A Universal Scheme for Learning

Vivek F. Farias*, Ciamac C. Moallemi*, Benjamin Van Roy*†, and Tsachy Weissman*
* Electrical Engineering, Stanford University, Stanford, CA 94305 USA Emails: {vivekf,ciamac,bvr,tsachy}@stanford.edu
† Management Science & Engineering, Stanford University Stanford, CA 94305 USA

***Abstract*— We consider the problem of optimal control of a Kth order Markov process so as to minimize long-term average cost, a framework with many applications in communications and beyond. Specifically, we wish to do so without knowledge of either the transition kernel or even the order K. We develop and analyze two algorithms, based on the Lempel-Ziv scheme for data compression, that maintain probability estimates along variable length contexts. We establish that eventually, with probability 1, the optimal action is taken at each context. Further, in the case of the second algorithm, we establish almost sure asymptotic optimality.**

I. Introduction

A large number of practically important control problems may be formulated as stochastic control problems. Very broadly, such problems require that one control some stochastic process so as to optimize an objective. In this work, we restrict our attention to Markov Decisions Problems, specifically the control of Kth order Markov processes, where we seek to optimize a cost function that decomposes linearly over time. Other than a knowledge of the function being optimized, the algorithms we study do not require knowing either the value of K, or the dynamics of the underlying process a priori. Such a setting is extremely general and applies to a broad variety of problems that arise in communications; we will subsequently present such an example. We begin, however, by clarifying our model:

Consider a system consisting of observations $\{X_t\}$ and actions $\{A_t\}$ that evolve as a random process over time. Observations and actions take values in the finite sets $\mathbb{X}$ and $\mathbb{A}$, respectively. Denote by $\mathcal{F}_t$ the σ-algebra generated by (X^t, A^t). We assume that

$$\Pr\left(X_t = x_t | \mathcal{F}_{t-1}\right) = P(x_t | X_{t-K}^{t-1}, A_{t-K}^{t-1}).$$

In other words, at time $t-1$, the next observation X_t is Markovian with transition kernel P, given the last K observations and actions.

A policy μ is a sequence of mappings $\{\mu_t\}$, where for each t the map $\mu_t : \mathbb{X}^t \times \mathbb{A}^{t-1} \to \mathbb{A}$ determines which action shall be chosen at time t given the history of observations and actions observed up to time t. In other words, under policy μ, actions will evolve according to the rule $A_t = \mu_t(X^t, A^{t-1})$. We will call a policy stationary, if there exists $T > 0$ and $L > 0$ so that

$$\mu_t(X^t, A^{t-1}) = \mu(X_{t-L}^t, A_{t-L}^{t-1}), \quad \forall\ t \geq T.$$

Define a cost function $g : \mathbb{X}^K \times \mathbb{A}^K \to [0, g_{\max}]$. Given a policy μ, we can define the long-term expected cost as

$$\limsup_{T\to\infty} \mathbb{E}_\mu \left[\frac{1}{T}\sum_{t=1}^{T} g(X_{t-K+1}^t, A_{t-K+1}^t)\right]. \tag{1}$$

We wish to discover the policy μ that minimizes this long-term expected cost. We will make the following technical assumption.

Assumption 1: There is a unique optimal stationary policy μ^*.

An important problem in joint source-channel coding may be cast in the above formalism.

Example 1: Let $\mathbb{X}$ and $\mathbb{Y}$ be source and channel alphabets, respectively. Consider transmitting a sequence of symbols $X^t \in \mathbb{X}^t$ across a memoryless channel. Let Y_t represent our choice of encoding at time t. Let $\hat{Y}_t$ represent the tth encoded symbol corrupted by the memoryless channel. For all t, let $d : \mathbb{Y}^L \to \mathbb{X}$ be some fixed decoder that decodes the tth symbol based on the past $L-1$ symbols and $\hat{Y}_t$. Given a single letter distortion measure $\rho : \mathbb{X} \times \mathbb{X} \to \mathbb{R}$ and letting

$$g(x^L, y^L) \triangleq \mathbb{E}\left[\rho\left(d(\hat{Y}^L), X_L\right) | X^L = x^L,\ Y^L = y^L\right],$$

we define as our optimization objective, finding a sequence of encoders $\mu_t : \mathbb{X}^t \to \mathbb{A}$ minimizing:

$$\limsup_{T\to\infty} \mathbb{E}_\mu \left[\frac{1}{T}\sum_{t=1}^{T} g(X_{t-L+1}^t, Y_{t-L+1}^t)\right].$$

Assuming the source to be Mth order Markov, and setting $K = \max(M, L)$, it is clear that the optimal coding problem at hand is amenable to our formulation with observation space $\mathbb{X}$ and action space $\mathbb{Y}$.

Versions of this problem have been considered in [1], [2]. In those works, the assumption of a known source, channel, and decoder translates into a priori knowledge of the transition kernel itself (and, of course, K). To contrast, our methods will require knowledge of the channel and decoder in order to evaluate the cost function g, but will not require knowledge of the source.

Various related problems have also been considered: [3] presents an asymptotically optimal algorithm for learning that can be applied when the transition kernel P is unknown but the order K is known. The algorithm relies on explicit exploration/exploitation stages and asymptotic optimality is guaranteed. Recently, [4] presented an algorithm for the problem of finding optimal Markov policies (in the class of all kth order policies for fixed k) for 'POMDPs' (processes wherein only some function of the state of the underlying Markov

process is observable). The guarantees presented there are in expectation and again asymptotic. Work presented in [5] considers an optimal control framework where the dynamics of the environment are not known and one wishes to select the best of a finite set of policies (or 'experts'). In contrast, our work can be thought of as one of competing with the set of all possible policies. The prediction problem for loss functions with memory and a Markov-modulated source considered in [6] is essentially a Markov Decision Problem as the authors point out; again, in this case, knowing the structure of the loss function implicitly gives the order of the underlying Markov process.

Our algorithms are inspired by the Lempel-Ziv algorithm for data compression [7]. This type of algorithm has been considered in a wide variety of contexts that also fall within our framework. In [8] (and references therein), a Lempel-Ziv inspired scheme is used for the prediction of unknown Markov sources of arbitrary order with memoryless loss functions. This problem is essentially again a special case of our formalism (wherein the choice of action has no impact on transitions). In [9], a Lempel-Ziv inspired scheme is used for the problem of cache prefetching. This problem, also, can be placed in our framework.

The remainder of this paper is organized as follows: In Section II we briefly discuss the classical solution to a Markov Decision Problem (that is, our problem, with full knowledge of the transition kernel). In Sections III and IV we present and analyze a candidate algorithm for universal learning. Section V discusses a modification of this algorithm that, at the cost of some extra exploration, achieves asymptotic optimality. We conclude with a discussion of interesting future directions.

II. Classical Solution

We assume from this point on that the cost function g depends on only the current observation. As we discuss in Section VI, the algorithms and results we present carry over to the case of a general cost function.

It is well known that given a discount factor $\alpha \in (0,1)$ sufficiently close to 1, the an policy μ^* that optimizes the discounted cost criteria

$$\mathbb{E}_\mu \left[\sum_{t=1}^{\infty} \alpha^{t-1} g(X_t) \right], \tag{2}$$

also optimized the long-term expected average cost. (See, for example, [10, Proposition 4.2.2].)

Noting that the process (X_t, A_t) is Markovian when the state space is augmented to include the past K observations and actions, the optimal policy can be discovered as follows. First, let $J^* : \mathbb{X}^K \times \mathbb{A}^{K-1} \to \mathbb{R}$ be the unique solution to Bellman's Equation,

$$J^*(x^K, a^{K-1}) = g(x_k) + \alpha \min_{a_K} \sum_{x_{K+1}} P(x_{K+1}|x^K, a^K) J^*(x_2^{K+1}, a_2^K). \tag{3}$$

J^* is referred to as the value function. Let μ^* be the policy that is greedy with respect to J^*. In other words, for $t \geq K$, set

$$\mu_t^*(x^t, a^{t-1}) = \operatorname*{argmin}_{a_t} \sum_{x_{t+1}} P(x_{t+1}|x_{t-K+1}^t, a_{t-K+1}^t) J^*(x_{t-K+1}^{t+1}, a_{t-K+2}^t).$$

Then, μ^* will be optimal under both the discounted and average cost criteria. Note, in particular, that $\mu_t^*(x^t, a^{t-1})$ depends only on $(x_{t-K+1}^t, a_{t-K+1}^{t-1})$.

III. A Universal Scheme

Solution of Bellman's Equation (3) requires knowledge of the transition kernel P. We would like to present an algorithm that requires no knowledge of P, or even of K. Inspired by the Lempel-Ziv algorithm for data compression [7], Algorithm 1 uses variable length contexts to dynamically adjust to the true order of the underlying process. Algorithm 1 proceeds as follows.

Time is parsed into intervals, or phrases, with the property that if the cth phrase covers time $\tau_c \leq t \leq \tau_{c+1} - 1$, then the observation/action sequence $(X_{\tau_c}^{\tau_{c+1}-1}, A_{\tau_c}^{\tau_{c+1}-2})$ will not have occurred as the prefix of any other phrase before time τ_c. At any point in time t, if the current phrase started at time τ_c, the context $(X_{\tau_c}^t, A_{\tau_c}^{t-1})$ is considered. Probability distributions for X_{t+1} given all choices of actions A_t are estimated using a Bayesian estimator based on a Dirichlet prior and past experience in the same context. Value function estimates are made by iterating the dynamic programming operator from (3) backwards over possible future contexts. Finally, a sequence of numbers $\{\gamma_e\}$ controls the probability that the algorithm attempts to exploit, that is take an optimal action at a context given current probability and value function estimates, or whether it chooses to explore, and take a random action in order to improve its estimates.

Note that Algorithm 1 can be implemented easily using a tree-like data structure. Nodes at depth ℓ correspond to contexts of the form $(x^\ell, a^{\ell-1})$ that have already been visited. Each such node can link to at most $|\mathbb{X}||\mathbb{A}|$ child nodes of the form $(x^{\ell+1}, a^\ell)$ at depth $\ell+1$. Each node maintains a count of how many times it has been seen as a context and maintains a value function estimate. Each phrase interval amounts to traversing a path from the root to a leaf, and adding an additional leaf. After each such path is traversed, the algorithm moves backwards along the path and updates only the counts and value function estimates along that path.

IV. Analysis

We start with a lemma. The proof of this lemma can be found in [11]. It follows in a straightforward fashion from the fact that the dynamic programming operator in (3) is a contraction mapping (see [10]).

Lemma 1: Under Algorithm 1, their exist constants $\bar{K} > 0$ and $\bar{\epsilon} > 0$ such that, for any context (x^s, a^{s-1}) with $s \geq K$, if all contexts (x^ℓ, a^ℓ) with $s \leq \ell \leq s + \bar{K}$ have been visited by time τ_c and for all ℓ with $s \leq \ell \leq s + \bar{K}$,

$$\|\hat{P}_c(\cdot|x^\ell, a^\ell) - P(\cdot|x_{\ell-K+1}^\ell, a_{\ell-K+1}^\ell)\|_1 \leq \bar{\epsilon},$$

Algorithm 1 A Lempel-Ziv inspired algorithm for learning.

$t \leftarrow 1$, $c \leftarrow 1$ {time and phrase indexes}
$\tau_c \leftarrow 1$ {start time of the cth phrase}
$N_c(\cdot) \leftarrow 0$ {context counts accumulated by time τ_c}
$\hat{P}_c(\cdot) \leftarrow 1/|\mathbb{X}|$ {transition probabilities estimated by time τ_c}
$\hat{J}_c(\cdot) \leftarrow 0$ {value function estimated by time τ_c}
for time t **do**
  observe X_t
  if $N_c(X_{\tau_c}^t, A_{\tau_c}^{t-1}) > 0$ **then** {are in a context that we have seen before?}
    with probability $\gamma_{N_c(X_{\tau_c}^t, A_{\tau_c}^{t-1})}$, pick A_t uniformly over $\mathbb{A}$ {explore}
    otherwise, pick A_t greedily according to $\hat{P}_c$, $\hat{J}_c$:

$$A_t \in \operatorname*{argmin}_{a_t} \sum_{x_{t+1}} \Big(\hat{P}(x_{t+1} | X_{\tau_c}^t, (A_{\tau_c}^{t-1}, a_t)) \hat{J}((X_{\tau_c}^t, x_{t+1}), (A_{\tau_c}^{t-1}, a_t)) \Big)$$

    {exploit}
  else {we are in a context not seen before}
    pick A_t uniformly over $\mathbb{A}$
    $N_{c+1}(\cdot) \leftarrow N_c(\cdot)$, $\hat{P}_{c+1}(\cdot) \leftarrow \hat{P}_c(\cdot)$, $\hat{J}_{c+1}(\cdot) \leftarrow \hat{J}_c(\cdot)$
    for s with $\tau_c \leq s \leq t$, in decreasing order **do**
      update context count

$$N_{c+1}(X_{\tau_c}^s, A_{\tau_c}^{s-1}) \leftarrow N_c(X_{\tau_c}^s, A_{\tau_c}^{s-1}) + 1$$

      update probability estimate

$$\hat{P}_{c+1}(x_s | x^{s-1}, a^{s-1}) \leftarrow \frac{N_{c+1}(x^s, a^{s-1}) + 1}{\sum_{x'} N_{c+1}((x^{s-1}, x'), a^{s-1}) + |\mathbb{X}|}$$

      update value function estimate

$$\hat{J}_{c+1}(X_{\tau_c}^s, A_{\tau_c}^{s-1}) \leftarrow g(X_s) + \alpha \min_{a_s} \sum_{x_{s+1}} \big(\hat{P}_{c+1}(x_{s+1} | X_{\tau_c}^s, (A_{\tau_c}^{s-1}, a_s)) \hat{J}_{c+1}((X_{\tau_c}^s, x_{s+1}), (A_{\tau_c}^{s-1}, a_s)) \big)$$

    end for
    $c \leftarrow c+1$, $\tau_c \leftarrow t+1$ {start the next phrase}
  end if
  $t \leftarrow t+1$
end for

then the action selected by acting greedily with respect to $\hat{P}_c$ and $\hat{J}_c$ at the context (x^s, a^{s-1}) is optimal.

Lemma 1 suggests that if all the estimated distributions up to $\bar{K}$ levels below a given context are $\bar{\epsilon}$ accurate, then an optimal decision can be made at a particular context. By the Strong Law of Large Numbers, if a context is visited infinitely often and explores at that context infinitely often, it will eventually have $\bar{\epsilon}$ accurate probability estimates almost surely. To ensure that this happens, we require two assumptions.

Assumption 2: There exists $p_{\min} > 0$ so that

$$P(x_{K+1} | x^K, a^K) > p_{\min},$$

for all sequences $x^{K+1} \in \mathbb{X}^{K+1}$, $a^K \in \mathbb{A}^K$.

Assumption 3: The sequence $\{\gamma_e\}$ satisfies $\gamma_e \downarrow 0$ as $e \to \infty$, and $\sum_{e=1}^{\infty} \gamma_e = \infty$.

With these assumptions, we can establish the following lemma. The proof relies on induction on the context length and the Second Borel-Cantelli Lemma. It can be found in [11].

Lemma 2: Under Algorithm 1, every context is visited infinitely often with probability 1.

Define $\tau_e^{(x^s, a^{s-1})}$ to be the time of the eth visit to the context (x^s, a^{s-1}). Using our two lemmas, we can immediately reach the following result, whose proof can be found in [11].

Theorem 1: Under Algorithm 1, for any context (x^s, a^{s-1}) with $s \geq K$,

$$\lim_{e \to \infty} \frac{1}{e} \sum_{i=1}^{e} \mathrm{I} \left\{ A_{\tau_i^{(x^s, a^{s-1})}} \neq \mu^*(x_{s-K+1}^s, a_{s-K+1}^{s-1}) \right\} = 0,$$

with probability 1. In other words, the fraction of time that a non-optimal decision is made at a context tends to 0 almost surely.

V. Limiting Context Length

The analysis provided by Theorem 1 is incomplete. It tells us that the fraction of time a non-optimal decision is made at a single context tends to zero almost surely. We would like to show, however, that the asymptotic average cost achieved by our algorithm tends to that of the optimal policy. For this, we need that the fraction of time non-optimal decisions are made to tend to zero. To see the distinction, note that although we might be performing better and better over time at any given context, as the depth of the tree increases we may be introducing many new contexts that have not been visited many times and, hence, at which we make non-optimal decisions.

In particular, imagine the state of the algorithm after some long interval of time. Short contexts, specifically those with length less than K, are not meaningful because they are not estimating real transition probabilities. Referring to the caricature in Figure 1, this is the upper-most triangle. Contexts of moderate length will have been visited many times and are likely to have estimates leading to optimal decisions—these are the mid-section of the tree in Figure 1. Contexts that are very long, however, will not have been visited many times and are likely to be non-optimal. We need to control the amount

of time the algorithm spends in such contexts and insure that it is vanishing. That is, we need to control the size of the bottom-most section of the tree in Figure 1. We present here an algorithm that maintains a size of at most $O(\log d)$ for this bottom-most section, where d is the depth of the tree.

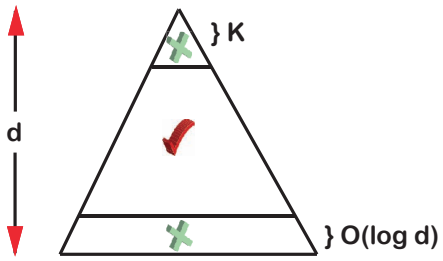

Fig. 1. Context tree for Algorithm 2

In Algorithm 2, at any given time, a maximum context length d is maintained, and d is only incremented whenever every context of length d has been visited at least once. At such points in time, the algorithm also updates transition probability and value function estimates. Note that the analyses of Lemma 1, Lemma 2, and Theorem 1 continue to hold for this modified algorithm. In particular, since every node will be visited infinitely often, it is clear that $d \uparrow \infty$ almost surely.

Note that the design of Algorithm 2 guarantees that the number of visits to a fixed context (x^s, a^{s-1}) grows exponentially as the depth of the tree is increased. In particular, if the depth of the tree is d, then every context of depth $d-1$ must have been visited at least once. Since a fixed context with length $s < d$ has $(|\mathbb{X}||\mathbb{A}|)^{d-s-1}$ such descendants, it will have been visited at least $(|\mathbb{X}||\mathbb{A}|)^{d-s-1}$ times. This fact allows us to prove the following lemma.

Lemma 3: Let $C > 1$ be a constant. Under Algorithm 2, there exists a random variable D such that $D < \infty$ with probability 1 and if the $d \geq D$,

$$\|\hat{P}_d(\cdot|x^\ell, a^\ell) - P(\cdot|x^\ell_{\ell-K+1}, a^\ell_{\ell-K+1})\|_1 \leq \bar{\epsilon}, \quad \forall\ K \leq \ell \leq d - C\log d.$$

In other words, all probability estimates are $\bar{\epsilon}$ accurate except those in the first K and last $O(\log d)$ levels of the tree.

Proof: We present a very brief outline of the proof; see [11] for details. Consider a time at which the tree has depth d, and let (x^ℓ, a^ℓ) be some sequence of observations and actions with $K \leq \ell \leq d - \log d$. For any choice of $x^{\ell+1}$, the context $(x^{\ell+1}, a^\ell)$ will have been visited at least $(|\mathbb{X}||\mathbb{A}|)^{d-\ell-2}$ times. Hence, the distribution $\hat{P}_d(x^\ell, a^\ell)$ will have been estimated by at least $|\mathbb{X}|^{d-\ell-1}|\mathbb{A}|^{d-\ell-2}$ samples. Let $P^e_d(\cdot|x^\ell, a^\ell)$ be the empirical distribution of those samples. Via Sanov's Theorem and the union bound one may then show:

$$\Pr\left(\|\hat{P}_d(\cdot|x^\ell, a^\ell) - P(\cdot|x^\ell_{\ell-K+1}, a^\ell_{\ell-K+1})\|_1 > \bar{\epsilon}\right) \leq \exp(-|\mathbb{X}|^{d-\ell-1}|\mathbb{A}|^{d-\ell-2}\bar{\epsilon}^2/8 + |\mathbb{X}|\log(|\mathbb{X}|^d|\mathbb{A}|^d + 1)).$$

Algorithm 2 A variant of the learning algorithm with depth limitation.

$t \leftarrow 1$, $c \leftarrow 1$, $\tau_c \leftarrow 1$
$d \leftarrow 1$ {current depth of the tree}
$N_c(\cdot) \leftarrow 0$ {context counts}
$\hat{P}_d(\cdot) \leftarrow 1/|\mathbb{X}|$, $\hat{J}_d(\cdot) \leftarrow 0$ {transition probability/value function estimates}
for each time t **do**
  observe X_t
  if $N_c(X^t_{\tau_c}, A^{t-1}_{\tau_c}) > 0$ and $t - \tau_c < d$ **then**
    with probability $\gamma_{N_c(X^t_{\tau_c}, A^{t-1}_{\tau_c})}$, pick A_t uniformly over $\mathbb{A}$
    otherwise, pick A_t greedily according to $\hat{P}_c$, $\hat{J}_c$:

$$A_t \in \operatorname*{argmin}_{a_t} \sum_{x_{t+1}} \Big(\hat{P}(x_{t+1}|X^t_{\tau_c}, (A^{t-1}_{\tau_c}, a_t)) \hat{J}((X^t_{\tau_c}, x_{t+1}), (A^{t-1}_{\tau_c}, a_t))\Big)$$

  else {we are in a context not seen before or a context that is too long}
    pick A_t uniformly over $\mathbb{A}$
    $N_{c+1}(\cdot) \leftarrow N_c(\cdot)$
    for each s with $\tau_c \leq s \leq t$, in increasing order **do**
      $N_{c+1}(X^s_{\tau_c}, A^{s-1}_{\tau_c}) \leftarrow N_c(X^s_{\tau_c}, A^{s-1}_{\tau_c}) + 1$
    end for
    $c \leftarrow c+1$, $\tau_c \leftarrow t+1$ {start the next phrase}
    if every context (x^d, a^{d-1}) has been visited at least once **then**
      $\hat{P}_{d+1}(\cdot) \leftarrow \hat{P}_d(\cdot)$, $\hat{J}_{d+1}(\cdot) \leftarrow \hat{J}_d(\cdot)$
      for each ℓ with $1 \leq \ell \leq d$, in decreasing order **do**
        for each $(x^\ell, a^{\ell-1}$ **do**

$$\hat{P}_{d+1}(x_\ell|x^{\ell-1}, a^{\ell-1}) \leftarrow \frac{N_{c+1}(x^\ell, a^{\ell-1}) + 1}{\sum_{x'} N_{c+1}((x^{\ell-1}, x'), a^{\ell-1}) + |\mathbb{X}|}$$

$$\hat{J}_{d+1}(x^\ell, a^{\ell-1}) \leftarrow g(x_\ell) + \alpha \min_{a_\ell} \sum_{x_{\ell+1}} \hat{P}_{d+1}(x_{\ell+1}|x^\ell, a^\ell)\hat{J}_{d+1}(x^{\ell+1}, a^\ell)$$

        end for
      end for
      $d \leftarrow d+1$ {increase the depth of the tree}
    end if
  end if
  $t \leftarrow t+1$
end for

Let $\mathcal{A}_d$ be the event that there is some (x^ℓ, a^ℓ) with $K \leq \ell \leq d - C \log d$ and

$$\|\hat{P}_d(\cdot|x^\ell, a^\ell) - P(\cdot|x^\ell_{\ell-K+1}, a^\ell_{\ell-K+1})\|_1 > \bar{\epsilon}.$$

Using essentially the union bound and our above estimates, one may then show:

$$\sum_{d=1}^{\infty} \Pr(\mathcal{A}_d) < \infty.$$

Then, by the First Borel-Cantelli Lemma, the events $\{\mathcal{A}_d\}$ can occur at most finitely many times, and the theorem is proved. ∎

Using Lemma 3, we can establish the following theorem. The proof can be found in [11].

Theorem 2: In addition to Assumption 3, suppose that

$$\gamma_e \leq \frac{a_1}{(\log e)^{1+a_2}},$$

for some constants $a_1, a_2 > 0$. Under Algorithm 2, the fraction of time non-optimal decisions are made is asymptotically 0. In other words, with probability 1,

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathrm{I}\left\{A_t \neq \mu_t^*(X^t, A^{t-1})\right\} = 0.$$

Theorem 2 suggests that as time goes on, the fraction of errors is getting smaller and smaller. Assumption 2 guarantees that the process is ergodic under the optimal policy, hence there is a λ^* such that under the optimal policy, with probability 1,

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} g(X_t) = \lambda^*.$$

As Algorithm 2 follows the optimal policy correctly over longer and longer intervals, it becomes possible to prove the following result, a proof of which may be found at [11].

Theorem 3: Given the assumption of Theorem 2, under Algorithm 2, with probability 1,

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} g(X_t) = \lambda^*.$$

Hence, Algorithm 2 does as well as the optimal policy asymptotically.

VI. Future Directions

We have presented and analyzed two Lempel-Ziv inspired algorithms for learning. The algorithms we presented apply when the cost is only a function of the current observation. They can, however, easily be extended to the more general framework where the cost depends on the last K observations and actions. To see this, note that Bellman's Equation (3) can be modified to handle this case, and the corresponding appropriate dynamic programming operator can be used in the value function estimation steps in Algorithms 1 and 2. Lemma 1 will continue to hold, and all the rest of the analysis will follow.

A number of interesting questions remain.

1) Can Assumption 2 be relaxed? This was important for our analysis, but does not hold in many real systems.
2) Can asymptotic optimality be established for an algorithm similar to Algorithm 1? In particular, Algorithm 2 requires that we limit depth in order to wait to explore contexts that are both unlikely and that may penalize us heavily. An algorithm without such restrictions is likely to converge to λ^* far quicker.
3) One interesting special case is when the next observation is Markovian given the past K observations and only the latest action. In this case, a variation of Algorithms 1 or 2 that uses contexts of the form (x^s, a) could be used. Here, the resulting tree would have exponentially fewer nodes and would be much quicker to converge to the optimal policy. Indeed, for systems where there are few likely paths, it might be possible to prove bounds on the rate of convergence.

Acknowledgments

The first author was supported by a supplement to NSF Grant ECS-9985229 provided by the MKIDS Program. The second author was supported by a Benchmark Stanford Graduate Fellowship.

References

[1] D. Teneketzis, "Optimal real-time encoding-decoding of markov sources in noisy environments," in *Proc. Math. Theory of Networks and Sys. (MTNS)*, Leuwen, Belgium, 2004.

[2] H. S. Witsenhausen, "On the structure of real-time source coders," *Bell System Technical Journal*, vol. 58, no. 6, pp. 1437–1453, July-August 1979.

[3] M. Kearns and S. Singh, "Near-optimal reinforcement learning in polynomial time," in *Proc. 15th International Conf. on Machine Learning*. San Francisco, CA: Morgan Kaufmann, 1998, pp. 260–268.

[4] E. Even-Dar, S. M. Kakade, and Y. Mansour, "Reinforcement learning in POMDPs," 2004, preprint. URL: http://www.cis.upenn.edu/~skakade/papers/rl/learn_pomdp.pdf.

[5] D. P. de Farias and N. Megiddo, "Combining expert advice in reactive environments," 2004, preprint. URL: http://web.mit.edu/~pucci/www/expertsfull.pdf.

[6] N. Merhav, E. Ordentlich, G. Seroussi, and M. J. Weinberger, "On sequential strategies for loss functions with memory," *IEEE Transactions on Information Theory*, vol. 48, no. 7, pp. 1947–1958, July 2002.

[7] J. Ziv and A. Lempel, "Compression of individual sequences via variable-rate coding," *IEEE Transactions on Information Theory*, vol. 24, no. 5, pp. 530–536, 1978.

[8] N. Merhav and M. Feder, "Universal prediction," *IEEE Transactions on Information Theory*, vol. 44, no. 6, pp. 2124–2147, October 1998.

[9] J. S. Vitter and P. Krishnan, "Optimal prefetching via data compression," *Journal of the ACM*, vol. 43, no. 5, pp. 771–793, 1996.

[10] D. P. Bertsekas, *Dynamic Programming and Optimal Control*. Belmont, MA: Athena Scientific, 1995, vol. 2.

[11] V. F. Farias, C. C. Moallemi, B. Van Roy, and T. Weissman, "Appendix to ISIT submission," 2005, URL: http://www.moallemi.com/ciamac/papers/isit-2005-appendix.pdf.