oracle data collection strategy, we first state an assumption on $\mathcal{D}$.

Assumption 1. *The data $\mathcal{D}$ collected over n state-action-reward samples has at least one observation of each state-action pair, (s,a), for which $\pi(a|s) > 0$.*

Assumption 1 ensures that Y_n is an unbiased estimator of $v(\pi)$ so that reducing MSE is equivalent to reducing variance. Before stating our main result, we provide intuition with a lemma that gives the optimal proportion for each action in a 2-depth tree.

Lemma 1. *Let $\mathbf{T}$ be a 2-depth stochastic tree MDP as defined in Definition 1 (see Figure 1 in Appendix B). Let $Y_n(s_1^1)$ be the estimated return of the starting state s_1^1 after observing n state-action-reward samples. Note that $v^\pi(s_1^1)$ is the expectation of $Y_n(s_1^1)$ under Assumption 1. Let $\mathcal{D}$ be the observed data over n state-action-reward samples. Minimal MSE, $\mathbb{E}_{\mathcal{D}}[(Y_n(s_1^1) - v^\pi(s_1^1))^2]$, is obtained by taking actions in each state in the following proportions:*

$$
\begin{aligned}
b^*(a|s_j^2) & \propto \pi(a|s_j^2)\sigma(s_j^2, a) \\
b^*(a|s_1^1) & \propto \sqrt{\pi^2(a|s_1^1)\Big[\sigma^2(s_1^1, a) + \gamma^2 \sum_{s_j^2} P(s_j^2|s_1^1, a) B^2(s_j^2)\Big]},
\end{aligned}
$$

where, $B(s_j^2) = \sum_a \pi(a|s_j^2)\sigma(s_j^2, a)$.

Proof (Overview): We decompose the MSE into its variance and bias terms and show that Y_n is unbiased under Assumption 1. Next note that the reward in the next state is conditionally independent of the reward in the current state given the current state and action. Hence we can write the variance in terms of the variance of the estimate in the initial state and the variance of the estimate in the final layer. We then rewrite the total samples of a state-action pair i.e $T_n(s_i^\ell, a)$ in terms of the proportion of the number of times the action was sampled in the state i.e $b(a|s_i^\ell)$. To do so, we take into account the tree structure to derive the expected proportion of times that action a is taken in each state in layer 2 as follows:

$$
b(a|s_i^2) = \frac{T_n(s_i^2, a)}{\sum_{a'} T_n(s_i^2, a')} \overset{(a)}{=} \frac{T_n(s_i^2, a)/n}{P(s_i^2|s_1^1, a) T_n(s_1^1, a)/n}
$$

where in (a) the action a is used to transition to state s_j^2 from s_1^1 and so $\sum_a T_n(s_i^2, a) = P(s_i^2|s_1^1, a) T_n(s_1^1, a)$. We next substitute the $b(a|s_i^\ell)$ for each state-action pair into

the variance expression and determine the b values that minimize the expression subject to $\forall s, \sum_a b(a|s) = 1$ and $\forall s, b(a|s) > 0$. The full proof is given in Appendix B. ■

Note that the optimal proportion in the leaf states, $b^*(a|s_j^2)$, is the same as in Carpentier and Munos [2011] (see Proposition 1) as terminal states can be treated as bandits in which actions do not affect subsequent states. The key difference is in the root state, s_1^1, where the optimal action proportion, $b^*(a|s_1^1)$ depends on the expected leaf state normalization factor $B(s_j^2)$ where s_j^2 is a state sampled from $P(\cdot|s_1^1, a)$. The normalization factor, $B(s_i^2)$, captures the total contribution of state s_i^2 to the variance of Y_n and thus actions in the root state must be chosen to 1) reduce variance in the immediate reward estimate and to 2) get to states that contribute more to the variance of the estimate. We explore the implications of the oracle action proportions in Lemma 1 with the following two examples.

Example 1. (Child Variance matters) Consider a 2-depth, 2-action tree MDP $\mathbf{T}$ with deterministic P, i.e., $P(s_2^2|s_1^1, 2) = P(s_1^2|s_1^1, 1) = 1$ and $\gamma = 1$ (see Figure 2 (Left) in Appendix C). Suppose the target policy is the uniform distribution in all states so that $\forall(s, a), \pi(a|s) = \frac{1}{2}$. The reward distribution variances are given by $\sigma^2(s_1^1, 1) = 400$, $\sigma^2(s_1^1, 2) = 600$, $\sigma^2(s_1^2, 1) = 400$, $\sigma^2(s_1^2, 2) = 400$, $\sigma^2(s_2^2, 1) = 4$, and $\sigma^2(s_2^2, 2) = 4$. So the right sub-tree at s_1^1 has higher variance (larger B-value) than the left sub-tree. Following the sampling rule in Lemma 1 we can show that $b^*(1|s_1^1) > b^*(2|s_1^1)$ (the full calculation is given in Appendix C). Hence the right sub-tree with higher variance will have a higher proportion of pulls which allows the oracle to get to the high variance s_1^2. Observe that treating s_1^1 as a bandit leads to choosing action 2 more often as $\sigma^2(s_1^1, 2) > \sigma^2(s_1^1, 1)$. However, taking action 2 leads to state s_2^2 which contributes much less to the total variance. Thus, this example highlights the need to consider the variance of subsequent states.

Example 2. (Transition Model matters) Consider a 2-depth, 2-action tree MDP $\mathbf{T}$ in which we have $P(s_1^2|s_1^1, 1) = p$, $P(s_1^2|s_1^1, 1) = 1 - p$, $P(s_3^2|s_1^1, 2) = p$, and $P(s_4^2|s_1^1, 2) = 1 - p$. This example is shown in Figure 2 (Right) in Appendix C. Following the result of Lemma 1 if $p \gg (1 - p)$ it can be shown that the variances of the states s_1^2 and s_3^2 have greater importance in calculating the optimal sampling proportions of s_1^1. The calculation is shown in Appendix D. Thus, less likely future states have less importance for computing the optimal sampling proportion in a given state.

Having developed intuition for minimal-variance action selection in a 2-depth tree MDP, we now give our main result that extends Lemma 1 to an L-depth tree.

Theorem 1. *Assume the underlying MDP is an L-depth tree MDP as defined in Definition 1. Let the estimated return of the starting state s_1^1 after n state-action-reward samples be defined as $Y_n(s_1^1)$. Note that the $v^\pi(s_1^1)$ is the expectation of $Y_n(s_1^1)$ under Assumption 1. Let $\mathcal{D}$ be the observed data over n state-action-reward samples. To minimize MSE $\mathbb{E}_{\mathcal{D}}[(Y_n(s_1^1)) - \mu(Y_n(s_1^1)))^2]$ the optimal sampling proportions for any arbitrary state is given by:*

$$b^*(a|s_i^\ell) \propto \sqrt{\pi^2(a|s_i^\ell)\Big[\sigma^2(s_i^\ell, a) + \gamma^2 \sum_{s_j^{\ell+1}} P(s_j^{\ell+1}|s_i^\ell, a) B^2(s_j^{\ell+1})\Big]},$$

where, $B(s_j^\ell)$ is the normalization factor defined as follows:

$$B(s_i^\ell) = \sum_a \sqrt{\pi^2(a|s_i^\ell)\left(\sigma^2(s_i^\ell, a) + \gamma^2 \sum_{s_j^{\ell+1}} P(s_j^{\ell+1}|s_i^\ell, a) B^2(s_j^{\ell+1})\right)} \tag{4}$$

Proof (Overview): We prove Theorem 1 by induction. Lemma 1 proves the base case of estimating the sampling proportion for level $L - 1$ and L. Then, for the induction step, we assume that all the sampling proportions from level L till some arbitrary level $\ell + 1$ can be subsequently built up using dynamic programming starting from level L. For states in level L to the states in level $\ell + 1$ we can compute $b^*(a|s_i^{\ell+1})$ by repeatedly applying Lemma 1. Then we show that at the level ℓ we get a similar recursive sampling proportion as stated in the theorem statement. The proof is given in Appendix E. ■

4.2 MSE OF THE ORACLE

In this subsection, we derive the MSE that the oracle will incur when matching the action proportions given by Theorem 1. The oracle is run for K episodes where each episode consist of L length trajectory of visiting state-action pairs. So the total budget is $n = KL$. At the end of the K-th episode the MSE of the oracle is estimated which is shown in Proposition 2. Before stating the proposition we introduce additional notation which we will use throughout the remainder of the paper. Let

$$T_t^k(s, a) = \sum_{i=0}^{k-1} \mathbb{I}\left\{\left(s_t^i, a_t^i\right) = (s, a)\right\}, \forall t, s, a \tag{5}$$

denote the total number of times that (s, a) has been observed in $\mathcal{D}$ (across all trajectories) up to time t in episode k and $\mathbb{I}\{\cdot\}$ is the indicator function. Similarly let

$$T_t^k(s, a, s') = \sum_{i=0}^{k-1} \mathbb{I}\left\{\left(s_t^i, a_t^i, s_{t+1}^i\right) = (s, a, s')\right\}, \forall t, s, a, s' \tag{6}$$

denote the number of times action a is taken in s to transition to s'. Finally we define the state sample $T_t^k(s) = \sum_a T_t^k(s, a)$ as the total number of times any state is visited and an action is taken in that state.

Proposition 2. *Let there be an oracle which knows the state-action variances and transition probabilities of the L-depth tree MDP* $\mathbf{T}$*. Let the oracle take actions in the proportions given by Theorem 1. Let* $\mathcal{D}$ *be the observed data over* n *state-action-reward samples such that* $n = KL$*. Then the oracle suffers an MSE of*

$$\mathcal{L}_n^* = \sum_{\ell=1}^{L} \Bigg[\frac{B^2(s_i^\ell)}{T_L^{*,K}(s_i^\ell)} + \gamma^2 \sum_a \pi^2(a|s_i^\ell) \sum_{s_j^{\ell+1}} P(s_j^{\ell+1}|s_i^\ell, a) \frac{B^2(s_j^{\ell+1})}{T_L^{*,K}(s_j^{\ell+1})} \Bigg]. \quad (7)$$

where, $T_L^{*,K}(s_i^\ell)$ *denotes the optimal state samples of the oracle at the end of episode* K*.*

The proof is given in Appendix F. From Proposition 2 we see that the MSE of the oracle goes to 0 as the number of episodes $K \to \infty$, and $T_L^{*,K}(s_i^\ell) \to \infty$ simultaneously for all $s_i^\ell \in \mathcal{S}$. Observe that if for every state s the total state counts $T_L^{*,K}(s) = cn$ for some constant $c > 0$ then the loss of the oracle goes to 0 at the rate $O(1/n)$.

4.3 REDUCED VARIANCE SAMPLING

The oracle data collection strategy provides intuition for optimal data collection for minimal-variance policy evaluation, however, it is *not* a practical strategy itself as it requires σ and P to be known. We now introduce a practical data collection algorithm – **Re**duced **Var**iance Sampling (ReVar) – that is agnostic to σ and P. Our algorithm follows the proportions given by Theorem 1 with the true reward variances replaced with an upper confidence bound and the true transition probabilities replaced with empirical frequencies. Formally, we define the desired proportion for action a in state s_i^ℓ after t steps as $\widehat{b}_{t+1}^k(a|s_i^\ell) \propto$

$$\sqrt{\pi^2(a|s_i^\ell)\Big[\widehat{\sigma^u}_t^{(2),k}(s_i^\ell, a) + \gamma^2 \sum_{s_j^{\ell+1}} \widehat{P}_t^k(s_j^{\ell+1}|s_i^\ell, a)\widehat{B}_t^{(2),k}(s_j^{\ell+1})\Big]}, \quad (8)$$

The upper confidence bound on the variance $\sigma^2(s_i^\ell, a)$, denoted by $\widehat{\sigma^u}_{t-1}^{(2),k}(s_i^\ell, a) = (\widehat{\sigma^u}_t^k(s_i^\ell, a))^2$, is defined as:

$$\widehat{\sigma^u}_t^k(s_i^\ell, a) := \widehat{\sigma}_t^k(s_i^\ell, a) + 2c\sqrt{\frac{\log(SAn(n+1)/\delta)}{T_t^k(s_i^\ell, a)}} \quad (9)$$

where, $\widehat{\sigma}_t^k(s_i^\ell, a)$ is the plug-in estimate of the standard deviation $\sigma(s_i^\ell, a)$, $c > 0$ is a constant depending on the boundedness of the rewards to be made explicit later, and $n = KL$ is the total budget of samples. Using an upper confidence bound on the reward standard deviations captures our uncertainty about $\sigma(s_i^\ell, a)$ needed to compute the true optimal proportions. The state transition model is estimated as:

$$\widehat{P}_t^k(s_j^{\ell+1}|s_i^\ell, a) = \frac{T_t^k(s_i^\ell, a, s_j^{\ell+1})}{T_t^k(s_i^\ell, a)} \quad (10)$$

where, $T_t^k(s_i^\ell, a, s_j^{\ell+1})$ is defined in (6). Further in (8), $\widehat{B}_t^k(s_j^{\ell+1})$ is the plug-in estimate of $B(s_j^{\ell+1})$. Observe that for all of these plug-in estimates we use all the past history till time t in episode k to estimate these statistics.

Eq. (8) allows us to estimate the optimal proportion for all actions in any state. To match these proportions, rather than sampling from $\widehat{b}_{t+1}^k(a|s_i^\ell)$, ReVar takes action I_{t+1}^k at time $t+1$ in episode k according to:

$$I_{t+1}^k = \arg\max_a \left\{ \frac{\widehat{b}_t^k(a|s_i^\ell)}{T_t^k(s_i^\ell, a)} \right\}. \quad (11)$$

This action selection rule ensures that the ratio $\widehat{b}_t^k(a|s_i^\ell)/T_t(s_i^L, a) \approx 1$. It is a deterministic action selection rule and thus avoids variance due to simply sampling from the estimated optimal proportions. Note that in the terminal states, s_i^L, the sampling rule becomes

$$I_{t+1}^k = \arg\max_a \left\{ \frac{\pi(a|s_i^L)\widehat{\sigma^u}_t^k(s_i^L, a)}{T_t^k(s_i^L, a)} \right\}$$

which matches the bandit sampling rule of Carpentier and Munos [2011, 2012].

We give pseudocode for ReVar in Algorithm 1. The algorithm proceeds in episodes. In each episode we generate a trajectory from the starting state s_1^1 (root) to one of the terminal state s_j^L (leaf). At episode k and time-step t in some arbitrary state s_i^ℓ the next action I_{t+1} is chosen based on (11). The trajectory generated is added to the dataset $\mathcal{D}$. At the end of the episode we update the model parameters, i.e. we estimate the $\widehat{\sigma}_t^k(s_i^\ell, a)$, and $\widehat{P}_t^k(s_i^{\ell+1}|s_j^\ell, a)$ for each state-action pair. Finally, we update $\widehat{b}_1^{k+1}(a|s_\ell^i)$ for the next episode using eq. (9).

Algorithm 1 Reduced **Var**iance Sampling (ReVar)

1: **Input:** Number of trajectories to collect, K.
2: **Output:** Dataset $\mathcal{D}$.
3: Initialize $\mathcal{D} = \emptyset$, $\widehat{b}_1^0(a|s_i^\ell)$ uniform over all actions in each state.
4: **for** $k \in 0, 1, \ldots, K$ **do**
5: Generate trajectory $H^k := \{S_t, I_t, R(I_t)\}_{t=1}^L$ by selecting I_t according to (11).
6: $\mathcal{D} \leftarrow \mathcal{D} \cup \{(H^k, \widehat{b}_L^k)\}$
7: Update model parameters and estimate $\widehat{b}_1^{k+1}(a|s_i^\ell)$ for each (s_i^ℓ, a).
8: Update $\widehat{b}_1^{k+1}(a|s_i^\ell)$ from level L to 1 following (8).
9: **Return** Dataset $\mathcal{D}$ to evaluate policy π.

4.4 REGRET ANALYSIS

We now theoretically analyze ReVar by bounding its regret with respect to the oracle behavior policy. We analyze ReVar under the assumption that P is known and so we are only concerned with obtaining accurate estimates of the reward means and variances. This assumption is only made for the regret analysis and is *not* a fundamental requirement of ReVar. Though somewhat restrictive, the case of known state transitions is still interesting as it arises in practice when state transitions are deterministic or we can estimate P much easier than we can estimate the reward means.

We first define the notion of regret of an algorithm compared to the oracle MSE $\mathcal{L}_n^*$ in (7) as follows:

$$\mathcal{R}_n = \mathcal{L}_n - \mathcal{L}_n^*$$

where, n is the total budget, and $\mathcal{L}_n$ is the MSE at the end of episode K following the sampling rule in (8). We make the following assumption that rewards are bounded:

Assumption 2. *The reward from any state-action pair has bounded range, i.e., $R_t(s,a) \in [-\eta, \eta]$ almost surely at every time-step t for some fixed $\eta > 0$.*

Note that this is a common assumption in the RL literature [Munos, 2005, Agarwal et al., 2019]. The reward can also be multi-modal as long as it is bounded. Then the regret of ReVar over a L-depth deterministic tree is given by the following theorem.

Theorem 2. *Let the total budget be $n = KL$ and $n \geq 4SA$. Then the total regret in a deterministic L-depth $\mathbf{T}$ at the end of K-th episode when taking actions according to* (8) *is given by*

$$\mathcal{R}_n \leq \widetilde{O}\left(\frac{B_{s_1^1}^2\sqrt{\log(SAn^{11/2})}}{n^{3/2}b_{\min}^{*,3/2}(s_1^1)} + \gamma\sum_{\ell=2}^{L}\max_{s_j^\ell,a}\pi(a|s_1^1)P(s_j^\ell|s_1^1,a)\frac{B_{s_j^\ell}^2\sqrt{\log(SAn^{11/2})}}{n^{3/2}b_{\min}^{*,3/2}(s_j^\ell)}\right)$$

where, the $\widetilde{O}$ hides other lower order terms and $B_{s_i^\ell}$ is defined in (4) *and $b_{\min}^*(s) = \min_a b^*(a|s)$.*

Note that if $L = 1$, $|\mathcal{S}| = 1$, we recover the bandit setting and our regret bound matches the bound in Carpentier and Munos [2011]. Note that MSE using data generated by any policy decays at a rate no faster than $O(n^{-1})$, the parametric rate. The key feature of ReVar is that it converges to the oracle policy. This means that asymptotically, the MSE based on ReVar will match that of the oracle. Theorem 2 shows that the regret scales like $O(n^{-3/2})$ if we have the $b_{\min}^*(s)$ over all states $s \in \mathcal{S}$ as some reasonable constant $O(1)$. In contrast, suppose we sample trajectories from a suboptimal policy, i.e., a policy that produces an MSE worse than that of the oracle for every n. This MSE gap never diminishes, so the regret cannot decrease at a rate faster than the oracle rate of $O(n^{-1})$. Finally, note that the regret bound in Theorem 2 is a problem dependent bound as it involves the parameter $b_{\min}^*(s)$.

Proof (Overview): We decompose the proof into several steps. We define the good event ξ_δ based on the state-action-reward samples $\mathcal{D}$ that holds for all episode k and time t such that $|\widehat{\sigma}_t^k(s,a) - \sigma(s,a)| \leq \epsilon$ for some $\epsilon > 0$ with probability $1-\delta$ made explicit in Corollary 1 . Now observe that MSE of ReVar is

$$\begin{aligned}\mathcal{L}_n &= \mathbb{E}_{\mathcal{D}}\left[\left(Y_n(s_1^1)) - v^\pi(s_1^1))\right)^2\right] \\ &= \mathbb{E}_{\mathcal{D}}\left[\left(Y_n(s_1^1)) - v^\pi(s_1^1))\right)^2 \mathbb{I}\{\xi_\delta\}\right] \\ &\quad + \mathbb{E}_{\mathcal{D}}\left[\left(Y_n(s_1^1)) - v^\pi(s_1^1))\right)^2 \mathbb{I}\left\{\xi_\delta^C\right\}\right] \qquad (12)\end{aligned}$$

Note that here we are considering a known transition function P.The first term in (12) can be bounded using

$$\begin{aligned}&\mathbb{E}_{\mathcal{D}}\left[\left(Y_n(s_1^1)) - v^\pi(s_1^1))\right)^2 \mathbb{I}\{\xi_\delta\}\right] = \mathbf{Var}[Y_n(s_1^1)]\mathbb{E}[T_n^k(s_1^1)] \\ &\leq \sum_a \pi^2(a|s_1^1)\left[\frac{\sigma^2(s_1^1,a)}{\underline{T}_n^{(2),k}(s_1^1,a)}\right]\mathbb{E}[T_n^k(s_1^1,a)] \\ &+ \gamma^2\sum_a \pi^2(a|s_1^1)\sum_{s_j^2} P^2(s_j^2|s_1^1,a) \\ &\quad \cdot \sum_{a'}\pi^2(a'|s_j^2)\left[\frac{\sigma^2(s_j^2,a')}{\underline{T}_n^{(2),k}(s_j^2,a')}\right]\mathbb{E}[T_n^k(s_j^2,a')]\end{aligned}$$

where, $\underline{T}^{(2),k}(s_1^1,a)$ is a lower bound to $T^{(2),k}(s_1^1,a)$ made explicit in Lemma 6, and $\underline{T}^{(2),k}(s_j^2,a)$ is a lower bound to $T^{(2),k}(s_1^1,a)$ made explicit in Lemma 5. We can combine these two lower bounds and give an upper bound to MSE in a two depth $\mathbf{T}$ which is shown Lemma 7. Finally, for the L depth stochastic tree we can repeatedly apply Lemma 7 to bound the first term. For the second term we set the $\delta = n^{-2}$ and use the boundedness assumption in Assumption 2 to get the final bound. The proof is given in Appendix H. ■

5 OPTIMAL DATA COLLECTION BEYOND TREES

The tree-MDP model considered above allows us to develop a foundation for minimal-variance data collection in decision problems where actions at one state affect subsequent states. One limitation of this model is that, for any non-initial state, s_i^ℓ, there is only a single state-action path that could have been taken to reach it. In a more general finite-horizon MDP, there could be many different paths to reach the same non-initial state. Unfortunately, the existence of multiple paths to a state introduces cyclical dependencies

between states that complicate derivation of the minimal-variance data collection strategy and regret analysis. In this section, we elucidate this difficulty by considering the class of directed acyclic graph (DAG) MDPs.

In this section we first define a DAG $\mathcal{G} \subset \mathbf{M}$. An illustrative figure of a 3-depth 2-action $\mathcal{G}$ is in Figure 3 of Appendix I .

Definition 2. (DAG MDP) A DAG MDP follows the same definition as the tree MDP in Definition 1 except $P(s'|s,a)$ can be non-zero for any s in layer ℓ, s' in layer $\ell+1$, and any a, i.e., one can now reach s' through multiple previous state-action pairs.

Proposition 3. *Let $\mathcal{G}$ be a 3-depth, A-action DAG defined in Definition 2. The minimal-MSE sampling proportions $b^*(a|s_1^1), b^*(a|s_j^2)$ depend on themselves such that $b(a|s_1^1) \propto f(1/b(a|s_1^1))$ and $b(a|s_j^2) \propto f(1/b(a|s_j^2))$ where $f(\cdot)$ is a function that hides other dependencies on variances of s and its children.*

The proof technique follows the approach of Lemma 1 but takes into account the multiple paths leading to the same state. The possibility of multiple paths results in the cyclical dependency of the sampling proportions in level 1 and 2. Note that in $\mathbf{T}$ there is a single path to each state and this cyclical dependency does not arise. The full proof is given in Appendix I. Because of this cyclical dependency it is difficult to estimate the optimal sampling proportions in $\mathcal{G}$. However, we can approximate the optimal sampling proportion that ignores the multiple path problem in $\mathcal{G}$ by using the tree formulation in the following way: At every time t during a trajectory τ^k call the Algorithm 1 in Appendix J to estimate $B_0(s)$ where $B_{t'}(s) \in \mathbb{R}^{L\times|\mathcal{S}|}$ stores the expected standard deviation of the state s at iteration t'. After L such iteration we use the value $B_0(s)$ to estimate $b(a|s)$ as follows:

$$b^*(a|s) \propto \sqrt{\pi^2(a|s)\left[\sigma^2(s,a) + \gamma^2 \sum_{s'} P(s'|s,a) B_0^2(s)\right]}.$$

Note that for a terminal state s we have the transition probability $P(s'|s,a) = 0$ and then the $b(a|s) = \pi(a|s)\sigma(s,a)$. This iterative procedure follows from the tree formulation in Theorem 1 and is necessary in $\mathcal{G}$ to take into account the multiple paths to a particular state. Also observe that in Algorithm 1 we use value-iteration for the episodic setting [Sutton and Barto, 2018] to estimate the the optimal sampling proportion iteratively.

6 EMPIRICAL STUDY

We next verify our theoretical findings with simulated policy evaluation tasks in both a tree MDP and a non-tree GridWorld domain. Our experiments are designed to answer the following questions: 1) can ReVar produce policy value estimates with MSE comparable to the oracle solution? and 2) does our novel algorithm lower MSE relative to on-policy sampling of actions? Full implementation details are given in Appendix J.

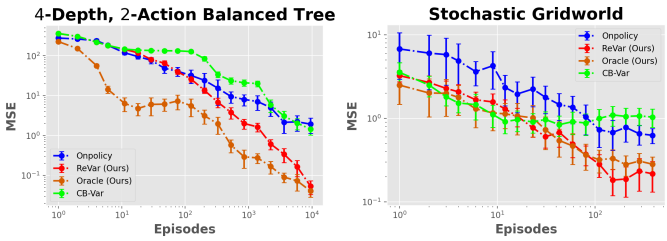

Figure 2: (Left) Deterministic 4-depth Tree. (Right) Stochastic gridworld. The vertical axis gives MSE and the horizontal axis is the number of episodes collected. Axes use a log-scale and confidence bars show one standard error.

Experiment 1 (Tree): In this setting we have a 4-depth 2-action deterministic tree MDP $\mathbf{T}$ consisting of 15 states. Each state has a low variance arm with $\sigma^2(s,1) = 0.01$ and high target probability $\pi(1|s) = 0.95$ and a high variance arm with $\sigma^2(s,1) = 20.0$ and low target probability $\pi(2|s) = 0.05$. Hence, the Onpolicy sampling which samples according to π will sample the second (high variance) arm less and suffer a high MSE. The CB-Var policy is a bandit policy that uses an empirical Bernstein Inequality [Maurer and Pontil, 2009] to sample an action without looking ahead and suffers high MSE. The Oracle has access to the model and variances and performs the best. ReVar lowers MSE comparable to Onpolicy and CB-Var and eventually matches the oracle's MSE.

Experiment 2 (Gridworld): In this setting we have a 4×4 stochastic gridworld consisting of 16 grid cells. Considering the current episode time-step as part of the state, this MDP is a DAG MDP in which there are multiple path to a single state. There is a single starting location at the top-left corner and a single terminal state at the bottom-right corner. Let $\mathbf{L}, \mathbf{R}, \mathbf{D}, \mathbf{U}$ denote the left, right, down and up actions in every state. Then in each state the right and down actions have low variance arms with $\sigma^2(s,\mathbf{R}) = \sigma^2(s,\mathbf{D}) = 0.01$ and high target policy probability $\pi(\mathbf{R}|s) = \pi(\mathbf{D}|s) = 0.45$. The left and top actions have high variance arms with $\sigma^2(s,\mathbf{L}) = \sigma^2(s,\mathbf{U}) = 0.01$ and low target policy probability $\pi(\mathbf{L}|s) = \pi(\mathbf{U}|s) = 0.05$. Hence, Onpolicy which goes right and down with high probability (to reach the terminal state) will sample the low variance arms more and suffer a high MSE. Similar to above, CB-Var fails to look ahead when selecting actions and thus suffers from high MSE. ReVar lowers MSE compared to Onpolicy and CB-Var and actually matches and then reduces MSE compared to the Oracle. We point out that the DAG structure of the Gridworld violates the tree-structure under which Oracle and ReVar were derived. Nevertheless, both methods lower MSE compared to Onpolicy.

7 CONCLUSION AND FUTURE WORKS

This paper has studied the question of how to take actions for minimal-variance policy evaluation of a fixed target policy. We developed a theoretical foundation for data collection in policy evaluation by deriving an oracle data collection policy for the class of finite, tree-structured MDPs. We then introduced a practical algorithm, ReVar, that approximates the oracle strategy by computing an upper confidence bound on the variance of the future cumulative reward at each state and using this bound in place of the true variances in the oracle strategy. We bound the finite-sample regret (excess MSE) of our algorithm relative to the oracle strategy. We also present an empirical study where we show that ReVar decreases the MSE of policy evaluation relative to several baseline data collection strategies including on-policy sampling. In the future, we would like to extend our derivation of optimal data collection strategies and regret analysis of ReVar to a more general class of MDPs, in particular, relaxing the tree structure and also considering infinite-horizon MDPs. Finally, real world problems often require function approximation to deal with large state and action spaces. This setting raises new theoretical and implementation challenges for ReVar where we intend to incorporate experimental design approaches [Pukelsheim, 2006, Mason et al., 2021, Mukherjee et al., 2022]. Another interesting direction is to incorporate structure in the reward distribution of arms Gupta et al. [2021, 2020]. Addressing these challenges is an interesting direction for future work.

Acknowledgements: The authors will like to thank Kevin Jamieson from Allen School of Computer Science & Engineering, University of Washington for pointing out several useful references. This work was partially supported by AFOSR/AFRL grant FA9550-18-1-0166.

References

Alekh Agarwal, Nan Jiang, Sham M Kakade, and Wen Sun. Reinforcement learning: Theory and algorithms. *CS Dept., UW Seattle, Seattle, WA, USA, Tech. Rep*, 2019.

András Antos, Varun Grover, and Csaba Szepesvári. Active learning in multi-armed bandits. In *International Conference on Algorithmic Learning Theory*, pages 287–302. Springer, 2008.

Jean-Yves Audibert, Rémi Munos, and Csaba Szepesvári. Exploration–exploitation tradeoff using variance estimates in multi-armed bandits. *Theoretical Computer Science*, 410(19):1876–1902, 2009.

Andrew G Barto. Intrinsic motivation and reinforcement learning. In *Intrinsically motivated learning in natural and artificial systems*, pages 17–47. Springer, 2013.

Guillaume Bouchard, Théo Trouillon, Julien Perez, and Adrien Gaidon. Online learning to sample. *arXiv preprint arXiv:1506.09016*, 2016.

Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. *arXiv preprint arXiv:1810.12894*, 2018.

Alexandra Carpentier and Rémi Munos. Finite-time analysis of stratified sampling for monte carlo. In *NIPS-Twenty-Fifth Annual Conference on Neural Information Processing Systems*, 2011.

Alexandra Carpentier and Rémi Munos. Minimax number of strata for online stratified sampling given noisy samples. In *International Conference on Algorithmic Learning Theory*, pages 229–244. Springer, 2012.

Alexandra Carpentier, Remi Munos, and András Antos. Adaptive strategy for stratified monte carlo sampling. *J. Mach. Learn. Res.*, 16:2231–2271, 2015.

Kamil Ciosek and Shimon Whiteson. OFFER: Off-environment reinforcement learning. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence (AAAI)*, 2017.

Samarth Gupta, Shreyas Chaudhari, Subhojyoti Mukherjee, Gauri Joshi, and Osman Yağan. A unified approach to translate classical bandit algorithms to the structured bandit setting. *IEEE Journal on Selected Areas in Information Theory*, 1(3):840–853, 2020. doi: 10.1109/JSAIT.2020.3041246.

Samarth Gupta, Shreyas Chaudhari, Subhojyoti Mukherjee, Gauri Joshi, and Osman Yağan. A unified approach to translate classical bandit algorithms to structured bandits. In *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3360–3364, 2021. doi: 10.1109/ICASSP39728.2021.9413628.

Josiah P Hanna, Philip S Thomas, Peter Stone, and Scott Niekum. Data-efficient policy evaluation through behavior policy search. In *International Conference on Machine Learning*, pages 1394–1403. PMLR, 2017.

Nan Jiang and Lihong Li. Doubly robust off-policy value evaluation for reinforcement learning. In *International Conference on Machine Learning*, pages 652–661. PMLR, 2016.

Blake Mason, Romain Camilleri, Subhojyoti Mukherjee, Kevin Jamieson, Robert Nowak, and Lalit Jain. Nearly optimal algorithms for level set estimation. *arXiv preprint arXiv:2111.01768*, 2021.

Andreas Maurer and Massimiliano Pontil. Empirical bernstein bounds and sample variance penalization. *arXiv preprint arXiv:0907.3740*, 2009.

Subhojyoti Mukherjee, KP Naveen, Nandan Sudarsanam, and Balaraman Ravindran. Efficient-ucbv: An almost optimal algorithm using variance estimates. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Subhojyoti Mukherjee, Ardhendu S Tripathy, and Robert Nowak. Chernoff sampling for active testing and extension to active regression. In *International Conference on Artificial Intelligence and Statistics*, pages 7384–7432. PMLR, 2022.

Rémi Munos. Error bounds for approximate value iteration. In *Proceedings of the National Conference on Artificial Intelligence*, volume 20, page 1006. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2005.

James Neufeld, Andras Gyorgy, Csaba Szepesvári, and Dale Schuurmans. Adaptive monte carlo via bandit allocation. In *International Conference on Machine Learning*, pages 1944–1952. PMLR, 2014.

Ian Osband, Charles Blundell, Alexander Pritzel, and Benjamin Van Roy. Deep exploration via bootstrapped dqn. *Advances in neural information processing systems*, 29, 2016.

Brendan O'Donoghue, Ian Osband, Remi Munos, and Volodymyr Mnih. The uncertainty bellman equation and exploration. In *International Conference on Machine Learning*, pages 3836–3845, 2018.

Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *International Conference on Machine Learning*, volume 2017, 2017.

Brahma Pavse, Ishan Durugkar, Josiah Hanna, and Peter Stone. Reducing sampling error in batch temporal difference learning. In *International Conference on Machine Learning*, pages 7543–7552. PMLR, 2020.

Friedrich Pukelsheim. *Optimal design of experiments*. SIAM, 2006.

Reuven Y. Rubinstein and Dirk P. Kroese. *The cross-entropy method: a unified approach to combinatorial optimization, Monte Carlo simulation and machine learning*. Springer Science & Business Media, 2013.

Richard S Sutton. Learning to predict by the methods of temporal differences. *Machine learning*, 3(1):9–44, 1988.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

Mohammad Sadegh Talebi and Odalric-Ambrym Maillard. Learning multiple markov chains via adaptive allocation. *arXiv preprint arXiv:1905.11128*, 2019.

Cameron Voloshin, Hoang M Le, Nan Jiang, and Yisong Yue. Empirical study of off-policy policy evaluation for reinforcement learning. *arXiv preprint arXiv:1911.06854*, 2019.