

Reinforcement Learning for Zero-Delay Coding of Markov Sources: Quantized Q-Learning for Near-Optimal Quantization

Liam Cregg, Tamás Linder and Serdar Yüksel*

December 14, 2022

Abstract

In the classical lossy coding problem, one is allowed to encode long sequences of source symbols in order to achieve a lower distortion. This is undesirable in many delay-sensitive applications. Accordingly, we consider the zero-delay case where one wishes to encode a source symbol causally. It has been shown that this problem lends itself to stochastic control techniques, leading to existence and structural results. However, attempting to develop analytical solutions or algorithmic implementation using these methods has been computationally difficult. To that end, we propose a reinforcement learning approach. First, we will establish some supporting results on regularity and stability properties. Afterwards, building on recent results on quantized Q-learning, we show that a quantized Q-learning algorithm can be used to obtain a near-optimal policy for this problem. Finally, we provide some relevant simulation results.

1 Zero-Delay Lossy Coding and Optimal Quantizers

We are interested in a variant of Shannon's lossy source coding problem: Given an information source $\{X_t\}_{t \geq 0}$ from a finite alphabet \mathbb{X} , we wish to compress this source at rate R bits per source symbol, and then reproduce the source as $\{\hat{X}_t\}_{t \geq 0}$, where \mathbb{X} is the (also finite) reproduction alphabet. In particular, a $(2^{RT}, T)$ block code encodes T source symbols $X_{[0, T-1]}$ as $\gamma^e : \mathbb{X}^T \rightarrow \{1, \dots, 2^{RT}\}$, and decodes them as $\gamma^d : \{1, \dots, 2^{RT}\} \rightarrow \mathbb{X}^T$. The goal is often to minimize the distortion, given by:

$$D_T(R) := \frac{1}{T} E \left[\sum_{t=0}^{T-1} d(X_t, \hat{X}_t) \right]$$

where $d : \mathbb{X} \times \mathbb{X} \rightarrow [0, \infty)$ is an (additive) distortion measure. We call a rate-distortion pair (R, D) achievable if there exists a sequence of $(2^{RT}, T)$ codes (γ^e, γ^d) such that

$$\limsup_{T \rightarrow \infty} D_T(R) \leq D$$

The minimum achievable distortion for a given rate R is denoted by $D(R)$, and classically we can obtain this distortion by taking the block length T to infinity (provided that the source is stationary and ergodic), i.e.

*The authors are with the Department of Mathematics and Statistics, Queen's University, Kingston, Ontario, Canada, K7L 3N6 (e-mail: (liam.cregg, tamas.linder, yuksel)@queensu.ca). Research is supported by the Natural Sciences and Engineering Research Council of Canada. The first author was supported by a Queen's University Department of Mathematics and Statistics Summer Research Award.

$$D(R) = \lim_{T \rightarrow \infty} D_T(R)$$

However, this approach requires encoding very large blocks of data at a time, and hence incurs large delay. This delay may not be allowable in many real-time applications. Accordingly, we study a causal variant of this lossy coding problem, where we enforce zero delay (hence a block coding approach is not viable). We will assume throughout that the source $\{X_t\}_{t \geq 0}$ is a discrete-time Markov process with probability matrix P , which is irreducible and aperiodic (and thus admits a unique invariant measure). After encoding, the (compressed) information is sent over a discrete noiseless channel with input and output alphabets $\mathcal{M} := \{1, \dots, M\}$.

Thus, the encoder is defined by an encoder policy $\{\gamma_t^e\}_{t \geq 0}$, where $\gamma_t^e : \mathcal{M}^t \times \mathbb{X}^{t+1} \rightarrow \mathcal{M}$. That is, the encoder can use all past encoder outputs and all past and current source inputs to generate the current encoder output. This can be viewed as the encoder policy selecting a quantizer $Q_t : \mathbb{X} \rightarrow \mathcal{M}$ using past information, then quantizing X_t as $q_t = Q_t(X_t)$ [1]. Then, the decoder generates the reconstruction \hat{X}_t without delay, using decoder policy $\{\gamma_t^d\}_{t \geq 0}$, where $\gamma_t^d : \mathcal{M}^{t+1} \rightarrow \mathbb{X}$. Thus we have $\hat{X}_t = \gamma_t^d(q_{[0,t]})$.

Note that, since the source alphabet is finite, there exists an optimal decoding policy for every encoding policy. Thus we will denote (with an abuse of notation) the encoding policy by $\gamma := \gamma^e$, and assume it is paired with an optimal decoding policy. We can then restrict our search only to optimal encoding policies.

In general, for the zero-delay coding problem, the goal is to minimize the average cost/distortion. In the infinite horizon case where $X_0 \sim \mu$, this is given by:

$$J(\mu, \gamma) := \limsup_{T \rightarrow \infty} \mathbf{E}_\mu^\gamma \left[\frac{1}{T} \sum_{t=0}^{T-1} d(X_t, \hat{X}_t) \right]$$

However, we will also consider the discounted cost problem, as this problem is easier to tackle using reinforcement learning methods (to be discussed later). That is, for some $\beta \in (0, 1)$, we wish to minimize:

$$J_\beta(\mu, \gamma) := \lim_{T \rightarrow \infty} \mathbf{E}_\mu^\gamma \left[\frac{1}{T} \sum_{t=0}^{T-1} \beta^t d(X_t, \hat{X}_t) \right]$$

For the finite horizon problems, we have the following results on the structure of optimal zero-delay codes, the first from Witsenhausen and the second from Walrand and Varaiya:

Theorem 1.1. [2] *For the problem of coding a Markov source over a finite time horizon T , any zero delay encoder policy $\gamma = \{\gamma_t\}$ can be replaced, without loss in distortion performance, by a policy $\hat{\gamma} = \{\hat{\gamma}_t\}$ which only uses $q_{[0,t-1]}$ and X_t to generate q_t , i.e., such that $q_t = \hat{\gamma}_t(q_{[0,t-1]}, X_t) \forall t = 1, \dots, T-1$.*

While this greatly simplifies our optimal encoder, we still have that its memory space expands with time. To avoid this, we define the following conditional probability measure on \mathbb{X} . Let $\mathcal{P}(\mathbb{X})$ be the space of probability measures on \mathbb{X} , and define $\pi_t \in \mathcal{P}(\mathbb{X})$ as:

$$\pi_t(A) := \Pr(X_t \in A | q_{[0,t-1]})$$

Theorem 1.2. [3] *For the problem of coding a Markov source over a finite time horizon T , any zero delay encoder policy $\gamma = \{\gamma_t\}$ can be replaced, without loss in distortion performance, by a policy $\hat{\gamma} = \{\hat{\gamma}_t\}$ which only uses π_t and X_t to generate q_t , i.e., such that $q_t = \hat{\gamma}_t(\pi_t, X_t) \forall t = 1, \dots, T-1$. Alternatively, at time t such a policy uses π_t to select a quantizer $Q_t = \hat{\gamma}_t(\pi_t)$ (where $Q_t : \mathbb{X} \rightarrow \mathcal{M}$), and then q_t is obtained by $q_t = Q_t(X_t)$.*

Encoders with the above structure are called *Walrand-Varaiya type policies* in [1], [4], or alternatively, *Markov policies* (because they endow Markov properties onto the process $\{\pi_t\}$, which we will see momentarily). We also have an infinite-horizon analog of the above theorem, which was proven in [4]:

Theorem 1.3. [4, Proposition 2] For the problem of coding an irreducible and aperiodic Markov source, for any initial distribution μ , there exists a stationary Walrand-Varaiya type policy γ^* that solves the infinite-horizon discounted cost problem, i.e. one that satisfies:

$$J_\beta(\mu, \gamma^*) = \inf_{\gamma \in \Gamma} J_\beta(\mu, \gamma)$$

where Γ is the set of all admissible encoder policies and $J_\beta(\mu, \gamma)$ is defined in (1).

Under Walrand-Varaiya type policies, it was shown in [1] that $\{\pi_t\}$ is a controlled Markov process with control $\{Q_t\}$. More specifically, we have the following result:

Theorem 1.4. [1] Under a Walrand-Varaiya type policy, the update equation for π_t is given by

$$\pi_{t+1}(x_{t+1}) = \frac{1}{\pi_t(Q_t^{-1}(q_t))} \sum_{x_t \in Q_t^{-1}(q_t)} P(x_{t+1}|x_t) \pi_t(x_t) \quad (1)$$

Therefore π_{t+1} is conditionally independent of $(\pi_{[0,t-1]}, Q_{[0,t-1]})$ given π_t and Q_t , and hence $\{\pi_t\}$ is a controlled Markov process with control $\{Q_t\}$.

Proof. Assume that we use a Walrand-Varaiya type policy. Then the quantizer output q_t is determined entirely by π_t and x_t . That is, $P(q_t|\pi_t, x_t) = 1_{\{Q_t(x_t)=q_t\}}$, where $Q_t = \gamma_t(\pi_t)$. Then we have:

$$\begin{aligned} \pi_{t+1}(x_{t+1}) &= \frac{P(x_{t+1}, q_t|q_{[0,t-1]})}{P(q_t|q_{[0,t-1]})} \\ &= \frac{\sum_{x_t \in \mathbb{X}} \pi_t(x_t) P(q_t|\pi_t, x_t) P(x_{t+1}|x_t)}{\sum_{x_{t+1} \in \mathbb{X}} \sum_{x_t \in \mathbb{X}} \pi_t(x_t) P(q_t|\pi_t, x_t) P(x_{t+1}|x_t)} \end{aligned}$$

Using the above fact that $P(q_t|\pi_t, x_t) = 1_{\{Q_t(x_t)=q_t\}}$, this simplifies to:

$$\pi_{t+1}(x_{t+1}) = \frac{1}{\pi_t(Q_t^{-1}(q_t))} \sum_{x_t \in Q_t^{-1}(q_t)} P(x_{t+1}|x_t) \pi_t(x_t)$$

□

We will denote the transition kernel induced by the above update equation by $P(d\pi_{t+1}|\pi_t, Q_t)$ (this is a distribution on $\mathcal{P}(\mathbb{X})$). We also define the following cost function for this Markov decision process (MDP) in terms of π_t and Q_t (this is the average distortion if the optimal decoder is used for a given Q_t).

$$c(\pi_t, Q_t) := \sum_{i=1}^M \min_{\hat{x} \in \mathbb{X}} \sum_{x \in Q_t^{-1}(i)} \pi_t(x) d(x, \hat{x}) \quad (2)$$

Note that by this definition of $c(\pi_t, Q_t)$ and our assumption that we are using an optimal decoder for a given encoder, we have:

$$\mathbf{E}_\mu^\gamma \left[\frac{1}{T} \sum_{t=0}^{T-1} c(\pi_t, Q_t) \right] = \mathbf{E}_\mu^\gamma \left[\frac{1}{T} \sum_{t=0}^{T-1} d(X_t, \hat{X}_t) \right]$$

1.1 A Topology on Quantizers

As we will be discussing convergence and continuity regarding quantizers, we need to define an appropriate topology. Viewing a quantizer Q as a map from \mathbb{X} to \mathcal{M} , we denote the i^{th} bin of Q as $B_i = Q^{-1}(i)$, $i = 1, \dots, M$, and we denote the set of all possible quantizers by \mathcal{Q} (since \mathbb{X} is finite, so is \mathcal{Q}). Following [1], [5], we note that a quantizer with bins $\{B_1, \dots, B_M\}$ can alternatively be represented as a stochastic kernel from \mathbb{X} to \mathcal{M} such that $Q(i|x) = 1_{\{x \in B_i\}}$, $i = 1, \dots, M$. Then if P is a probability measure on \mathbb{X} , we denote by PQ the joint probability measure $PQ(x, y) = P(x)Q(y|x)$. If we introduce the equivalence relation $Q \equiv Q'$ iff $PQ = P'Q'$, then we can imbue these equivalence classes with the weak convergence topology (that is, we say $Q_n \rightarrow Q$ weakly iff $PQ_n \rightarrow PQ$ weakly). Under this topology, [1] showed the following property of the controlled Markov chain $\{\pi_t\}$.

Lemma 1.5. [1, Lemma 11]. *The transition kernel $P(d\pi_{t+1}|\pi_t, Q_t)$ is weakly continuous in (π_t, Q_t) . That is,*

$$\int_{\mathcal{P}(\mathbb{X}) \times \mathcal{Q}} f(\pi') P(d\pi'|\pi, Q)$$

is continuous on $\mathcal{P}(\mathbb{X}) \times \mathcal{Q}$ for all continuous and bounded f .

We briefly remark on the setup so far.

Remark 1.

1. We now have a controlled Markov chain $\{\pi_t\}$, which takes values in $\mathcal{P}(\mathbb{X})$, with Q -valued control $\{Q_t\}$. Equipped with an appropriate cost function (2), we see that finding an optimal Walrand-Varaiya type policy is equivalent to finding an optimal policy for this MDP (see Section 2 and [6] for more information on MDPs). Furthermore, in light of Theorem 1.3, this Walrand-Varaiya type policy is optimal for the discounted-cost zero-delay coding problem. Other important structural results, such as near-optimality conditions and existence of Walrand-Varaiya type solutions for the average-cost problem were also presented in [4], leveraging this stochastic control perspective.
2. Given this formulation, we would like to utilize tools from stochastic control to find this optimal policy in practice. Indeed, if we are given the transition kernel $P(x_{t+1}|x_t)$, then we know exactly how $\{\pi_t\}$ evolves and the cost function (2). Then, there exist iterative algorithms (in fact, such algorithms are used in [2] and [4] to prove some of the above theorems) to solve for an optimal policy. However, in this setup, our dynamics and the transition kernel are quite complicated and so implementing such an algorithm is very difficult. Therefore, we propose to use reinforcement learning.
3. We note that, although our information source \mathbb{X} is finite, the state space for our new MDP is $\mathcal{P}(\mathbb{X})$ and therefore uncountable. We will see how this impacts the approach proposed above in the following section.

2 Q-learning and Quantized Q-learning

We begin with a remark on notation. In order to align with the standard Q-learning literature, we have re-used a significant amount of notation from the previous section. We will unify the notation in Section 4, but for now we will consider the notation in this section to be totally distinct, and will redefine any notation shared with the previous section.

2.1 Q-learning for Finite Models

As in [6, Chapter 2], we define a *Markov decision process* as a 4-tuple $(\mathbb{X}, \mathbb{U}, P, c)$, where:

1. \mathbb{X} is the *state space*, which we assume is Polish (i.e. a Borel subset of a complete, separable metric space).
2. \mathbb{U} is the *action space*, also Polish.
3. $P = P(\cdot|x, u)$ is the *transition kernel*, a stochastic kernel on \mathbb{X} given $\mathbb{X} \times \mathbb{U}$.
4. $c : \mathbb{X} \times \mathbb{U} \rightarrow [0, \infty)$ is the *cost function*.

For now, we will assume \mathbb{X} and \mathbb{U} are both finite, and deal with the infinite case shortly.

An *admissible policy* is a sequence $\gamma = \{\gamma_t\}_{t \geq 0}$ such that $\gamma_t : \mathbb{U}^t \times \mathbb{X}^{t+1} \rightarrow \mathbb{U}$. Such a policy, along with the transition kernel P and an initial distribution $X_0 \sim \mu$, define a unique distribution for $(X_t, U_t)_{t \geq 0}$. The goal (for the infinite-horizon, discounted-cost case) is to find a policy γ minimizing:

$$J_\beta(\mu, \gamma) := \lim_{T \rightarrow \infty} E_\mu^\gamma \left[\sum_{t=0}^{T-1} \beta^t c(X_t, U_t) \right]$$

for some $\beta \in (0, 1)$.

We define the optimal value function as the above cost when an optimal policy is used:

$$J_\beta^*(\mu) := \inf_\gamma J_\beta(\mu, \gamma)$$

We also note that if $\mu = \delta_x$, we denote the above by $J_\beta^*(x)$. A key result in stochastic control theory is that a function satisfies the optimal value function iff it satisfies the discounted cost optimality equation (DCOE):

$$J_\beta^*(x) = \min_{u \in \mathbb{U}} \left\{ c(x, u) + \beta \sum_{y \in \mathbb{X}} J_\beta^*(y) P(y|x, u) \right\} \quad (3)$$

As previously mentioned, there are several algorithms one could use to find an optimal policy based on the above DCOE. However many of these require usage of the transition kernel, which (as noted in Remark 1.1) may be impossible or at least very complicated. A common workaround is to use Q-learning, which we now define.

Let $Q_t : \mathbb{X} \times \mathbb{U} \rightarrow \mathbb{R}$ be the *Q-factor* at time $t \geq 0$. Suppose that, starting at some arbitrary Q_0 , the decision maker applies an arbitrary admissible policy γ and for $t \geq 0$ updates its Q-factors as follows:

$$Q_{t+1}(x, u) = (1 - \alpha_t(x, u))Q_t(x, u) + \alpha_t(x, u)(c(x, u) + \beta \min_{v \in \mathbb{U}} Q_t((X_{t+1}, v))) \quad (4)$$

Assumption 2.1. For all (x, u) and for all $t \geq 0$, we have

1. $\alpha_t(x, u) \in [0, 1]$.
2. $\alpha_t(x, u) = 0$ if $(x, u) \neq (X_t, U_t)$.
3. $\alpha_t(x, u)$ is a (deterministic) function of $(x_0, u_0), \dots, (x_t, u_t)$.
4. $\sum_{t \geq 0} \alpha_t(x, u) = \infty$ almost surely.
5. $\sum_{t \geq 0} \alpha_t^2(x, u) < \infty$ almost surely.

A well-known result in stochastic control theory is the following:

Theorem 2.1. *Under Assumption 2.1, the algorithm in (4) converges almost surely to the fixed point $Q^*(x, u)$ of the following mapping:*

$$Q^*(x, u) = F(Q^*)(x, u) = c(x, u) + \beta \int_{\mathbb{X}} \min_{v \in \mathbb{U}} \{Q(y, u)\} P(dy|x, u)$$

Note that by taking the minimum of the above equation over \mathbb{U} for each x , we get the solution of the DCOE (3), and hence the policy made up of these minimizing actions is optimal.

Although a powerful algorithm, there is a clear issue when dealing with infinite state and/or action spaces. In particular, it is impossible to visit each state-action pair infinitely often, and therefore the algorithm is not guaranteed to converge. A potential solution is to use “quantized” Q-learning - that is, we approximate the original MDP using some other MDP with finite state and action spaces, and run Q-learning on this model. Under some additional assumptions, [7] showed that one can indeed achieve near-optimality for the original MDP in this fashion. We now mention some key results from that paper.

2.2 Finite Model Approximations

Action Space

Assumption 2.2. *Our original MDP has the following properties:*

1. *The stochastic kernel $P(\cdot|x, u)$ is weakly continuous in (x, u) , i.e. $P(\cdot|x_n, u_n) \rightarrow P(\cdot|x, u)$ weakly $\forall (x_n, u_n) \rightarrow (x, u)$.*
2. *The cost function c is continuous and bounded.*
3. *The action space \mathbb{U} is compact.*
4. *The state space \mathbb{X} is σ -compact.*

Since \mathbb{U} is assumed compact, there exists $\mathbb{U}_n := (u_{n,1}, \dots, u_{n,k_n}) \subset \mathbb{U}$ such that \mathbb{U}_n is a $\frac{1}{n}$ -net in \mathbb{U} . Then, let $\text{MDP}_n := (\mathbb{X}, \mathbb{U}_n, P, c)$ and define $J_{\beta,n}^*(x)$ as the optimal value function of MDP_n (as in (3)). Then we have the following:

Theorem 2.2. [8, Theorem 3.16] *Under Assumption 2.2, for any compact $K \subset \mathbb{X}$, we have*

$$\lim_{n \rightarrow \infty} \sup_{x \in K} |J_{\beta,n}^*(x) - J_{\beta}^*(x)| = 0$$

That is, we can approximate the MDP with a finite-state one. So in the following, we assume that Assumption 2.2 holds and that \mathbb{U} is finite.

State Space

Let $\{B_i\}_{i=1}^M$ be a partition of \mathbb{X} , and let $\mathbb{Y} := \{y_1, \dots, y_M\}$ where $y_i \in B_i$. We define a *quantizer* on \mathbb{X} as a mapping $q : \mathbb{X} \rightarrow \mathbb{Y}$, such that

$$q(x) = y_i \quad \text{if } x \in B_i$$

Now let $\psi \in \mathcal{P}(\mathbb{X})$ be a distribution on \mathbb{X} . Then with an abuse of notation we define the resulting conditional distribution

$$\psi(A|y_i) := \frac{\psi(A)}{\psi(B_i)}$$

Let $\hat{\text{MDP}} := (\mathbb{Y}, \mathbb{U}, \hat{c}, \hat{P})$, where \hat{c} and \hat{P} are defined as the mean of the original c and P over the quantization bins. That is:

$$\begin{aligned}\hat{c}(y_i, u) &:= \int_{B_i} c(x, u) \psi(dx|y_i) \\ \hat{P}(y_j|y_i, u) &:= \int_{B_i} P(B_j|x, u) \psi(dx|y_i)\end{aligned}\tag{5}$$

Then let \hat{J}_β be the optimal value function for $\hat{\text{MDP}}$, and note that we can extend this function over \mathbb{X} by making it constant over each B_i , i.e.

$$\hat{J}_\beta(x) := \hat{J}_\beta(y_i) \quad \forall x \in B_i$$

We also define:

$$\|L^-\|_\infty := \max_{i=1, \dots, M-1} \sup_{x, x' \in B_i} \|x - x'\|$$

And note that under Assumption 2.2, \mathbb{X} is σ -compact, and hence there exists a sequence of partitions $\{B_i\}_{i=1}^M$ such that $\|L^-\|_\infty \rightarrow 0$ and $\bigcup_{i=1}^{M-1} B_i \uparrow \mathbb{X}$ as $M \rightarrow \infty$. Then we have:

Theorem 2.3. [8, Theorem 4.27] Under Assumption 2.2, we have \forall compact $K \subset \mathbb{X}$ and as $\|L^-\|_\infty \rightarrow 0$,

$$\sup_{x_0 \in K} |\hat{J}_\beta(x_0) - J_\beta^*(x_0)| \rightarrow 0$$

and

$$\sup_{x_0 \in K} |J_\beta(x_0, \hat{\gamma}) - J_\beta^*(x_0)| \rightarrow 0$$

where $\hat{\gamma}$ is the optimal policy of $\hat{\text{MDP}}$ extended to \mathbb{X}

2.3 Quantized Q-learning

Based on the above theorems, we can find a near-optimal policy for our original MDP by quantizing our state and action spaces finely enough, and finding an optimal policy for the new MDP. However, we do not necessarily know that a Q-learning algorithm for this new MDP will converge, as the quantization process introduces some non-Markovian features into the MDP [7]. A result from [7] does in fact guarantee convergence of a Q-learning algorithm (to the finite model described in the previous section) under slightly different assumptions. The algorithm is exactly as in (4), but using the quantized states, that is:

$$\begin{aligned}Q_{t+1}(q(x), u) &= (1 - \alpha_t(q(x), u))Q_t(q(x), u) + \\ &\quad \alpha_t(q(x), u)(c(x, u) + \beta \min_{v \in \mathbb{U}} Q_t(q(X_{t+1}), v))\end{aligned}\tag{6}$$

Assumption 2.3. In the above Q-learning algorithm, we have:

1.

$$\alpha_t(y, u) = \begin{cases} \frac{1}{1 + \sum_{k=0}^t 1_{(Y_k, U_k) = (y, u)}} & (Y_t, U_t) = (y, u) \\ 0 & \text{otherwise} \end{cases}$$

2. The policy γ chooses control actions independently of everything and randomly, i.e.

$$\Pr(\gamma(\cdot) = u_i) = p_i \quad \forall i = 1, \dots, |\mathbb{U}|$$

where $p_i > 0 \forall i$ and $\sum_i p_i = 1$.

3. Under the above policy γ , the state process $\{X_t\}$ admits a unique invariant measure ψ^* .

Theorem 2.4. [7, Theorem 3.2] Under Assumption 2.3, for each $(y_i, u) \in \mathbb{Y} \times \mathbb{U}$, the algorithm in (6) converges to:

$$Q^*(y_i, u) = \hat{c}(y_i, u) + \beta \sum_{y_j \in \mathbb{Y}} \hat{P}(y_j | y_i, u) \min_{v \in \mathbb{U}} Q^*(y_j, v),$$

where \hat{c} and \hat{P} are defined as in (5) with $\psi = \psi^*$ (i.e. they are defined using the unique invariant measure of $\{X_t\}$).

We remark briefly on the connection of quantized Q-learning to our original zero-delay coding problem:

Remark 2.

1. We were able to reduce the zero-delay coding problem to an MDP in Section 1, but were unable to run effective algorithms on it due to the complexity of the system dynamics. Moreover, the new MDP had an infinite state space, and thus it was impossible to use standard Q-learning on it (recall that this MDP had π_t , a probability distribution, as its state).
2. Given the above theorems, we can use a quantized Q-learning algorithm to find a near-optimal policy for this MDP, and then apply the resulting policy to the original zero-delay coding problem. We just need to confirm that Assumption 2.2 and Assumption 2.3 hold.
3. Indeed, from Lemma 1.5 and our setup of the MDP, we already have that Assumption 2.2 holds. Parts 1 and 2 of Assumption 2.3 are determined by algorithm design, and so can be met. We are thus left with proving that part 3 is met, i.e. that the process $\{\pi_t\}$ admits a unique invariant measure under a random exploration policy. The next section is dedicated to showing this fact.

3 Unique Ergodicity Under a Memoryless Exploration Policy

Recall our setup from Section 1. In particular, we have a controlled Markov process $\{\pi_t\}$, with control $\{Q_t\}$, where $Q_t : \mathbb{X} \rightarrow \mathcal{M}$. Here \mathbb{X} is our source alphabet, \mathcal{M} is our message set, and we have $Q_t(X_t) = q_t$.

We wish to show that if we choose the Q_t randomly, $\{\pi_t\}$ admits a unique invariant measure. Note that under a randomized policy, we can view a given quantizer output $q_t \in \mathcal{M}$ as an observation of the true state X_t , dependent on an i.i.d. noise variable Z_t . That is, we let $Q = \{Q_1, \dots, Q_m\}$ (m is finite since \mathbb{X} is finite). Then let $q_t = h(x_t, z_t) = Q_{z_t}(x_t)$, where Z_t is an i.i.d random variable taking values in $\mathcal{Z} := \{1, \dots, m\}$, with $Z_0 \sim R$ positive everywhere. Then we can use some tools from the literature of Partially Observed Markov Processes (POMPs) to determine the ergodicity of $\{\pi_t\}$.

Predictor and Filter Merging

Given the above characterization, we can view π_t as what is known as a *predictor* in the POMDP literature. Recall the recursion equation (1) and note that this is dependent on the initialization of π_0 , also called the *prior*. We denote the predictor process resulting from the prior ν as $\{\pi_t^\nu\}_{t \geq 0}$. A common question in the POMDP literature is when measures such as the predictor are *stable*, which means that the process eventually forgets an incorrect prior, $\nu \neq \mu$. More formally, we introduce the following definitions:

Definition 3.1. Two sequences of probability measures $\{A_t\}_{t \geq 0}$ and $\{B_t\}_{t \geq 0}$ merge weakly if $\forall f \in C_b(\mathbb{X})$, we have $\lim_{t \rightarrow \infty} |\int f dA_t - \int f dB_t| = 0$.

Definition 3.2. For two probability measures A and B , the total variation norm is given by $\|A - B\|_{TV} = \sup_{\|f\|_\infty \leq 1} |\int f dA - \int f dB|$ for f measurable.

Definition 3.3. A predictor process $\{\pi_t\}$ is stable in the sense of weak merging in expectation if for any $f \in C_b(\mathbb{X})$ and any prior ν with $\mu \ll \nu$, we have $\lim_{n \rightarrow \infty} E^\mu[|\int f d\pi_t^\mu - \int f d\pi_t^\nu|] = 0$.

Definition 3.4. A predictor process $\{\pi_t\}$ is stable in the sense of total variation in expectation if for any prior ν with $\mu \ll \nu$, we have $\lim_{n \rightarrow \infty} E^\mu[\|\pi_t^\mu - \pi_t^\nu\|_{TV}] = 0$.

Note that merging (respectively, stability) in total variation implies merging (respectively, stability) weakly since the continuous and bounded functions $C_b(\mathbb{X})$ are a subset of the measurable and bounded functions.

The above definitions are of interest due to a result from [9, Theorem 2] relating stability to ergodicity. We state and prove a slightly modified version of this result here, adapted for our setup.

Theorem 3.1. Assume that there exists a unique invariant measure $\zeta(dx, dq)$ for the Markov process $\{X_t, q_t\}_{t \geq 0}$ and that the predictor process $\{\pi_t^\mu\}_{t \geq 0}$ is stable in the sense of Definition 3.3. Then there is at most one invariant measure for the joint Markov process $\{X_t, q_t, \pi_t^\nu\}_{t \geq 0}$ for any prior ν .

Proof. Throughout, we use the notation $\nu(f) := \int f d\nu$. Assume that $m_1, m_2 \in \mathcal{P}(\mathbb{X} \times \mathcal{M} \times \mathcal{P}(\mathbb{X}))$ are two invariant measures for the joint process $\{X_t, q_t, \pi_t^\nu\}$. Then their projections on $\mathbb{X} \times \mathcal{M}$ are invariant for $\{X_t, q_t\}_{t \geq 0}$. Then, by unique invariance of $\zeta(dx, dq)$ we have

$$m_i(dx, dq, d\nu) = P_{m_i}(d\nu|x, q)\zeta(dx, dq)$$

Then we show that $m_1(F) = m_2(F)$ for each F on a set of measure-determining functions [9], namely those s.t. $F(x, q, \nu) = \phi(x, q)H(\nu(\phi_1), \dots, \nu(\phi_l))$, where $\phi \in C(\mathbb{X} \times \mathcal{M})$, $\phi_1, \dots, \phi_l \in C(\mathbb{X})$, H is bounded and Lipschitz continuous with constant L_H , and $l \in \mathbb{N}$.

Let S be the transition operator associated with the process $\{X_t, q_t, \pi_t^\nu\}$. Then by invariance we have for $i = 1, 2$:

$$m_i(F) = \int_{\mathbb{X} \times \mathcal{M} \times \mathcal{P}(\mathbb{X})} \frac{1}{n} \sum_{j=0}^{n-1} S^j F(x, q, \nu) P_{m_i}(d\nu|x, q) \zeta(dx, dq)$$

And thus,

$$\begin{aligned} & |m_1(F) - m_2(F)| \\ & \leq \int_{\mathbb{X} \times \mathcal{M} \times \mathcal{P}(\mathbb{X}) \times \mathcal{P}(\mathbb{X})} \frac{1}{n} \sum_{j=0}^{n-1} |S^j F(x, q, \nu_1) - S^j F(x, q, \nu_2)| P_{m_1}(x, q, \nu_1) P_{m_2}(x, q, \nu_2) \zeta(dx, dq) \\ & \leq L_H \|\phi\| \int_{\mathbb{X} \times \mathcal{M} \times \mathcal{P}(\mathbb{X}) \times \mathcal{P}(\mathbb{X})} \frac{1}{n} \sum_{j=0}^{n-1} E^\mu \left[\sum_{i=1}^l |\pi_j^{\nu_1}(\phi_i) - \pi_j^{\nu_2}(\phi_i)| \right] P_{m_1}(x, y, \nu_1) P_{m_2}(x, q, \nu_2) \zeta(dx, dq) \end{aligned}$$

Since the predictors are stable in the sense of Definition 3.3, and by the dominated convergence theorem, the last line converges to zero as $n \rightarrow \infty$. \square

Note that the above result concerns the joint process $\{X_t, q_t, \pi_t\}_{t \geq 0}$. The following theorem from [9] extends this to $\{q_t, \pi_t\}_{t \geq 0}$.

Theorem 3.2. [9, Theorem 3] *If the joint process $\{X_t, q_t, \pi_t\}_{t \geq 0}$ is Feller and admits at most one invariant measure, then $\{q_t, \pi_t\}_{t \geq 0}$ admits at most one invariant measure.*

Note that the above Feller assumption is trivially satisfied in the case where \mathbb{X} and \mathcal{M} are finite, since $\{\pi_t\}_{t \geq 0}$ was already shown to be Feller in [1]. We have the following result concerning $\{\pi_t\}_{t \geq 0}$.

Theorem 3.3. *If the joint process $\{q_t, \pi_t\}_{t \geq 0}$ admits at most one invariant measure, then $\{\pi_t\}_{t \geq 0}$ admits at most one invariant measure.*

Proof. Let $m(dq, d\pi)$ be the unique invariant measure for $\{q_t, \pi_t\}$. Then we have,

$$\begin{aligned} m(dq, d\pi) &= P(dq|\pi)P(d\pi) \\ &= P(x \in Q^{-1}(dq)|\pi)P(d\pi) \\ &= \pi(Q^{-1}(dq))P(d\pi) \end{aligned}$$

Now, $\pi(Q^{-1}(dq))$ (i.e. the pushforward measure of π under Q) depends only on the process $\{q_t, \pi_t\}$ and the choice of Q_t . But Q_t is chosen i.i.d. and randomly, and thus $m(dq, d\pi)$ induces a unique pushforward measure $\pi(Q^{-1}(dq))$. Therefore $P(d\pi)$ is a unique invariant measure for $\{\pi_t\}$. \square

Predictor Stability

In light of Theorem 3.1, we want to show that our predictor process $\{\pi_t\}_{t \geq 0}$ is stable in the sense of Definition 3.3. While stability (and even unique ergodicity) of the predictor have been studied extensively (see e.g. [10], [11] for discussions), the assumptions of these theorems are in general quite strong and not applicable for large classes of sources. For example, they may require that the state has a uniformly positive transition density, etc. [10].

In order to generalize our sources and quantizers as much as possible, we will utilize results in [12], which relate the stability of the predictor to stability of the *filter*. The filter has the same form as the predictor but is further conditioned on q_t (that is, $\pi_t^* = P(x_t|q_{[0,t]})$). We have the following result.

Lemma 3.4. [12, Theorem 2.11] *The filter merges in total variation in expectation if and only if the predictor merges in total variation in expectation.*

And finally, we present a result to prove that the filter is stable in total variation in expectation. Recall that we defined $q_t = h(x_t, z_t) = Q_{z_t}(x_t)$, where Q_{z_t} was our random quantizer choice. Then we denote $h_x(\cdot) := h(x, \cdot)$. The following lemma is originally from [13], which was presented in a more general context.

Lemma 3.5. [13, Corollary 5.5] *Consider the conditional probability measure defined by $P(q|x) = R(h_x^{-1}(q))$. If this is strictly positive for all x, q , then the filter converges in total variation in expectation.*

To show that this density is positive in the quantizer case, recall that under a memoryless exploration policy we have that R is positive everywhere. Then, since \mathcal{Q} contains every possible quantizer $Q : \mathbb{X} \rightarrow \mathcal{M}$, we have that $\forall(x, q)$, there exists at least one quantizer s.t. $Q(x) = q$, and thus $h_x^{-1}(q)$ is nonempty $\forall(x, q)$. Then, since R is positive everywhere, we obtain by Lemma 3.5 that the filter is stable in total variation in expectation.

Note that the above arguments also hold as long as, $\forall(x, q)$, our set of quantizers contains *at least one* quantizer Q st $Q(x) = q$, and therefore we could operate with a reduced set of quantizers. More concretely, we impose the following assumption on our set of quantizers \mathcal{Q} .

Assumption 3.1. *Let $\bar{\mathcal{Q}}_{xq} = \{Q \in \mathcal{Q} : Q(x) = q\}$. Then $\forall x \in \mathbb{X}$ and $\forall q \in \mathcal{M}$ we have $\bar{\mathcal{Q}}_{xq} \neq \emptyset$.*

To summarize, we have the following theorem.

Theorem 3.6. *If the set of quantizers \mathcal{Q} satisfies Assumption 3.1, then there exists a memoryless exploration policy such that the Markov process $\{\pi_t\}_{t \geq 0}$ admits a unique invariant probability measure.*

Proof. First, under Assumption 3.1 and by Lemma 3.5, the filter is stable in total variation.

Then, by Lemma 3.4, the predictor is stable in total variation in expectation, which implies weak stability in expectation, and so by Theorems 3.1, 3.2, and 3.3, we have that $\{\pi_t\}_{t \geq 0}$ admits at most one invariant measure.

Finally, the existence of an invariant measure for $\{\pi_t\}$ is guaranteed since $\{\pi_t\}$ is weak Feller (see Lemma 1.5) on a compact space $\mathcal{P}(\mathbb{X})$. \square

In light of Theorem 3.6 we now have that the quantized Q-learning results in [7] are applicable. We now present some relevant algorithms.

4 Algorithms

4.1 Quantizing π_t

Since the state space \mathbb{X} is finite, say with $|\mathbb{X}| = m$, then $\mathcal{P}(\mathbb{X})$ is a simplex in \mathbb{R}^m . For a given belief π_t and n , we wish to find the nearest (in terms of Euclidean distance) $\hat{\pi}_t = [\frac{k_1}{n}, \dots, \frac{k_m}{n}]$, where $k_i \in \mathbb{Z}$. Then we can use the algorithm in e.g. [14], [15] to quantize π_t as follows.

Algorithm 1: Predictor Quantization

```

1  Set Parameters
2     $\pi_t = (p_1, \dots, p_m)$ 
3     $n$ 
4    for  $i = 1, \dots, m$ 
5       $k'_i = \lfloor np_i + \frac{1}{2} \rfloor$ 
6    end
7    Set  $n' = \sum_i k'_i$ 
8    if  $n = n'$ 
9      Return  $(\frac{k'_1}{n}, \dots, \frac{k'_m}{n})$ 
10   else
11     for  $i = 1, \dots, m$ 
12        $\delta_i = k'_i - np_i$ 
13     end
14     Sort  $\delta_i$  s.t.  $\delta_{i_1} \leq \dots \leq \delta_{i_m}$ 
15     Set  $\Delta = n' - n$ 
16     if  $\Delta > 0$ 
17        $k_{i_j} = \begin{cases} k'_{i_j} & j = 1, \dots, m - \Delta \\ k'_{i_j} - 1 & j = m - \Delta + 1, \dots, m \end{cases}$ 
18     else
19        $k_{i_j} = \begin{cases} k'_{i_j} + 1 & j = 1, \dots, |\Delta| \\ k'_{i_j} & j = |\Delta| + 1, \dots, m \end{cases}$ 
20     Return  $(\frac{k_1}{n}, \dots, \frac{k_m}{n})$ 
```

We have the following lemma regarding the radius of these quantization bins under the above algorithm.

Lemma 4.1. [14, Proposition 2] *The maximum radius of the quantization regions for π_t under the L_∞ norm is given by*

$$b_\infty = \frac{1}{n}(1 - \frac{1}{m})$$

Also note that the number of bins for π_t when using **Algorithm 1** is related to n by the following relation:
 $\# \text{ bins} = \binom{n+m-1}{m-1}$ [14].

4.2 Quantized Q-learning

Using the above algorithm to quantize π_t , we have the following algorithm for quantized Q-learning.

Algorithm 2: Quantized Q-learning

- 1 **Set Parameters**
- 2 State space (\mathbb{X}) and state transition kernel ($P(x_{t+1}|x_t)$)
- 3 Distribution of X_0 (π_0)
- 4 Parameter for quantization of π_t (n)
- 5 Set of quantizers (\mathcal{Q})
- 6 **Initialize** arbitrary Q-table of size $\binom{n+m-1}{m-1} \times |\mathcal{Q}|$
- 7 **Initialize** state x_0 according to π_0
- 8 **Quantize** π_0 according to **Algorithm 1**, call this $\hat{\pi}_0$
- 9 Select quantizer Q_0 according to randomized exploration policy γ
- 10 **Quantize** x_0 according to $q_0 = Q_0(x_0)$
- 11 **for** $t = 0 \dots T - 1$
- 12 Receive cost of quantization according to $c(\pi_t, Q_t)$ (see (2))
- 13 Receive x_{t+1} according to $P(x_{t+1}|x_t)$
- 14 Receive π_{t+1} according to update equation (1)
- 15 Quantize π_{t+1} according to **Algorithm 1**, call this $\hat{\pi}_{t+1}$
- 16 Update Q-table using (6)
- 17 Select quantizer Q_{t+1} according to randomized exploration policy γ
- 18 Quantize x_{t+1} according to $q_{t+1} = Q_{t+1}(x_{t+1})$
- 19 **end**

Theorem 4.2. *Under Assumption 3.1 and as $n \rightarrow \infty$, the above algorithm gives a near-optimal policy for the zero-delay coding problem.*

Proof. First, Assumption 3.1 allows us to use Theorem 3.6 to guarantee a unique invariant measure for $\{\pi_t\}$ under a random exploration policy. Therefore Assumption 2.3 is met. Then from Theorem 2.4, this algorithm converges. Furthermore, as $n \rightarrow \infty$, by Lemma 4.1 we get $\|L^-\|_\infty \rightarrow 0$. Then by Theorem 2.3, the policy we get from this algorithm performs near-optimally when applied to the original MDP. By our discussions in Section 1, this policy is then near-optimal for the zero-delay coding problem. \square

5 Examples

For all the following, we will use MSE as our distortion measure, i.e. $d(x, \hat{x}) = (x - \hat{x})^2$, and a high discount factor (0.9999). We approximate the long-term discounted cost by calculating a finite-horizon discounted cost with $T = 10^6$.

5.1 Effects of Increasing n for a Fixed Rate

Say we consider all the two-cell quantizers (i.e. $\mathcal{M} := \{1, 2\}$) on $\mathbb{X} = \{1, \dots, 5\}$ (i.e. $m = 5$) with a randomly generated transition matrix. We let n in the quantization of π_t vary, and run **Algorithm 2** for these different values of n . Recall that the number of bins in this quantization is given by $\binom{n+m-1}{m-1}$. The first graph below shows the performance gain as we increase the number of bins, while the second shows the convergence of the algorithm for different numbers of bins. Each line in the second graph corresponds to increasing the value of n by 1.

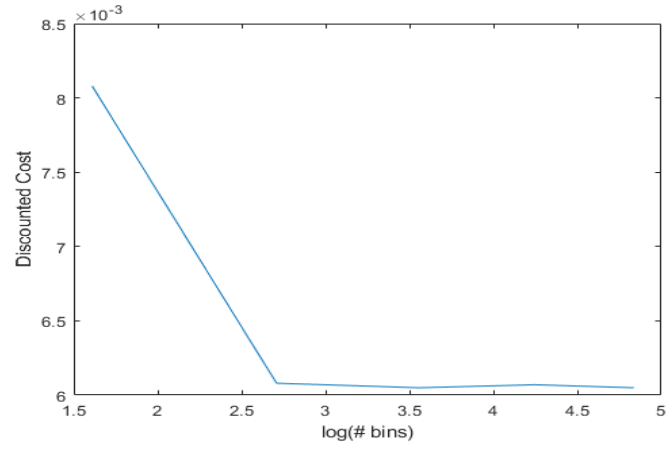


Figure 1: Long-term discounted cost of learned policies

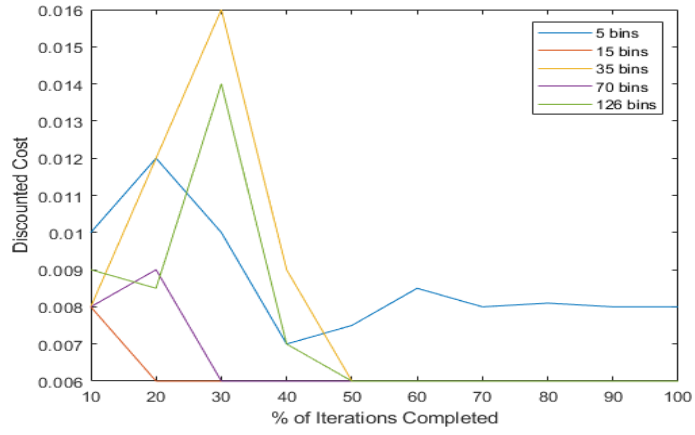


Figure 2: Convergece of learned policies

Note that the quantization gains are not significant after $n = 2$ (which corresponds to 15 bins), which indicates that this is a sufficient quantization level for a near-optimal policy.

Similarly, for 2-cell quantizers on $\mathbb{X} = \{1, \dots, 16\}$,

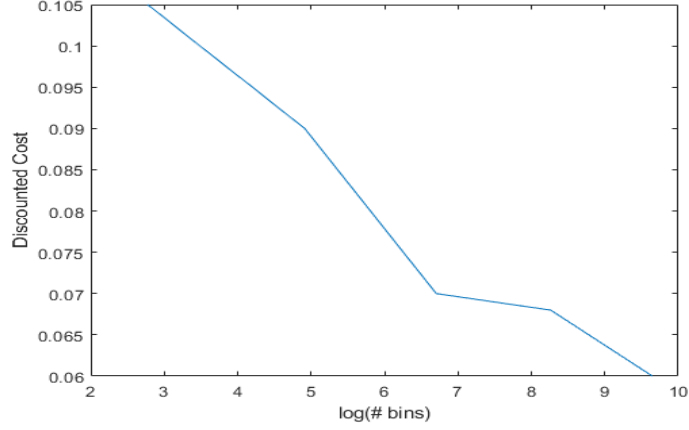


Figure 3: Long-term discounted cost of learned policies

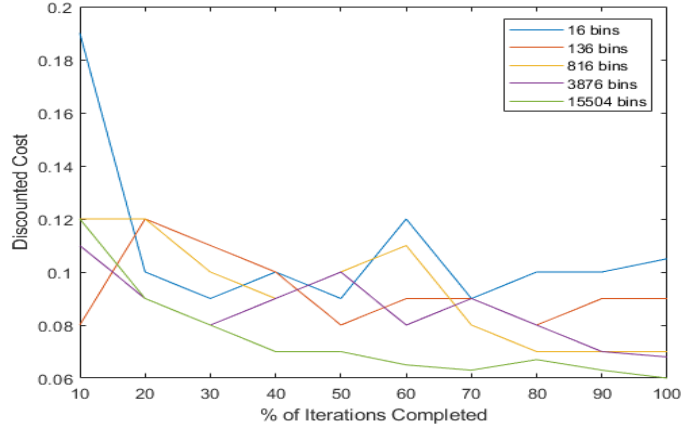


Figure 4: Convergece of learned policies

In this example, additional quantization may be required to get closer to the optimal policy.

5.2 Comparison to Lloyd-Max Quantizer

In these simulations, we plot the distortion for different values of n and for different quantizer rates (i.e. different sizes of \mathcal{M}). We also plot the comparison with a Lloyd-Max quantizer (another common algorithm to find an optimal quantizer), and note that our algorithm results in a lower discounted cost. We use the source alphabet $\mathbb{X} = \{1, \dots, 13\}$, and the rate is calculated by $\log_2(|\mathcal{M}|)$.

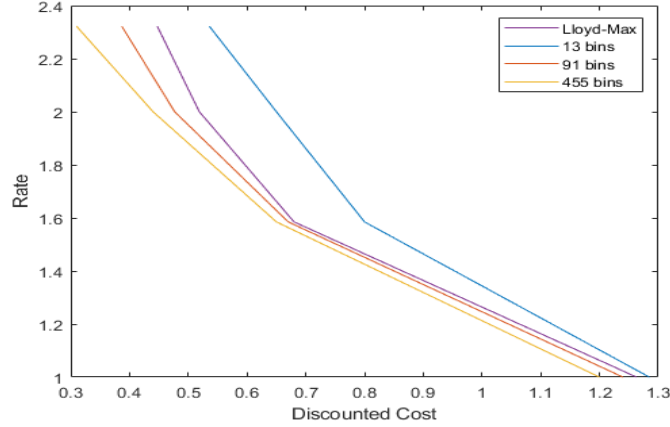


Figure 5: Comparison with Lloyd-Max in Markov case

Note that for coarse quantizations, a Lloyd-Max quantizer may perform better, but as we increase the quantization, the Q-learning approach gives a lower distortion for a given rate.

Finally, we also note that our algorithm matches very closely with a Lloyd-Max quantizer in the case where the source is i.i.d., as can be seen below. Note that in this case, the Q-learning algorithm only visits one state (the one given by quantizing the distribution of X_0), and hence raising n provides no performance gain. Here we use $\mathbb{X} = \{1, \dots, 8\}$.

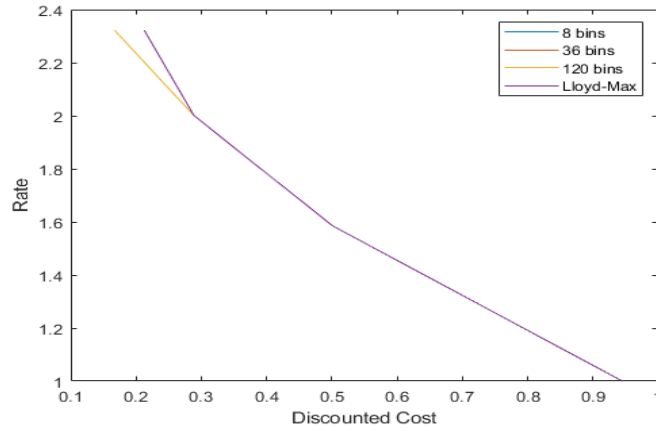


Figure 6: Comparison with Lloyd-Max in i.i.d. case

We now remark on the algorithm performance and feasibility:

Remark 3.

1. As previously mentioned, the number of bins grows quickly with n , and so running the algorithm with high n requires high amounts of memory in order to store a large Q-table. However, the number of actual visited $\hat{\pi}_t$ tends to be much lower than the total number of bins, and so it may be possible to

increase the efficiency of this algorithm by quantizing π_t in a non-uniform fashion (the results from [7] used for convergence of the algorithm allow a non-uniform quantization).

2. As the number of bins grows, the time for the algorithm to converge becomes much higher, since it must visit more states. One must trade much longer algorithm runtime for potential performance increases. Also, since we only have weak continuity of the transition kernel for $\{\pi_t\}$, we do not have a bound on what quantization level we need to obtain a given performance (results from [7] provide these bounds under stronger notions of continuity). Therefore it is somewhat trial-and-error to find the appropriate quantization level for a given application.
3. As mentioned in Section 1, $\{\pi_t\}$ updates in a Markovian fashion, and so we do not need to store the previous values of π_t . Furthermore, from Section 3 we have that the process $\{\pi_t\}$ is stable (i.e. it “forgets” an incorrect prior) under the random exploration policy used in learning. Therefore we do not need to worry about errors accumulating over time, and in fact we can start the algorithm from any distribution π_0 (the choice to initialize to the stationary distribution in **Algorithm 2** is just for convenience).

6 Future Work

We note that the above Q-learning algorithm only converges when $\beta \in (0, 1)$ (i.e. for the discounted cost problem) and therefore is not applicable for the average cost problem, which is generally what we are interested in for source coding. It may be possible to adapt the Q-learning algorithm to solve the average cost problem (see e.g. [16]), but this would require an extension of the results in [7] to this algorithm, which may require additional constraints on the source \mathbb{X} . Nevertheless, we can obtain a good approximation to the average cost problem by taking β to be close enough to 1 (such an approach is called the “vanishing discount” method, see e.g. [6, Chapter 5]). The necessary assumptions for this approach hold, at least when the source is finite, so we can still obtain near-optimality for the average cost problem.

Furthermore, we would like to extend the algorithm to the case when \mathbb{X} is continuous. The primary obstacle here would be that the number of possible quantizers is infinite, and so a method of quantizing the space of quantizers would need to be developed. We note however that quantizing the action space (in this context, the set of quantizers) is considered in the results from [7], and also all of the predictor stability/unique invariance arguments in Section 3 hold when \mathbb{X} is continuous. So with some minor alterations, the algorithm will still converge in the continuous case. As mentioned above, one would still have to show that this gives a near-optimal policy for the average cost problem, using the vanishing discount (or similar) method.

Finally, we wish to consider the case when the channel over which q_t is sent is noisy. While most of the results from Section 1 carry over in this case, additional considerations will have to be taken in other sections (e.g. in proving uniqueness of the invariant measure).

References

- [1] T. Linder and S. Yüksel, “On optimal zero-delay coding of vector markov sources,” *IEEE Trans. Inf. Theory*, vol. 60, no. 10, pp. 5975–5991, Oct. 2014.
- [2] H. Witsenhausen, “On the structure of real-time source coders,” *Bell System Technical Journal*, vol. 58, no. 6, pp. 1437–1451, 1979.
- [3] J. Walrand and P. Varaiya, “Optimal causal coding-decoding problems,” *IEEE Trans. Inf. Theory*, vol. IT-29, no. 6, pp. 814–820, Nov. 1983.
- [4] R. Wood, T. Linder, and S. Yüksel, “Optimal zero delay coding of markov sources: Stationary and finite memory codes,” *IEEE Trans. Inf. Theory*, vol. 63, no. 9, pp. 5968–5980, Sep. 2017.

- [5] S. Yüksel and T. Linder, “Optimization and convergence of observation channels in stochastic control,” *SIAM J. on Control and Optimization*, vol. 50, pp. 864–887, 2012.
- [6] O. Hernandez-Lerma and J. Lasserre, *Discrete-time Markov Control Processes*. Springer, 1996.
- [7] A. Kara, N. Saldi, and S. Yüksel, “Q-learning for MDPs with general spaces: Convergence and near optimality via quantization under weak continuity,” 2021. DOI: 10.48550/ARXIV.2111.06781.
- [8] N. Saldi, T. Linder, and S. Yüksel, *Finite Approximations in Discrete-Time Stochastic Control: Quantized Models and Asymptotic Optimality*. Springer, Cham, 2018.
- [9] G. D. Masi and Ł. Stettner, “Ergodicity of hidden markov models,” *Math. Control Signals Syst.*, vol. 17, no. 4, pp. 269–296, Oct. 2005.
- [10] P. Chigansky, R. Liptser, and R. van Handel, “Intrinsic methods in filter stability,” *Handbook of Non-linear Filtering*, Aug. 2009.
- [11] R. Douc and C. Matias, “Asymptotics of the maximum likelihood estimator for general hidden markov models,” *Bernoulli*, vol. 7, no. 3, pp. 381–420, 2001.
- [12] C. McDonald and S. Yüksel, “Converse results on filter stability criteria and stochastic non-linear observability,” 2018. DOI: 10.48550/ARXIV.1812.01772.
- [13] R. van Handel, “The stability of conditional markov processes and markov chains in random environments,” *Ann. Probab.*, vol. 37, no. 5, pp. 1876–1925, Sep. 2009.
- [14] Y. Reznik, “An algorithm for quantization of discrete probability distributions,” *DCC 2011*, pp. 333–342, Mar. 2011.
- [15] N. Saldi, S. Yüksel, and T. Linder, “Asymptotic optimality of finite model approximations for partially observed markov decision processes with discounted cost,” *IEEE Transactions on Automatic Control*, vol. 65, no. 1, pp. 130–142, 2020.
- [16] J. Abounadi, D. Bertsekas, and V. Borkar, “Learning algorithms for markov decision processes with average cost,” *SIAM Journal on Control and Optimization*, vol. 40, no. 3, pp. 681–698, 2001.