An Approximate Reinforcement Learning Algorithm for Controlled Diffusion Processes*

Erhan Bayraktar and Ali Devran Kara †

March 16, 2022

Abstract

We study a reinforcement learning algorithm for continuous time stochastic control problems. The proposed algorithm uses the sampled state process by discretizing the state and control action spaces under piece-wise constant control processes. We show that the algorithm converges to the optimality equation of a finite Markov decision process (MDP). Using this MDP model, we provide an upper bound for the approximation error for the optimal value function of the continuous time control problem. Furthermore, we give upper-bounds for the performance loss of the learned control process compared to the optimal admissible control process of the original problem. The provided error upper-bounds are functions of the time and space discretization parameters, and they reveal the effect of different levels of the approximation: (i) approximation of the continuous time control problem by an MDP, (ii) use of piece-wise constant control processes, (iii) space discretization. Finally, we state a time complexity bound for the proposed algorithm as a function of the time and space discretization parameters.

1 Introduction

The goal of this paper is to develop a convergent reinforcement algorithm when the decision maker only has access to the state process and the cost realizations, and establish error bounds for the performance of the learned control process compared to the optimal admissible control as a function of the algorithm parameters.

*E. Bayraktar is partially supported by the National Science Foundation under grant DMS-2106556 and by the Susan M. Smith chair.

†The authors are with the Department of Mathematics, University of Michigan, Ann Arbor, MI, USA, Email: {erhan,alikara}@umich.edu
1.1 Preliminaries

We start with the setup of the paper by defining the dynamics of the control problem. The object is to study a controlled diffusion process, $X(\cdot)$, given by the following stochastic differential equation

$$X(t) = X_0 + \int_0^t b(X(s), u(s)) ds + \int_0^t \sigma(X(s), u(s)) dB(s)$$  \hspace{1cm} (1)

for $t \geq 0$ where $X(t) \in \mathbb{X}$, and $\mathbb{X} \subset \mathbb{R}$. $B(\cdot)$ is the driving noise which is assumed to be a Wiener process, and $u(\cdot) \in \mathbb{U}$ is the control process with measurable paths. We assume that the control is non-anticipative such that for $0 \leq s < t$ the noise increments, $B(t) - B(s)$, are independent of $B(y), u(y)$ for $y \leq s$. We note that the results presented in this paper will be applicable for multidimensional spaces as well, however, we will assume that everything is one-dimensional for ease of notation.

We will later put assumptions on the model, which will guarantee the existence of strong solutions to the diffusion process (1), under admissible control processes (Assumption 2).

The objective of the controller is to minimize the following infinite horizon discounted cost function

$$W_\beta(x_0, u) = E \left[ \int_0^\infty e^{-\beta s} c(X(s), u(s)) ds \right]$$  \hspace{1cm} (2)

where the expectation is with respect to the initial point $X(0) = x_0$ and the given control process $u$, the stage-wise cost function is denoted by $c : \mathbb{X} \times \mathbb{U} \rightarrow \mathbb{R}$, and $\beta$ is some discount factor.

The optimal cost is denoted by

$$W_\beta^*(x_0) := \inf_u W_\beta(x_0, u)$$

where the infimum is taken over all admissible control processes.

1.2 Problem Formulation and a Proposed Algorithm

Our goal is to provide a reinforcement learning algorithm which converges to some approximately optimal control policy under certain assumptions.

We assume that the drift, diffusion, and the cost functions $b, \sigma$ and $c$ are not known. By observing the state process $X(t)$ and the cost realizations $c(X(t), u(t))$, we try to learn a control policy. Furthermore, we keep track of the state and the cost processes via sampling the time and discretizing
and $U$ spaces. To this end, we fix a sampling interval $h > 0$, and finite subsets $X_h \subset X$, $U_h \subset U$. We put the $h$ dependence on the finite spaces to recover the cases where the space discretization rate depends on the time discretization rate. We also define the mapping $\phi_X : X \to X_h$, to map the original value of the state variables to the discrete set $X_h$, e.g. a natural choice would be a nearest neighbour map.

For the exploration phase, we use a randomized piecewise constant control process, such that at $i \times h$, for $i = 0, 1, \ldots$, we choose some $\hat{u} \in U_h$ randomly and apply it for the time interval $[i \times h, (i + 1) \times h)$. Hence, the resulting exploration process $\hat{u}(t)$ is a piecewise constant process. We note that, to make sure the exploration of all possible action values, the random selection of $\hat{u} \in U_h$ should put a positive probability to every element of $U_h$. The distribution on the random selection of the action will clearly affect the speed of the exploration. However, we will not focus on the role of the exploration control on the learning speed. The only assumption we will have on the exploration process for the algorithm to converge is to visit every element of $U_h$ with a positive probability.

After we fix the exploration control process $\hat{u}(t)$, we observe the controlled Markov chain process $X_n$

$$X_n := X(n \times h), \text{ for } n = 0, 1, 2, \ldots$$

Note that $X_n$ takes values from a continuous set $X$. We further map these values to the finite set $X_h$, to construct the stochastic process $\hat{X}_n$, which is defined as

$$\hat{X}_n := \phi_X(X_n)$$

(3)

The process $\hat{X}_n$ is the discretized version of the sampled diffusion process, and hence it takes values from $X_h$. We note that the learning algorithm will be constructed using the process $\{\hat{X}_n\}_n$.

We now present the proposed algorithm formally as follows:

1. Chose a sampling interval $h > 0$.
2. Choose finite subsets $X_h \subset X$ and $U_h \subset U$.
3. Define a mapping $\phi_X : X \to X_h$ (e.g. a nearest neighbour map).
4. Select a $U_h$ valued piecewise constant exploration process $\hat{u}(t)$, which selects an action $\hat{u} \in U_h$ randomly and independent of everything else such that every $\hat{u} \in U_h$ can be chosen with positive probability.

3
5. Observe the discrete process \( \hat{X}_n \) (see (3)), and for all \((\hat{x}, \hat{u}) \in X_h \times U_h\), initiate the Q values by choosing \( Q_0(\hat{x}, \hat{u}) \), choose the learning rates \( \alpha_k(\hat{x}, \hat{u}) \) (see Assumption 3), using the cost realization \( c(X(n \times h), \hat{u}) \) update
\[
Q_{k+1}(\hat{x}, \hat{u}) = (1 - \alpha_k(\hat{x}, \hat{u}))Q_k(\hat{x}, \hat{u}) + \alpha_k(\hat{x}, \hat{u})\left(c(X(k \times h), \hat{u}) \times h + \beta_h \min_{v \in U_h} Q_k(\hat{X}_{k+1}, v)\right)
\]
(4)
where \( \beta_h = e^{-\beta \times h} \), and \( \hat{X}_{k+1} \) is the sampled state we observe following \( \hat{X}_k = \hat{x} \).

6. In Section 4, we will prove that under suitable assumptions, the iterations \( Q_k : X_h \times U_h \rightarrow \mathbb{R} \), converge almost surely to some \( Q^* : X_h \times U_h \rightarrow \mathbb{R} \). In Section 2, we will show that limit values are the Q-values of some finite controlled Markov chain. Using \( Q^* \), we define a mapping \( \gamma_h : X_h \rightarrow U_h \) such that for any \( \hat{x} \in X_h \)
\[
\gamma_h(\hat{x}) = \arg \min_{\hat{u} \in U_h} Q^*(\hat{x}, \hat{u}).
\]

7. Define the control process \( u_h(t) \) such that
\[
u_h(t) = \gamma_h(\phi_X(X(i \times h))) \text{, for } t \in [i \times h, (i + 1) \times h),
\]
that is \( u_h \) is a piece-wise constant process, which changes value at the sampling instances according to the learned map \( \gamma_h \).

We are then interested in the following question:

**Problem:** If we use the learned control \( u_h \) for the continuous time model, what do we loose compared the optimal admissible control process. In other words, we are interested in the difference
\[
W_\beta(x_0, u_h) - W_\beta^*(x_0).
\]
(5)

We will try to bound this term in terms of the sampling interval \( h \) and the state and control action spaces’ discretization errors \( L_X \) and \( L_U \) that are defined as
\[
L_X := \sup_{x \in X} |x - \phi_X(x)|,
\]
\[
L_U := \sup_{u \in U, u_h \in U_h} |u - u_h|.
\]
We will also analyze the asymptotic case, that is, we will show that under certain assumptions

\[ W_\beta(x_0, u_h) - W^*_\beta(x_0) \to 0 \]

as \( h, L_X, L_U \to 0 \).

To answer these questions, we will show that the iterations in (4) converge to the optimal Q values of some finite Markov chain which approximates the value function of the diffusion process in (1) with the cost function (2).

1.3 Literature Review

The results and the techniques used in the paper are related to reinforcement learning algorithms for continuous and discrete time control problems, finite time-space approximations of diffusion processes, and approximation methods for Markov decision processes. In the following, we summarize the related works in the literature by their main objectives:

**Reinforcement learning for stochastic control problems** Optimal control of stochastic (or deterministic) dynamical systems typically requires a perfect knowledge about the system components, however, the correct underlying model of the control problem is usually unknown or too complicated to work with. The objective of reinforcement learning is to estimate or learn the relevant information about the problem such as the value function or the optimal policy by interacting and observing the system. Majority of the literature, however, deals with discrete-time and sequential decision making problems. A popular reinforcement learning algorithm, called Q learning ([33]), for example, is proposed to learn the Q factors, which is closely related to the value function, for discrete time and space problems. Q learning is guaranteed to converge under mild assumptions for discrete settings without any information on the transition models or the cost function of the problem by observing the state and cost realizations ([28, 9]).

Even though, Q learning is simple to implement and requires minimal knowledge about the system, it is not effective for large or continuous state and action spaces. To overcome the dimension challenges, one can try to learn an approximate version of the problem. The approximation can be done through various methods: One can use function approximations for the optimal value function (see [25, 29]). To approximate the value functions, neural networks, state aggregation, or linear approximation techniques with finitely many linearly independent basis functions can be used. Neural network approximation methods lack convergence proofs due to the nature
of the neural networks. For state aggregation and linear approximation methods, convergence can be shown, however, a rigorous error analysis is usually not available. Some related work includes [24, 20, 7, 26, 23] and references therein. For the aforementioned works, typically, either a careful parametrization of the value functions or strong density assumptions on the transition kernels are required. However, for general continuous time problems, these assumptions might be too strong. For example, the sampled continuous time stochastic controlled process we will study in this paper, can only be shown to have weakly continuous transition dynamics (see Lemma 1 and Lemma 2). In [11], it is shown that, by choosing a finite subset of the action space and discretizing the state space, one can consistently learn nearly optimal control policies for systems with weakly continuous transition models.

**Approximations for continuous time control problems** Space approximation methods allow one to use learning algorithms for control problems with large state spaces, however, these methods only work for discrete time decision making problems. Control problems in continuous time, in general, are not feasible to work with numerically. Due to complex nature of the problems, solutions of the control process may not be available, which results in approximation attempts on the time domain, either through the state process or the optimality equation. [18] provides a general Markov chain approximation method for the controlled diffusion processes, by directly approximating the continuous time state process with a finite space controlled Markov chain. It is shown that under certain 'consistency' conditions, state process and value function approximations are asymptotically close to the solutions of the continuous time process. [15, 16] establish convergence rates for such approximation methods. Improved convergence rates for finite approximation methods are later presented in [4, 3]. We note that, these works study the approximation of an available model and do not focus on learning an approximate model when the dynamics are not available to the decision maker.

**Reinforcement learning in continuous time** Learning and planning in continuous time problems becomes much more challenging in continuous mainly because if the complex dynamics and optimality equation of the problems. To overcome these issues, common approaches include: focusing on simpler models such as deterministic dynamics or linear Gaussian dynamics with quadratic cost functions (LQG problems), or discretizing time and space to work with a finite Markov decision process which approximates the original continuous problem.

Working in deterministic or LQG settings allows one to analyze the op-
timality equation in a rigorous way, and to develop learning algorithms that work directly in continuous time. [31, 32] study exploration and exploitation trade-off in LQG setup through entropy regularization. [5] proposes various TD-learning type algorithms to learn the value function in continuous time for deterministic dynamics. [14] develops a Q learning algorithm that works directly in continuous time for deterministic dynamics with Lipschitz continuous controls.

Another common approach for continuous time reinforcement learning is to discretize the time, and state and action spaces. [21] proposes a method building on finite difference and finite element approximations building on [18] for a general controlled diffusion process, and proves that the learned approximations are asymptotically optimal as the discretization parameters converge to 0. Similarly, [22] studies learning of finite approximations of continuous time problems for deterministic dynamics, here, different from the previous works, the focus is on the non-uniform quantization of the state space for better and faster learning. [27, 2] study continuous time reinforcement learning through “advantage functions” for deterministic and LQG settings by arguing that as the time discretization gets finer, Q values become less informative to evaluate the performance of the actions. We will show later in this paper that, in order to make the Q estimates informative, one needs to keep the space discretization rate at least as high as the time discretization rate. Although, we only study Q function estimates in this paper, same condition also guarantees the advantage function estimates in [27, 2] to admit a limit.

Our contributions In this paper, we will study an approximate Q learning algorithm for general continuous time stochastic control problems by discretizing the time, and state and action spaces. The discretization in time will assumed to be uniform, however, state and action spaces can be discretized in a non-uniform way. We will show that the algorithm will converge under mild assumptions, even when the space quantization is non-uniform. Different from earlier works, we will then provide error bounds and convergence rates in terms of the discretization parameters. The bounds will suggest that, even though the algorithm converges for general space quantization, the performance of the learned value functions and policies will depend on the quantization scheme. Furthermore, provided error bounds will emphasize the effects of different levels of the approximation such as estimating the diffusion process with a controlled Markov chain, use of piece-wise constant policies, and state and action space quantization. Lastly, we will also discuss the effect of discretization parameters on the learning speed of the algorithm.
1.4 Outline of the Paper

In Section 2, we construct a finite space Markov decision process (MDP) that will serve to approximate the diffusion process (1). In particular, in Section 2.1, we present a controlled discrete time control process, which has the same distributions as the diffusion process at the sampling instances, when the diffusion process is controlled using piece-wise constant control functions. In Section 2.2, we provide a finite space controlled Markov chain, that is constructed based on the MDP from Section 2.1, using state aggregation methods.

In Section 3, we analyze the differences between the value function of the diffusion process (1), and the value function of the finite space MDP constructed in Section 2.2. Furthermore, we provide upper bounds for the error (or regret) of the control policy designed for the finite space MDP, when it is used to control the diffusion process, in terms of the approximation parameters; where the comparison is with respect to the performance of the optimal admissible control process.

In Section 4, we show that the algorithm (4) converges, under certain assumptions, to the Q values of the finite space MDP constructed in Section 2.2, and thus, we provide error bounds for the learned policy using the results from Section 3.

Finally, in Section 5, towards a practical purpose, we analyze the convergence rate of the algorithm (4), with respect to the sampling interval lengths, $h$, and the number of iterations where we provide time complexity bounds.

2 Approximate Markov Chains

In this section, we provide two controlled Markov chains which will help us to analyze the error term (5).

2.1 A Markov Chain Construction with Exact Approximation of the Diffusion Process

The first Markov chain we will present will have the same finite dimensional distributions with the sampled diffusion process under piece-wise constant control processes.

Let $X$ be the state space, and $U_h$ be the control action space of the Markov chain.
We define the transition probabilities as follows: For any $k \in \mathbb{Z}_+$, distribution of the state $X_k$ conditioned on the past state and action variables, is determined by the diffusion process (1), such that, conditioned on $(x_{k-1}, \ldots, x_0, u_{k-1}, \ldots, u_0)$, $X_k$ has the same distribution as

$$X(h) = x(0) + \int_0^h b(X(s), u(s))ds + \int_0^h \sigma(X(s), u(s))dB(s)$$

(6)

where $x(0) = x_{k-1}$ and $u(s) = u_{k-1}$ for all $0 \leq s \leq h$. Hence, for any $A \in \mathcal{B}(X)$

$$Pr(X_k \in A|x_{[0,k-1]}, u_{[0,k-1]}) = \mathcal{T}_h(A|x_{k-1}, u_{k-1})$$

where $x, u_{[0,k-1]} := x_0, \ldots, x_{k-1}, u_0, \ldots, u_{k-1}$, such that

$$\mathcal{T}_h(dx_k|x_{k-1}, u_{k-1}) \sim X(h)$$

where $X(h)$ determined by (6) and where $\mathcal{T}_h$ is the transition kernel of the Markov chain which is a stochastic kernel from $X \times U_h$ to $X$.

We also define a stage-wise cost function $c_h: X \times U_h \rightarrow \mathbb{R}_+$ such that for any $(x, u) \in X \times U_h$

$$c_h(x, u) := c(x, u) \times h$$

where $c$ is the cost function of the diffusion process (see (2)).

We now define the infinite horizon discounted cost function

$$J_{\beta_h}(x_0, \gamma) := E_{x_0}^{\mathcal{T}_h, \gamma} \left[ \sum_{k=0}^{\infty} \beta_h^k c_h(X_k, U_k) \right]$$

(7)

where $\beta_h := e^{-\beta \times h}$, and $\gamma$ is an admissible policy. An admissible policy is a sequence of control functions $\{\gamma_k, k \in \mathbb{Z}_+\}$ such that $\gamma_k$ is measurable with respect to the $\sigma$-algebra generated by the information variables $I_k = \{X_{[0,k)}, U_{[0,k-1]}\}$, $k \in \mathbb{N}$, $I_0 = \{Y_0\}$, where

$$U_k = \gamma_k(I_k), \quad k \in \mathbb{Z}_+,$$

are the $U_h$-valued control actions. We define $\Gamma$ to be the set of all such admissible policies. The optimal cost function is defined as

$$J_{\beta_h}^*(x_0) := J_{\beta_h}(x_0, \gamma).$$

(8)
Remark 1. An important property of the MDP model we will make use of, is the following one: suppose that we are given an admissible policy \( \gamma \in \Gamma \) defined for the MDP, we define the following control process \( u(t) \) such that
\[
u(t) = \gamma(X(k \times h)), \text{ for } t \in [k \times h, (k + 1) \times h)
\]
which is a piece-wise constant control process. Then, the controlled Markov chain state process \( x_k \) under the policy \( \gamma \) and the controlled diffusion process \( X(t) \) under the control process defined in (9) have the same distributions at the sampling instances if they start from the same initial points, that is, for any \( k \in \mathbb{Z}_+ \),
\[X_k \sim X(k \times h)\].

2.2 Finite State MDP Construction by Discretization of the State Space

We now construct an MDP with a finite state space by discretizing the state space \( \mathbb{X} \).

We start by choosing a collection of disjoint sets \( \{B_i\}_{i=1}^M \) such that \( \bigcup_i B_i = \mathbb{X} \), and \( B_i \cap B_j = \emptyset \) for any \( i \neq j \). Furthermore, we choose a representative state, \( \hat{x}_i \in B_i \), for each disjoint set. For this setting, we denote the new finite state space by \( \mathbb{X}_h := \{\hat{x}_1, \ldots, \hat{x}_M\} \). We put the dependence on the parameter \( h \), since we will let the size of the finite set, \( M \) go to \( \infty \) as \( h \to 0 \). The mapping from the original state space \( \mathbb{X} \) to the finite set \( \mathbb{X}_h \) is done via
\[
\phi_\mathbb{X}(x) = \hat{x}_i \quad \text{if } x \in B_i.
\]
Furthermore, we choose a weighing measure \( \pi^*(\cdot) \in \mathcal{P}(\mathbb{X}) \) on \( \mathbb{X} \) such that \( \pi^*(B_i) > 0 \) for all \( i \in \{1, \ldots, M\} \). We now define normalized measures using the weight measure on each separate quantization bin \( B_i \) such that
\[
\hat{\pi}^*_x(A) := \frac{\pi^*(A)}{\pi^*(B_i)}, \quad \forall A \subset B_i, \quad \forall i \in \{1, \ldots, M\}
\]
that is \( \hat{\pi}^*_x \) is the normalized weight measure on the set \( B_i \), \( \hat{x}_i \) belongs to.

We now define the cost and transition kernels for this finite set using the normalized weight measures such that for any \( \hat{x}_i, \hat{x}_j \in \mathbb{X}_h \)
\[
C^*_h(\hat{x}_i, u) = \int_{B_i} c_h(x, u) \hat{\pi}^*_x(dx)
\]
\[
P^*_h(\hat{x}_j|\hat{x}_i, u) = \int_{B_i} T_h(B_j|x, u) \hat{\pi}^*_x(dx).
\]
where $T_h$ is the transition model for the MDP constructed in Section 2.1.

Having defined the finite state space $X_h$, the cost function $C_h^*$ and the transition model $P_h^*$, we can now introduce the optimal value function for this finite model. We denote the optimal cost function, which is defined on $X_h$, by $\hat{J}_{\beta_h}: X_h \rightarrow \mathbb{R}$. Note that $\hat{J}_{\beta_h}$ satisfies the following Bellman equation for any $\hat{x}_i \in X_h$

$$\hat{J}_{\beta_h}(\hat{x}_i) = \inf_{u \in U_h} \left\{ C_h^*(\hat{x}_i, u) + \beta_h \sum_{\hat{x}_1 \in X_h} \hat{J}_{\beta_h}(\hat{x}_1) P_h^*(\hat{x}_1|\hat{x}_i, u) \right\}$$  \hspace{1cm} (13)

We can easily extend this function over the state space $X$ by making it constant over the quantization bins. In other words, if $\hat{x}_i \in B_i$, where $\bigcup_i B_i = X$, for any $x \in B_i$, we write

$$\hat{J}_{\beta_h}(x) := \hat{J}_{\beta_h}(\hat{x}_i).$$

Furthermore, the following fixed point equation can be written for the finite MDP:

$$Q_h^*(\hat{x}_i, u) = C_h^*(\hat{x}_i, u) + \beta_h \sum_{\hat{x}_1} \min_v Q_h^*(\hat{x}_1, v) P_h^*(\hat{x}_1|\hat{x}_i, u).$$

**Remark 2.** We will prove that the iterations (4) converge to some $Q^*$ which satisfies the above fixed point equation, under suitable conditions on the diffusion process.

We further define uniform error bounds resulting from the discretization of the state and actions spaces such that

$$L_X := \sup_{x \in X} |x - \phi_X(x)|,$$

$$L_U := \sup_{u \in U} \min_{u_h \in U_h} |u - u_h|.$$  \hspace{1cm} (11)

The following result gives an upper bound for the performance loss resulting from applying the policy designed for the finite space MDP on the MDP with continuous state and action spaces.

For the result, we consider a general controlled Markov chain with transition kernel $T$, stage-wise cost function $c$, and a discount factor $\beta$.

**Assumption 1.** There exists $\alpha_{c}^{X}, \alpha_{c}^{U}, \alpha_{T}^{X}, \alpha_{T}^{U} < \infty$ such that

- $|c(x, u) - c(x', u)| \leq \alpha_{c}^{X}|x - x'|$, 
- $|c(x, u) - c(x', u)| \leq \alpha_{c}^{U}|u - u'|$, 
- $|T(x, u) - T(x', u)| \leq \alpha_{T}^{X}|x - x'|$, 
- $|T(x, u) - T(x', u)| \leq \alpha_{T}^{U}|u - u'|$. 

11
\[ |c(x,u) - c(x,u')| \leq \alpha_c^u |u-u'|, \]
\[ W_1(\mathcal{T}(\cdot|x,u), \mathcal{T}(\cdot|x',u)) \leq \alpha_T^x |x-x'|, \]
\[ W_1(\mathcal{T}(\cdot|x,u), \mathcal{T}(\cdot|x,u')) \leq \alpha_T^u |u-u'|, \]

where \( W_1 \) denotes the first order Wasserstein distance.

**Proposition 1** ([11]). Let \( \hat{\gamma} \) denote the optimal policy designed for the finite controlled Markov chain, under Assumption 1, if \( \beta \alpha_T^X < 1 \) we then have

\[
\sup_{x_0 \in \mathcal{X}} |J_\beta(x_0, \hat{\gamma}) - J_\beta^*(x_0)| \leq \frac{\alpha_c^X - \beta \alpha_T^X \alpha_c^U + \beta \alpha_T^X \alpha_T^U}{(1-\beta)(1-\beta \alpha_T^X)} L_X + \frac{\alpha_c^X}{(1-\beta)(1-\beta \alpha_T^X)} L_X,
\]

in particular, if we choose \( \alpha_T := \max(\alpha_T^X, \alpha_T^U) \), and \( \alpha_c := \max(\alpha_c^X, \alpha_c^U) \), we can write

\[
\sup_{x_0 \in \mathcal{X}} |J_\beta(x_0, \hat{\gamma}) - J_\beta^*(x_0)| \leq \frac{\alpha_c}{(1-\beta)(1-\beta \alpha_T)} (L_X + L_U).
\]

### 3 Approximation of the Diffusion Process by a Finite MDP

Recall that we are interested in the term

\[ W_\beta(x_0, u_h) - W_\beta^*(x_0) \]

where \( u_h \) the control process obtained with \( \gamma_h \) that is learned via (4). The first term represents the cost induced by the application of the control designed for the finite space MDP, when it is used for the continuous time model. The second term represents the optimal cost for the continuous time model. In Section 4, we will prove that learned policy \( \gamma_h \) is optimal for the MDP defined in Section 2.2. Hence, in this section, we will assume that \( \gamma_h \) solves (13).

We write the following

\[
W_\beta(x_0, u_h) - W_\beta^*(x_0) = W_\beta(x_0, u_h) - J_{\beta_h}(x_0, \gamma_h) + J_{\beta_h}(x_0, \gamma_h) - J_{\beta_h}^*(x_0) + J_{\beta_h}^*(x_0) - W_\beta^*(x_0).
\]

In what follows, we will analyze each term separately.
3.1  Analysis of Term (14)

The following result provides a bound for the difference between the value function of a diffusion process controlled with piece-wise constant control processes and the value function of an MDP controlled with an admissible policy.

Assumption 2. For the diffusion process given in (1), we assume that

- \( \sup_{x,u} |b(x,u)| \leq B \), and \( \sup_{x} |\sigma(x,u)| \leq B \), for some \( B < \infty \).

- \( |b(x,u) - b(x',u')| + |\sigma(x,u) - \sigma(x',u')| \leq K (|x - x'| + |u - u'|) \) for some \( K < \infty \).

- \( \sup_{x,u} |c(x,u)| \leq C \) for some \( C < \infty \).

- \( |c(x,u) - c(x',u')| \leq K (|x - x'| + |u - u'|) \) for some \( K < \infty \).

Note that these assumptions are sufficient for the existence of a unique strong solution to (1) under admissible control processes.

Proposition 2. Let \( \gamma \in \Gamma \) be an admissible policy for the sampled controlled Markov chain, and \( u_h(\cdot) \) be the corresponding piece-wise constant control process for the diffusion process such that

\[ u_h(t) = \gamma(x(i \times h)), \text{ for } t \in [i \times h, (i + 1) \times h]. \]

Under Assumption 2, we have that for any \( x_0 \in X \)

\[ |W_\beta(x_0, u_h) - J_\beta(x_0, \gamma)| \leq hB + \frac{KBh}{1 - e^{-\beta h}} \left( h + \sqrt{\frac{2h}{\pi}} \right). \]

Proof. The proof can be found in Appendix A.

3.2  Analysis of Term (15)

For the analysis of (15), we will use Theorem 1. However, in order to use Theorem 1, we need to calculate the Lipschitz constants of the sampled controlled Markov chain in terms of the Lipschitz constants of the diffusion process introduced in Assumption 2.

In what follows, we will focus on the controlled Markov chain constructed in Section 2.1. Recall that we have

\[ T_h(dx_k|x_{k-1}, u_{k-1}) \sim X(h) \]
where
\[ X(h) = X(0) + \int_0^h b(X(s), u(s))ds + \int_0^h \sigma(X(s), u(s))dB(s) \]
with \( x(0) = x_{k-1} \) and \( u(s) = u_{k-1} \) for all \( 0 \leq s \leq h \).
Furthermore,
\[ \beta_h := e^{-\beta h} \]
\[ c_h(x, u) := c(x, u) \times h. \]

**Lemma 1.** Under Assumption 2,
\[ W_1(\mathcal{T}_h(\cdot|y, u), \mathcal{T}_h(\cdot|x, u)) \leq |x - y|e^{(K + \frac{K^2}{2})h}. \]
where \( W_1 \) denotes the first order Wasserstein distance.

*Proof.* The proof can be found in Appendix B.

**Lemma 2.** Under Assumption 2, if \( h < 1 \)
\[ W_1(\mathcal{T}_h(\cdot|x, u), \mathcal{T}_h(\cdot|x, \hat{u})) \leq |u - \hat{u}|2Ke^{2K^2h}. \]
where \( W_1 \) denotes the first order Wasserstein distance.

*Proof.* The proof can be found in Appendix C.

**Proposition 3.** Under Assumption 2, if \( \beta > K + \frac{K^2}{2} \)
\[ \sup_{x_0 \in X} \left| J_{\beta_h}(x_0, \hat{\gamma}_h) - J_{\beta_h}^*(x_0) \right| \leq \frac{Kh - Khe^{h(K + \frac{K^2}{2} - \beta) + 2K^2he^{h(2K^2 - \beta)}}}{(1 - e^{-\beta h})(1 - e^{h(K + \frac{K^2}{2} - \beta)})}L_U + \frac{Kh}{(1 - e^{-\beta h})(1 - e^{h(K + \frac{K^2}{2} - \beta)})}L_X \]

*Proof.* The proof follows directly from Proposition 1, Lemma 1 and Lemma 2, by noting that \( \alpha^X_c, \alpha^U_c \leq hK \).

### 3.3 Analysis of Term (16)

Recall that (16) deals with the optimal value function of the sampled controlled Markov chain and the optimal value function of the controlled diffusion process. Hence, we make use of finite difference approximation methods for Bellman equations. The following result, taken from [17, 10], gives an upper bound on the performance loss of the piece-wise constant policies applied for the diffusion processes.
Lemma 3 ([17, 10]). Let $u^*_h$ denote the piece-wise constant control process, which is constant over the intervals $[nh, n(h+1))$, that achieves the minimum cost for (2). Under Assumption 2, we have

$$W_\beta(x_0, u^*_h) - W^*_\beta(x_0) \leq Nh^{1/4}$$

for some constant $N < \infty$, which only depends on the discount factor $\beta$ and the Lipschitz coefficients of the model.

Remark 3. One might expect that if the dynamics of the diffusion process are changing slowly, and the cost function does not have rapid changes with respect to the state process, then the time discretization leads to smaller performance losses. Indeed, as it is shown in [10, Proposition 2.4], if $W_\beta$ and $c$ are regular enough, namely if

$$\sup_u \left( \|Lu(LuW_\beta)\|_\infty + \|Lu c\|_\infty \right) < \infty$$

where $Lu$ is the generator function of the diffusion process for some control action $u$, then one might have

$$W_\beta(x_0, u^*_h) - W^*_\beta(x_0) \leq Nh.$$  

We are now ready to analyze (16).

Proposition 4. Under Assumption 2

$$J^*_{\beta_h}(x_0) - W^*_\beta(x_0) \leq hB + \frac{KBh}{1 - e^{-\beta h}} \left( h + \sqrt{\frac{2h}{\pi}} \right) + Nh^{1/4}.$$  

Proof. We start by the following bound

$$J^*_{\beta_h}(x_0) - W^*_\beta(x_0) \leq J^*_{\beta_h}(x_0) - W_\beta(x_0, u^*_h) + W_\beta(x_0, u^*_h) - W^*_\beta(x_0)$$

$$\leq J^*_{\beta_h}(x_0) - W_\beta(x_0, u^*_h) + Nh^{1/4}$$

where the last bound follows directly from Theorem 3. For the first term, let $\gamma^*_h$ denote the corresponding control policy for the controlled Markov chain (that has the same law at the sampling instances as $u^*_h$). Note that this policy is not necessarily optimal for the sampled controlled Markov chain. Hence, we have the following bound:

$$J^*_{\beta_h}(x_0) - W_\beta(x_0, u^*_h) \leq J_{\beta_h}(x_0, \gamma^*_h) - W_\beta(x_0, u^*_h)$$

$$\leq hB + \frac{KBh}{1 - e^{-\beta h}} \left( h + \sqrt{\frac{2h}{\pi}} \right).$$
The last bound follows from Theorem 2.

Combining what we have so far, we can conclude that

\[ J_{\beta_h}^*(x_0) - W_{\beta}^*(x_0) \leq hB + \frac{KBh}{1 - e^{-\beta h}} \left( h + \sqrt{\frac{2h}{\pi}} \right) + Nh^\frac{1}{4}. \]

\[ \square \]

3.4 Near Optimality of the Approximate Control

Combining the results, we have presented so far, we can now state the main theorem of this section:

**Theorem 1.** Under Assumption 2, we have

\[ W_{\beta}(x_0, u_h) - W_{\beta}^*(x_0) \leq e(h, L_X, L_U) \]

where

\[ e(h, L_X, L_U) := 2 \left( hB + \frac{KBh}{1 - e^{-\beta h}} \left( h + \sqrt{\frac{2h}{\pi}} \right) \right) \]

\[ + \frac{Kh - Khe^{h(K+\frac{K^2}{2}-\beta)} + 2K^2he^{h(2K^2-\beta)}}{(1 - e^{-\beta h})(1 - e^{h(K+\frac{K^2}{2}-\beta)})} L_U \]

\[ + \frac{Kh}{(1 - e^{-\beta h})(1 - e^{h(K+\frac{K^2}{2}-\beta)})} L_X \]

\[ + Nh^\frac{1}{4}. \]  

(17)

where \( u_h \) the control process obtained with \( \gamma_h \), which is optimal for the finite MDP model constructed in Section 2.2.

Note that, as stated in Remark 3, the last term in Theorem 1 can be replaced by \( Nh \), if the dynamics and the cost function are regular enough, e.g. if they do not change rapidly with time.

**Corollary 1.** For small \( h \), the upper bound derived in Theorem 1, behaves as

\[ C \left( \sqrt{h} + \frac{L_X + L_U}{h} + h^\frac{1}{4} \right) \]

for some \( C < \infty \). This representation makes the distinction between the effects of different steps of the approximation. The first term results from
the Markov chain approximation of the diffusion process, the second term is due to the state and action space discretization, and finally the last term is due to the piece-wise constant control processes. We note again that the last term may be replaced with $h$ if the dynamics are regular enough.

4 Convergence of the Learning Algorithm and Near Optimality of Learned Policies

In this section we present the main results of the paper.

We first show that the iterations in (4) converge to the optimal Q values of the approximate controlled Markov chain constructed in Section 2.2. Recall that the Q value iterations are in the following form:

$$Q_{k+1}(\hat{x}, \hat{u}) = (1 - \alpha_k(\hat{x}, \hat{u}))Q_k(\hat{x}, \hat{u}) + \alpha_k(\hat{x}, \hat{u})\left(c(X(k \times h), \hat{u}) \times h + \beta_h \min_{v \in U_h} Q_k(\hat{X}_{k+1}, v)\right)$$  \hspace{1cm} (18)

where $\beta_h = e^{-\beta \times h}$, and $\hat{X}_{k+1}$ is the discretized sampled state we observe following $\phi_X(x(i \times h)) = \hat{x}$. Furthermore, $\phi_X$ maps the original state space $X$ to the finite subset $X_h$.

**Assumption 3.**

i. $\alpha_t(\hat{x}, \hat{u}) = 0$ unless $(\hat{X}_t, u_t) = (\hat{x}, \hat{u})$. Furthermore,

$$\alpha_t(\hat{x}, \hat{u}) = \frac{1}{1 + \sum_{k=0}^{t-1} 1_{\{\hat{X}_k = \hat{x}, u_k = \hat{u}\}}}$$

We note that, this means that $\alpha_k(\hat{x}, \hat{u}) = \frac{1}{k}$ if $\hat{X}_k = \hat{x}, u_k = \hat{u}$, if $k$ is the instant of the $k$th visit to $(\hat{x}, \hat{u})$, as this will be crucial in the averaging of the Markov chain dynamics

ii. The controlled diffusion process converges to its unique invariant measure under stationary control processes. In particular, under the exploration policy, which is assumed to be random and stationary, the diffusion process $X(t)$ converges to its invariant measure.

iii. Every $\hat{x} \in X_h$ and $\hat{u} \in U_h$ is visited infinitely often during exploration.

**Remark 4.** For the stability assumption (ii), testable conditions can be found in the literature (see e.g. [8]), however, we do not present them here as it is beyond the purpose of this paper.
The third assumption (iii) is a usual requirement for reinforcement learning algorithms, and it is a realistic assumption, since the driving noise is assumed to be a Wiener process and the exploration control actions are chosen randomly such that any action can be chosen with positive probability.

**Proposition 5.** Under Assumption 3, the iterations (18) converge almost surely to some $Q^*_h : X_h \times U_h \to \mathbb{R}$, which satisfies for every $\hat{x}, \hat{u} \in X_h \times U_h$

$$Q^*_h(\hat{x}, \hat{u}) = C^*_h(\hat{x}, \hat{u}) + \beta_h \sum_{\hat{x}_1 \in X_h} \min_{\hat{u}_1 \in U_h} Q^*_h(\hat{x}_1, \hat{u}_1) P^*_h(\hat{x}_1|\hat{x}, \hat{u})$$

where $C^*_h$ and $P^*_h$ are defined in (12)

**Proof.** The proof can be found in Appendix D. \qed

Once, $Q^*_h$ is obtained, one can construct the policies such that

$$\gamma_h(\hat{x}) = \arg\min_{\hat{u}} Q^*_h(\hat{x}, \hat{u})$$

Using these policies, following control processes are defined

$$u_h(t) = \gamma_h(\phi_X(X(i \times h)), t \in [i \times h, (i+1) \times h). \ (19)$$

Hence, $u_h$ is a piece-wise constant process, which changes value at the sampling instances according to the learned map $\gamma_h$.

The following result is a direct implication of Theorem 1, and Theorem 5:

**Theorem 2.** Under Assumption 2, and Assumption 3, iterations in (18) converge to some $Q^*_h$. For the learned control process $u_h$ (see (19)), we have

$$W_\beta(x_0, u_h) - W_\beta^*(x_0) \leq e(h, L_X, L_U)$$

where $e(h, L_X, L_U)$ is defined in (17), and where $u_h$ is the control policy learned by using the approximate Q-learning algorithm (4).

Furthermore, for small $h$, we have that

$$e(h, L_X, L_U) \leq C \left( \sqrt{h} + \frac{L_X + L_U}{h} + h^{1/4} \right)$$

for some $C < \infty$.

We now present results for the asymptotic case.

The first one states that if we first increase the quantization rate and thus if the quantization error goes to 0, then if we take the sampling interval to 0; the error bound goes to 0.
Corollary 2. Under Assumption 2,

\[
\lim_{h \to 0} \lim_{L_X, L_U \to 0} (W_\beta(x_0, u_h) - W_\beta^*(x_0)) = 0.
\]

We now define the quantization rate and the quantization error as a function of \( h \) and denote them by \( L_X(h), L_U(h) \):

Corollary 3. Under Assumption 2,

\[
\lim_{h \to 0} (W_\beta(x_0, u_h) - W_\beta^*(x_0)) = 0
\]

if \( L_X(h), L_U(h) \) go to 0 at a faster rate than \( h \), i.e. if \( \frac{L_X(h)}{h}, \frac{L_U(h)}{h} \to 0 \) as \( h \to 0 \).

5 A Discussion on the Convergence Rate of the Learning and the Effect of Sampling and Quantization Rates on the Learning Speed

For approximation accuracy, finer sampling intervals and higher quantization rates lead to smaller error bounds. However, it is clear that finer sampling intervals and higher quantization rates also result in slower learning. In particular, higher quantization rate of the state and action spaces results in larger aggregate state and action spaces which in turn leads to dimension issues for the learning. Furthermore, finer sampling intervals increases the discount factor \( \beta_h \) of the approximate Q learning algorithm in (18), and higher discount rates make the iterations in (18) to converge at a slower rate by increasing the effective horizon of the problem.

First, we note that the algorithm presented in this paper relies on the convergence of the state process to its stationary distribution, since the state aggregation results in non-Markovian dynamics. Hence, the convergence speed of the algorithm depends on the convergence to the invariant measure of the process. Our motivation in this section is to study the effect of state and time discretization parameters on the speed of the Q learning algorithm. However, these discretization parameters do not affect the speed of the convergence to the stationary distribution, as this is related to the dynamics of the underlying diffusion process. Thus, for a simpler presentation, we will assume the state process starts from its stationary distribution under the exploration policy and always stays there during the exploration.

The following well known result ([6, Theorem 5]) is stated using the notation of this paper. The result provides a sample complexity bound for
the near optimal Q estimates when the learning rate is linear, i.e. \( \alpha_k = \frac{1}{k} \), as in this paper.

**Proposition 6 ([6])**. Let \( Q_T \) be the value of the Q-learning algorithm using linear learning rate at time \( T \). Then with probability at least \( 1 - \delta \), for any positive constant \( \psi \) we have \( \| Q_T - Q^*_h \| \leq \epsilon \), given that

\[
T = \Omega \left( (L + \psi L + 1) \frac{2^{1-\beta_h} \ln \frac{V_{\text{max}}}{\epsilon}}{\ln \frac{|X_h| |U_h| |V_{\text{max}}|}{\delta \psi (1 - \beta_h)} (\psi \epsilon (1 - \beta_h))^2} \right),
\]

where \( V_{\text{max}} := \| \hat{J}_{\beta_h} \|_{\infty} \leq \frac{h \| c \|_{\infty}}{1 - \beta_h} \), and \( L \) is the covering time for the algorithm, that is the smallest time for every state and action pair to be visited at least once.

The above result can be simplified for small enough \( h \). We can write that

\[
T \lesssim \left( \frac{1}{\epsilon} \right) \frac{\ln (L + \psi L + 1)}{h} \frac{\ln (\frac{|X_h| |U_h|}{\delta \epsilon h})}{\epsilon^2 h^2},
\]

where we use \( \lesssim \) as we drop some constant and logarithmic dependence.

From (21), we observe that decreasing the time discretization parameter \( h \), increases the sample complexity exponentially. Note that, as the required sample size increases exponentially, the duration we need to observe the diffusion process in real time to get \( \epsilon \)-near estimates also increases, since the exponential increase in the sample complexity dominates the decrease rate on \( h \).

For the effect of the space discretization, we can see that the sample complexity increases in a logarithmic way with the second term, but the dominant effect is caused by the increase on the cover time \( L \), as we have that \( L \geq |X_h| |U_h| \), and depending on the sampling frequency of the samples, the cover time can be even greater. Nonetheless, the cover time increases at least linearly with the increase on the size of the aggregate state and action spaces, which in turn increases the sample complexity at a polynomial rate depending on the parameter \( h \).

Figure 1 shows the change on the required time when for \( h \in [0.7, 0.9] \), \( \epsilon \in [0.1, 0.5] \) and when \( |X_h| \) and \( |U_h| \) are assumed to be order of \( \frac{1}{h} \).

However, this is the sample complexity to achieve \( \epsilon \)-near estimates of \( Q^*_h \) which is also an estimate of the true value function (see Theorem 2 and Corollary 1). Hence, one needs to pick the sampling parameter \( h \) in
Figure 1: Sample complexity for different values of error and sampling parameters

a careful way considering the trade-off between the convergence speed and the approximation accuracy. Following Figure 2 provides an example for the error bounds given a given level sample points $T \sim 9500$. The graph on the left shows the difference between $Q_T$ and $Q_h^*$ (see (18)), clearly the error for the learning of the approximate model decays as $h$ increases for a fixed level of sample points since the learned model becomes simpler as $h$ increases. The graph on the right shows represents the difference

$$|Q_T - Q^*| \leq |Q_T - Q_h^*| + |Q_h^* - Q^*|,$$

with proper scaling. Note that the first term is the distance from the approximate Q value, whereas the second term is the approximation error. We can see that after a certain value, increasing $h$ results on greater total error for these specific parameter intervals.

The sample complexity bound presented in (20) reveals that the increase is exponential in $\frac{1}{1-\beta_h}$, this rate is clearly not desired and it turns out that it can be avoided using different learning rates, $\alpha_k$, rather than using linear learning rates. For example, using polynomial learning rates, $\alpha_k = \frac{1}{k^\omega}$ for some $\omega \in (1/2, 1)$, one can achieve following sample complexity for $\epsilon$-near estimates for small enough $h$ (see [6, Theorem 4]):

$$T \lesssim \left( \frac{L^{1+3\omega}\ln(|X_h||U_h|)}{h^2\epsilon^2} \right)^{\frac{1}{\omega}} + \left( \frac{L}{h} \ln\left(\frac{1}{\epsilon}\right) \right)^{\frac{1}{1-\omega}}.$$
where the exponential increase with respect to $h$ is eliminated.

Further improvements and variations can be achieved for the sample complexity and the convergence rate, using different learning rates, e.g. rescaled learning rates, carefully chosen constant learning rates, or with different variations of the Q learning algorithm such as speedy Q learning, or variance reduced Q learning (see e.g. [1, 30, 19]).

We note that, even though different convergence rates can be derived using different reinforcement learning algorithms, if the algorithms are constructed using the time and state discretization procedure as in this paper, the learned value function will be the value function of the approximate MDP model constructed in Section 2.2. Hence, the convergence speed can be improved with different learning rates or different Q learning variants, however.

$$\sqrt{h} + \frac{L_X + L_Y}{h} + h^{\frac{1}{4}}$$

which is the error upper-bound for the approximation via state and time discretization and piece-wise constant polices, will not change.

6 Conclusion

We have constructed an approximate Q learning algorithm for a controlled diffusion process through discretization in time and space. We have showed that this algorithm converges under an ergodicity assumption on the state process. Furthermore, we have showed that the limit Q values satisfy the optimality equation of a finite Markov decision process, which has the same
distribution as the diffusion process at the sampling points when the diffusion process is controlled with a piece-wise constant control process. Using these observations, we have derived upper bounds, as a function of the discretization parameters, for the approximation error of the learned policies compared to the performance of the optimal admissible control process.

Possible future directions, building on the analysis in this paper are as follows:

- When we discretize the state space, the aggregated state process is no longer a Markov process, hence, we use an ergodicity assumption, to guarantee the convergence of the Q learning algorithm. However, if the quantization is fine enough, one might expect the learning algorithm to stay in a set of values with sufficiently small variations even without the ergodicity assumption. Hence, a possible future problem is to relax the ergodicity assumption we consider here.

- To deal with continuous state and action spaces, we have used an aggregation approach, and we were able to provide rigorous error bounds. One can study the same problem using different function approximation methods to deal with the continuous spaces, such as linear function approximations or neural networks.

- In this paper, we have used the traditional Q learning algorithm, for the simplicity of the presentation. For faster learning rates, different variations such as the variance reduction techniques, can be considered. We note that, if one uses the same time and space quantization scheme, the learned value functions will be the same as in this paper, however, using different variations of the Q learning algorithm will change the learning speed.

- The analysis used in this paper, can be extended to partially observed systems using the results from [12, 13] under proper filter stability conditions.

- We have not considered the exploration and exploitation trade-off; one might study this relation considering the provided convergence rate in this paper.
A Proof of Proposition 2

Note that the way we constructed the controlled Markov chain in Section 2.1, implies that the state process for the controlled Markov chain, and the state process for the diffusion process have the same distribution at the sampling instances. That is

\[ X_k \sim X(k \times h), \text{ for all } k \in \mathbb{Z}_+. \]

We can then write that

\[
|W_\beta(x_0, u_h) - J_\beta(x_0, \gamma)| = \left| E \left[ \int_0^\infty e^{-\beta s} c(X(s), u_h(s)) \, ds \right] - \left( \sum_{k=0}^{\infty} \beta_h^k c_h(X_k, U_k) \right) \right|
\]

\[
= \left| E \left[ \sum_{k=0}^{\infty} \int_{kh}^{(k+1)h} e^{-\beta s} c(X(s), u_h(s)) \, ds \right] - \left( \sum_{k=0}^{\infty} \beta_h^k c_h(X_k, U_k) \right) \right|
\]

\[
\leq \sum_{k=0}^{\infty} \int_{kh}^{(k+1)h} E \left[ \left| e^{-\beta_s} c(X(s), u_h(s)) - e^{-\beta_h k c_h(X_k, U_k)} \right| \right] \, ds
\]

We know focus on the term inside:

\[
E \left[ \left| e^{-\beta_s} c(X(s), u_h(s)) - e^{-\beta_h k c_h(X_k, U_k)} \right| \right]
\]

\[
\leq E \left[ \left| e^{-\beta_s} c(X(s), u_h(s)) - e^{-\beta_h k c_h(X_k, U_k)} \right| + E \left[ \left| e^{-\beta_h k c_h(X_k, U_k)} \right| \right] \right]
\]

\[
\leq e^{-\beta_h k} (1 - e^{-\beta_h}) \|c\|_{\infty} + e^{-\beta_h k} KE \|X(s) - X_k\|
\]

for the last step, we used the fact that \( u_h(s) = U_k \) as it is a piece-wise constant control process. For the second term, we have that for \( s \in [kh, (k+1)h) \)

\[
X(s) = X_k + \int_{kh}^{s} b(x(r), u_h(r)) \, dr + \int_{kh}^{s} \sigma(x(r), u_h(r)) \, dB(r).
\]

Thus, using Assumption 2 we can write

\[
E \left[ \|X(s) - X_k\| \right] \leq \|b\|_{\infty} \int_{kh}^{s} dr + \|\sigma\|_{\infty} E \left[ \int_{kh}^{s} dB(r) \right]
\]

\[
\leq Bh + BE[|Z_h|] = B \left( h + \sqrt{\frac{2h}{\pi}} \right)
\]

where \( Z_h \) is normally distributed with mean 0 and variance \( h \).
By combining everything we have so far, we write

$$\begin{align*}
|W_\beta(x_0, u_h) - J_\beta(x_0, \gamma)| & \leq \sum_{k=0}^{\infty} \int_{kh}^{(k+1)h} e^{-\beta hk} (1 - e^{-\beta h}) \|c\|_\infty + e^{-\beta hk} KB \left( h + \sqrt{\frac{2h}{\pi}} \right) ds \\
& = \sum_{k=0}^{\infty} \left( e^{-\beta hk} (1 - e^{-\beta h}) \|c\|_\infty + e^{-\beta hk} KB \left( h + \sqrt{\frac{2h}{\pi}} \right) \right) h \\
& = hB + \frac{KBh}{1 - e^{-\beta h}} \left( h + \sqrt{\frac{2h}{\pi}} \right)
\end{align*}$$

By combining everything we have so far, we write

$$\begin{align*}
|W_\beta(x_0, u_h) - J_\beta(x_0, \gamma)| & \leq \sum_{k=0}^{\infty} \int_{kh}^{(k+1)h} e^{-\beta hk} (1 - e^{-\beta h}) \|c\|_\infty + e^{-\beta hk} KB \left( h + \sqrt{\frac{2h}{\pi}} \right) ds \\
& = \sum_{k=0}^{\infty} \left( e^{-\beta hk} (1 - e^{-\beta h}) \|c\|_\infty + e^{-\beta hk} KB \left( h + \sqrt{\frac{2h}{\pi}} \right) \right) h \\
& = hB + \frac{KBh}{1 - e^{-\beta h}} \left( h + \sqrt{\frac{2h}{\pi}} \right)
\end{align*}$$

### B Proof of Lemma 1

We are interested in the distance between the distributions of the following random variables

$$x(h) = x(0) + \int_0^h b(x(s), u) ds + \int_0^h \sigma(x(s), u) dB(s)$$

$$y(h) = y(0) + \int_0^h b(y(s), u) ds + \int_0^h \sigma(y(s), u) dB(s).$$

We define $z(h) := x(h) - y(s)$, whose dynamics are given by

$$z(h) = z(0) + \int_0^h \tilde{b}(x(s), y(s), u) ds + \int_0^h \tilde{\sigma}(x(s), y(s), u) dB(s)$$

where

$$\tilde{b}(x(s), y(s), u) := b(x(s), u) - b(y(s), u)$$

$$\tilde{\sigma}(x(s), y(s), u) := \sigma(x(s), u) - \sigma(y(s), u).$$

Using the Ito formula

$$E[Z(h)^2] \leq E[Z_0^2] + \int_0^h 2Z_t \tilde{b}(x(s), y(s), u) ds + \int_0^h \tilde{\sigma}^2(x(s), y(s), u) ds$$

$$\leq Z_0^2 + 2K \int_0^h E[Z_s^2] ds + K^2 \int_0^h E[Z_s^2] ds.$$

We can then use Gronwall’s inequality to write

$$E[Z(h)^2] \leq Z_0^2 e^{(2K + K^2)h}.$$
Using the Holder’s inequality, we can further write

\[ E[|X(h) - Y(h)|] \leq |x(0) - y(0)| e^{(K + \frac{K^2}{2})h}, \]

which concludes the proof.

\section*{C Proof of Lemma 2}

We first define the following random variables

\[ x(h) = x(0) + \int_0^h b(x(s), u) ds + \int_0^h \sigma(x(s), u) dB(s) \]
\[ y(h) = x(0) + \int_0^h b(y(s), \hat{u}) ds + \int_0^h \sigma(y(s), \hat{u}) dB(s). \]

By defining \( z(h) := x(h) - y(s) \), whose dynamics are given by

\[ z(h) = \int_0^h \tilde{b}(x(s), y(s), u, \hat{u}) ds + \int_0^h \tilde{\sigma}(x(s), y(s), u, \hat{u}) dB(s) \]

where

\[ \tilde{b}(x(s), y(s), u, \hat{u}) := b(x(s), u) - b(y(s), \hat{u}) \]
\[ \tilde{\sigma}(x(s), y(s), u, \hat{u}) := \sigma(x(s), u) - \sigma(y(s), \hat{u}). \]

Using the Holder’s inequality, the boundedness, and the continuity properties of \( b, \sigma \), under the assumption that \( h < 1 \), we can write

\[ E[Z(h)^2] \leq E\left[ \int_0^h (KZ(s) + K|u - \hat{u}|)^2 ds \right] + E\left[ \left( \int_0^h (KZ(s) + K|u - \hat{u}|) dB(s) \right)^2 \right] \]
\[ = 2E\left[ \int_0^h (KZ(s) + K|u - \hat{u}|)^2 ds \right] \leq 4K^2 \int_0^h E[Z(s)^2] ds + 4K^2h|u - \hat{u}|^2 \]

under the assumption that \( h < 1 \), and using the Gronwall inequality

\[ E[Z(h)^2] \leq 4K^2|u - \hat{u}|^2 e^{4K^2h}. \]

Using the Holder’s inequality, we can conclude that

\[ E[|X(h) - Y(h)|] \leq 2Ke^{2K^2h}|u - \hat{u}|. \]
D Proof of Proposition 5

Proof. Note that the sampled state process \( \{X(i \times h)\}_i \) is a controlled Markov chain as it is described in Section 2.1. Furthermore, with the map \( \phi_X \), the algorithm (18), learns a quantized version of a controlled Markov chain. Hence, the result is a direct consequence of [11, Theorem 3.1]. However, we give a sketch of the proof for completeness.

We start by writing (18), in a more compact from by defining \( C_h := c(x(i \times h), \hat{u}) \times h \), and \( \hat{X}_1 := \phi_X(X(\langle i+1 \rangle \times h)) \):

\[
Q_{k+1}(\hat{x}, \hat{u}) = (1 - \alpha_k(\hat{x}, \hat{u})) Q_k(\hat{x}, \hat{u}) + \alpha_k(\hat{x}, \hat{u}) \left( C_h + \beta_h \min_{\hat{v} \in \hat{U}_h} Q_k(\hat{X}_1, \hat{v}) \right)
\]

We define

\[
\begin{align*}
\Delta_k(\hat{x}, \hat{u}) &:= Q_k(\hat{x}, \hat{u}) - Q_h^k(\hat{x}, \hat{u}) \\
F_k(\hat{x}, \hat{u}) &:= C_h + \beta_h V_k(\hat{X}_1) - Q_h^k(\hat{x}, \hat{u}) \\
\hat{F}_k(\hat{x}, \hat{u}) &:= C_h^* (\hat{x}, \hat{u}) + \beta_h \sum_{\hat{x}_1} V_k(\hat{x}_1) P_h^k(\hat{x}_1 | \hat{x}, \hat{u}) - Q_h^k(\hat{x}, \hat{u}),
\end{align*}
\]

where \( V_k(\hat{x}) := \min_{\hat{v} \in \hat{U}_h} Q_k(\hat{x}, \hat{v}) \).

Then, we can write the following iteration

\[
\Delta_{k+1}(\hat{x}, \hat{u}) = (1 - \alpha_k(\hat{x}, \hat{u})) \Delta_k(\hat{x}, \hat{u}) + \alpha_k(\hat{x}, \hat{u}) F_k(\hat{x}, \hat{u}).
\]

Now, we write \( \Delta_k = \delta_k + w_k \) such that

\[
\begin{align*}
\delta_{k+1}(\hat{x}, \hat{u}) &:= (1 - \alpha_k(\hat{x}, \hat{u})) \delta_k(\hat{x}, \hat{u}) + \alpha_k(\hat{x}, \hat{u}) \hat{F}_k(\hat{x}, \hat{u}) \\
w_{k+1}(\hat{x}, \hat{u}) &:= (1 - \alpha_k(\hat{x}, \hat{u})) w_k(\hat{x}, \hat{u}) + \alpha_k(\hat{x}, \hat{u}) r_k(\hat{x}, \hat{u})
\end{align*}
\]

where \( r_k := F_k - \hat{F}_k = \beta V_k(\hat{X}_1) - \beta_h \sum_{\hat{x}_1} V_k(\hat{x}_1) P_h^k(\hat{x}_1 | \hat{x}, \hat{u}) + C_h - C_h^k(\hat{x}, \hat{u}) \).

Next, we define

\[
\begin{align*}
r_k^*(\hat{x}, \hat{u}) &:= \beta h V^*(\hat{X}_1) - \beta h \sum_{\hat{x}_1} V^*(\hat{x}_1) P_h^k(\hat{x}_1 | \hat{x}, \hat{u}) + C_h - C_h^k(\hat{x}, \hat{u})
\end{align*}
\]

We further separate \( w_k = u_k + v_k \) such that

\[
\begin{align*}
u_{k+1}(\hat{x}, \hat{u}) &:= (1 - \alpha_k(\hat{x}, \hat{u})) u_k(\hat{x}, \hat{u}) + \alpha_k(\hat{x}, \hat{u}) e_k(\hat{x}, \hat{u}) \\
v_{k+1}(\hat{x}, \hat{u}) &:= (1 - \alpha_k(\hat{x}, \hat{u})) v_k(\hat{x}, \hat{u}) + \alpha_k(\hat{x}, \hat{u}) r_k^*(\hat{x}, \hat{u})
\end{align*}
\]
where $e_k = r_k - r^*_k$.

We now show that $v_k(\hat{x}, \hat{u}) \to 0$ almost surely for all $(\hat{x}, \hat{u})$. Note that, because of the way we chose the learning rates $\alpha_k$, $v_k(\hat{x}, \hat{u})$ is only updated when the process hits $\hat{x}, \hat{u}$ during the exploration. Thus, we define the following stopping times

$$\tau(n + 1) = \{ \min k > \tau(n) : \phi_{\mathcal{X}}(x_k) = \hat{x}, u_k = \hat{u} \}$$

where $\tau(0) = 0$. In what follows, we will focus on the $v_n(\hat{x}, \hat{u})$ process, since we assume that every $(\hat{x}, \hat{u})$ pair is visited infinitely often, these stopping times are bounded almost surely and hence, we can make sure that $n \to \infty$ as $k \to \infty$. Furthermore, we have that $\alpha_n(\hat{x}, \hat{u}) = \frac{1}{n}$, as $n$ is the number of times the we have hit $(\hat{x}, \hat{u})$ pair.

When $\alpha_n(\hat{x}, \hat{u}) = \frac{1}{n}$ for every $(\hat{x}, \hat{u})$ pair, the problem reduces to

$$v_{n+1}(\hat{x}, \hat{u}) = \frac{1}{n} \sum_{n' = 0}^{n-1} r^*_n(\hat{x}, \hat{u}).$$

Above, even though, we do not write the time dependence on $\hat{X}_1$, distribution of the $\hat{X}_1$ is different every time we make the update, since the marginal distribution of $x_i$ is different. However, using the fact that the random variables $(\hat{X}_1, X_1)$ form a controlled Markov chain since they are sampled at the stopping times $\tau(k)$, and using the invariant measure of the original state process, we can write

$$\lim_{n \to \infty} \frac{1}{n} \sum_{n' = 0}^{n-1} V^*(\hat{X}_1) = \int_B \sum_j V^*(\hat{x}_j) \mathcal{T}_h(B_j|x, \hat{u}) \hat{\pi}^*_x(dx)$$

$$\lim_{n \to \infty} \frac{1}{n} \sum_{n' = 0}^{n-1} C_h = \lim_{n \to \infty} \frac{1}{n} \sum_{n' = 0}^{n-1} c(x(\tau(n') \times h), \hat{u}) \times h = \int_B c_h(x, u) \hat{\pi}^*_x(dx).$$

where $B$ is the quantization bin $\hat{x}$ belongs to. Hence, we have proved that $v_k(\hat{x}, \hat{u}) \to 0$, noting

$$C^*_h(\hat{x}_i, u) = \int_{B_i} c_h(x, u) \hat{\pi}^*_x(dx)$$

$$P^*_h(\hat{x}_j|\hat{x}_i, u) = \int_{B_i} \mathcal{T}_h(B_j|x, u) \hat{\pi}^*_x(dx).$$
Now, we go back to the iterations:

\[
\begin{align*}
\delta_{k+1}(\hat{x}, \hat{u}) &= (1 - \alpha_k(\hat{x}, \hat{u}))(\delta_k(\hat{x}, \hat{u}) + \alpha_k(\hat{x}, \hat{u})\hat{F}_k(\hat{x}, \hat{u})) \\
u_{k+1}(\hat{x}, \hat{u}) &= (1 - \alpha_k(\hat{x}, \hat{u}))(\hat{u}k(\hat{x}, \hat{u}) + \alpha_k(\hat{x}, \hat{u})e_k(\hat{x}, \hat{u})) \\
v_{k+1}(\hat{x}, \hat{u}) &= (1 - \alpha_k(\hat{x}, \hat{u}))(\hat{v}_k(\hat{x}, \hat{u}) + \alpha_k(\hat{x}, \hat{u})r^*_k(\hat{x}, \hat{u})).
\end{align*}
\]

Note that, we want to show \(\Delta_k = \delta_k + u_k + v_k \to 0\) almost surely and we have that \(v_k(\hat{x}, \hat{u}) \to 0\) almost surely for all \((\hat{x}, \hat{u})\). The following analysis holds for any path that belongs to the probability one event in which \(v_k(I, u) \to 0\). For any such path and for any given \(\epsilon > 0\), we can find an \(N < \infty\) such that \(\|v_k\|_\infty < \epsilon\) for all \(k > N\) as \((I, u)\) takes values from a finite set.

We now focus on the term \(\delta_k + u_k\) for \(k > N\):

\[
(\delta_{k+1} + u_{k+1})(\hat{x}, \hat{u}) = (1 - \alpha_k(\hat{x}, \hat{u}))(\delta_k + u_k)(\hat{x}, \hat{u}) + \alpha_k(\hat{x}, \hat{u})(\hat{F}_k + e_k)(\hat{x}, \hat{u}).
\]

Observe that for \(k > N\),

\[
(\hat{F}_k + e_k)(\hat{x}, \hat{u}) = (F_k - r^*_k)(\hat{x}, \hat{u}) = \beta_h V_k(\hat{X}_1) - \beta_h V^*(\hat{X}_1) \leq \beta_h \max_{\hat{x}, \hat{u}} |Q_k(\hat{x}, \hat{u}) - Q^*(\hat{x}, \hat{u})| = \beta_h \|\Delta_k\|_\infty
\]

\[
\leq \beta_h \|\delta_k + u_k\|_\infty + \beta_h \epsilon
\]

where the last step follows from the fact that \(v_k \to 0\) almost surely. By choosing \(C < \infty\) such that \(\beta := \beta_h(C + 1)/C < 1\), for \(\|\delta_k + u_k\|_\infty > C\epsilon\), we can write that

\[
\beta_h \|\delta_k + u_k + \epsilon\|_\infty \leq \beta \|\delta_k + u_k\|_\infty.
\]

Now we rewrite (22)

\[
(\delta_{k+1} + u_{k+1})(\hat{x}, \hat{u}) \leq (1 - \alpha_k(\hat{x}, \hat{u}))(\delta_k + u_k)(\hat{x}, \hat{u}) + \alpha_k(\hat{x}, \hat{u})(\hat{F}_k + e_k)(\hat{x}, \hat{u}) \leq (1 - \alpha_k(\hat{x}, \hat{u}))(\delta_k + u_k)(\hat{x}, \hat{u}) + \alpha_k(\hat{x}, \hat{u})\beta \|\delta_k + u_k\|_\infty.
\]

By [9, Lemma 3], \((\delta_{k+1} + u_{k+1})(\hat{x}, \hat{u})\) tends to 0 for \(\|\delta_k + u_k\|_\infty > C\epsilon\). This shows that the condition \(\|\delta_k + u_k\|_\infty > C\epsilon\) cannot be sustained indefinitely. Next, we show that once the process hits below \(C\epsilon\) it always stays there. Suppose \(\|\delta_k + u_k\|_\infty < C\epsilon\),

\[
(\delta_{k+1} + u_{k+1})(\hat{x}, \hat{u}) \leq (1 - \alpha_k(\hat{x}, \hat{u}))(\delta_k + u_k)(\hat{x}, \hat{u}) + \alpha_k(\hat{x}, \hat{u})\beta \|\delta_k + u_k\|_\infty + \epsilon
\]

\[
\leq (1 - \alpha_k(\hat{x}, \hat{u}))(C\epsilon + \alpha_k(\hat{x}, \hat{u})\beta \|\delta_k + u_k\|_\infty + \epsilon)
\]

\[
= (1 - \alpha_k(\hat{x}, \hat{u}))C\epsilon + \alpha_k(\hat{x}, \hat{u})\beta \|\delta_k + u_k\|_\infty + \epsilon
\]

\[
\leq (1 - \alpha_k(\hat{x}, \hat{u}))C\epsilon + \alpha_k(\hat{x}, \hat{u})C\epsilon, \quad (\beta \|\delta_k + u_k\|_\infty + \epsilon)
\]

\[
= C\epsilon.
\]

29
Then, we can write \( \|\delta_{k+1} + u_{k+1}\|_\infty < C\epsilon \).

Thus, taking \( \epsilon \to 0 \), we can conclude that \( \Delta_k = \delta_k + u_k + v_k \to 0 \) almost surely.

Therefore, the process \( Q_k \), determined by the algorithm in (18), converges almost surely to \( Q^*_h \). \qed

References

[1] M. G. Azar, R. Munos, M. Ghavamzadeh, and H. J. Kappen. Speedy q-learning. In *NIPS*, 2011.

[2] L. C. Baird. Reinforcement learning in continuous time: Advantage updating. In *Proceedings of 1994 IEEE International Conference on Neural Networks (ICNN’94)*, volume 4, pages 2448–2453. IEEE, 1994.

[3] G. Barles and E.R. Jakobsen. Error bounds for monotone approximation schemes for Hamilton-Jacobi-Bellman equations. *SIAM J. Numer. Anal.*, 43(2):540–558.

[4] G. Barles and E.R. Jakobsen. On the convergence rate of approximation schemes for Hamilton-Jacobi-Bellman equations. *Math. Model. Numer. Anal.*, 36:33–54.

[5] K. Doya. Reinforcement learning in continuous time and space. *Neural computation*, 12(1):219–245, 2000.

[6] E. Even-Dar, Y. Mansour, and P. Bartlett. Learning rates for q-learning. *Journal of machine learning Research*, 5(1), 2003.

[7] C. Gaskett and A. Zelinsky D. Wettergreen. Q-learning in continuous state and action spaces. In *Australasian joint conference on artificial intelligence*, pages 417–428. Springer, 1999.

[8] M. Hairer. Convergence of markov processes. *Lecture Notes 2010*.

[9] T. Jaakkola, M. I. Jordan, and S. P. Singh. On the convergence of stochastic iterative dynamic programming algorithms. *Neural computation*, 6(6):1185–1201, 1994.

[10] E. R. Jakobsen, A. Picarelli, and C. Reisinger. Improved order 1/4 convergence for piecewise constant policy approximation of stochastic control problems. *Electronic Communications in Probability*, 24:1–10, 2019.
[11] A. D. Kara, N. Saldi, and S. Yüksel. Q-learning for mdps with general spaces: Convergence and near optimality via quantization under weak continuity. arXiv preprint arXiv:2111.06781, 2021.

[12] A. D. Kara and S. Yuksel. Convergence of finite memory q-learning for pomdps and near optimality of learned policies under filter stability. arXiv preprint arXiv:2103.12158, 2021.

[13] A. D. Kara and S. Yuksel. Near optimality of finite memory feedback policies in partially observed markov decision processes. Journal of Machine Learning Research, 23(1):1–46, 2022.

[14] J. Kim, J. Shin, and I. Yang. Hamilton-jacobi deep q-learning for deterministic continuous-time systems with Lipschitz continuous controls. Journal of Machine Learning Research, 22(206):1–34, 2021.

[15] N. V. Krylov. On the rate of convergence of finite-difference approximations for Bellman’s equations. St. Petersburg Math. J., 9:639–650.

[16] N. V. Krylov. On the rate of convergence of finite-difference approximations for Bellman’s equations with variable coefficients. Probab. Theory Relat. Fields, 117:1–16.

[17] N. V. Krylov. Approximating value functions for controlled degenerate diffusion processes by using piece-wise constant policies. Electronic Journal of Probability, 4:1–19, 1999.

[18] H.J. Kushner and P.G. Dupuis. Numerical methods for stochastic control problems in continuous time, volume 24. Springer Science & Business Media, 2001.

[19] G. Li, Y. Wei, Y. Chi, Y. Gu, and Y. Chen. Sample complexity of asynchronous q-learning: Sharper analysis and variance reduction. arXiv preprint arXiv:2006.03041, 2020.

[20] F. C. Melo, S. P. Meyn, and I. M. Ribeiro. An analysis of reinforcement learning with function approximation. In Proceedings of the 25th international conference on Machine learning, pages 664–671, 2008.

[21] R. Munos and P. Bourgine. Reinforcement learning for continuous stochastic control problems. Advances in neural information processing systems, 10, 1997.

31
[22] R. Munos and A. Moore. Variable resolution discretization in optimal control. *Machine learning*, 49(2):291–323, 2002.

[23] D. Shah and Q. Xie. Q-learning with nearest neighbors. *arXiv preprint arXiv:1802.03900*, 2018.

[24] S. P. Singh, T. Jaakkola, and M. I. Jordan. Reinforcement learning with soft state aggregation. *Advances in neural information processing systems*, pages 361–368, 1995.

[25] C. Szepesvári. Algorithms for reinforcement learning. volume 4, pages 1–103, 2010.

[26] C. Szepesvári and William D. Smart. Interpolation-based q-learning. 2004.

[27] C. Tallec, L. Blier, and Y. Ollivier. Making deep q-learning methods robust to time discretization. In *International Conference on Machine Learning*, pages 6096–6104. PMLR, 2019.

[28] J. N. Tsitsiklis. Asynchronous stochastic approximation and q-learning. *Machine Learning*, 16:185–202, 1994.

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

[30] M. Wainwright. Stochastic approximation with cone-contractive operators: Sharper $l_\infty$-bounds for q-learning. *arXiv preprint arXiv:1905.06265*, 2019.

[31] H. Wang, T. Zariphopoulou, and X. Y. Zhou. Reinforcement learning in continuous time and space: A stochastic control approach. *J. Mach. Learn. Res.*, 21(198):1–34, 2020.

[32] H. Wang and X. Y. Zhou. Continuous-time mean–variance portfolio selection: A reinforcement learning framework. *Mathematical Finance*, 30(4):1273–1308, 2020.

[33] C. J. C. H. Watkins and P. Dayan. Q-learning. *Machine Learning*, 8:279–292, 1992.