TECHNICAL RESEARCH REPORT

An Asymptotically Efficient Algorithm for Finite Horizon Stochastic Dynamic Programming Problems

by Hyeong Soo Chang, Michael C. Fu, Steven I. Marcus

TR 2003-26

An Asymptotically Efficient Algorithm for Finite Horizon Stochastic Dynamic Programming Problems

Hyeong Soo Chang*, Michael C. Fu†, and Steven I. Marcus†

Abstract

We present a novel algorithm, called "Simulated Annealing Multiplicative Weights", for approximately solving large finite-horizon stochastic dynamic programming problems. The algorithm is "asymptotically efficient" in the sense that a finite-time bound for the sample mean of the optimal value function over a given finite policy space can be obtained, and the bound approaches the optimal value as the number of iterations increases. The algorithm updates a probability distribution over the given policy space with a very simple rule, and the sequence of distributions generated by the algorithm converges to a distribution concentrated only on the optimal policies for the given policy space. We also discuss how to reduce the computational cost of the algorithm to apply it in practice.

Keywords: stochastic dynamic programming, sampling, learning algorithm, simulated annealing

*H. S. Chang is with the Department of Computer Science and Engineering at Sogang University, Seoul, Korea, and can be reached by e-mail at hschang@ccs.sogang.ac.kr.

†M. Fu and S. I. Marcus are with the Institute for Systems Research at the University of Maryland, College Park, and can be reached by e-mail at mfu@rhsmith.umd.edu and marcus@eng.umd.edu, respectively.

1 Introduction

Consider a discrete-time stationary dynamic system with a finite horizon H:

$$x_{t+1} = f(x_t, a_t, w_t) \text{ for } t = 0, 1, ..., H-1,$$

where f is the "next state function", x_t is a random variable ranging over a set X giving the state at time t, a_t is the control to be chosen from a nonempty subset $A(x_t)$ of a given set of available controls C at time t, and w_t is a random disturbance uniformly and independently selected from [0,1] at time t, representing the uncertainty in the system.

For the control of the above system, we are given a nonstationary policy $\pi = \{\pi_t | \pi_t : X \rightarrow A(X), t = 0, 1, ..., H-1\}$. We let $\tilde{\Pi}$ be the set of all possible nonstationary policies. The problem we consider is finding an optimal policy in $\tilde{\Pi}$ that achieves the expected optimal total (discounted) reward over H or the *expected optimal value for* $\tilde{\Pi}$,

$$\sup_{\pi \in \tilde{\Pi}} E[V^{\pi}(x_0)], \quad x_0 \sim \delta$$

where δ is the initial state distribution and

$$V^{\pi}(x) = E_{w_0,...,w_{H-1}} \left[\sum_{t=0}^{H-1} \gamma^t R(x_t, \pi_t(x_t), w_t) \middle| x_0 = x \right], 0 < \gamma \leq 1,$$

where γ is a discount factor and R is a reward function that maps given three-tuple (x, a, w) with $x \in X$, $a \in C$ and $w \in [0, 1]$ to a nonnegative real number. We assume that throughout the present paper, the value of γ is fixed and $\sup_{x \in X, a \in C, w \in [0,1]} R(x, a, w) \leq \frac{1}{H}$ for simplicity (with a simple scaling, similar results for general bounded ranges can be obtained). We suppress the subscript H on V for the length of the horizon for the another usage below. The function f, together with X, A, and R make up a Markov Decision Process (MDP) or a stochastic dynamic programming problem (see, e.g., [1] or [5] for a substantial discussion on MDPs).

This paper presents a simple algorithm, called "*Simulated Annealing Multiplicative Weights*" (SAMW), to approximately solve large finite horizon MDPs, based on the "weighted majority algorithm" [8] within the context of "adaptive sampling-based learning". Specifically, we exploit the recent work of "multiplicative weights algorithm" from Freund and Schapire [4] in a completely different context: noncooperative repeated two-player bimatrix zero-sum games.

Given a finite set of policies $\Pi \subset \tilde{\Pi}$, the SAMW algorithm works with a probability distribution over Π, with the goal of concentrating probability mass on the best policies in the search space. The algorithm does not assume any structure on Π. The state and/or action spaces may be infinite. The algorithm is "asymptotically efficient" in the sense that a finite-time bound for the sample mean of the expected optimal value over Π can be obtained, and the bound approaches the expected optimal value as the number of iterations increases. Basically, at each iteration the algorithm updates the

probability distribution over Π using a very simple rule from the "simulated" value of following each policy in Π with a sampled initial state with respect to δ. A control parameter is associated with the algorithm and with a suitable "annealing" of the value of the parameter, the sequence of the distributions generated by the algorithm converges to a distribution concentrated only on the best policies in the set Π.

In some sense, the philosophy behind the algorithm is similar to the well-known Simulated Annealing (SA) algorithm [6] for solving *static* function optimization problems. SA can be viewed as a sequence of homogeneous Markov chains, each related with a fixed temperature value. The temperature is decreased or annealed between subsequent chains after the current chain is allowed to achieve equilibrium or probability distribution over the solution space, where the equilibrium is, in theory, reached by doing a local search that depends on the current candidate solution and the current value of the temperature parameter. But the equilibrium is, in practice, difficult to obtain. The sequence of equilibria converges to a distribution concentrated only on the best solutions. In contrast, SAMW does not perform any local search. It directly updates a probability distribution over the given policy space at each iteration and has a much simpler tuning process than SA. In this regard, it may be said that SAMW is a "compressed" version of SA with an extension to "dynamic" function optimization. The algorithmic feature of SAMW distinguishes itself from the existing "simulation-based" optimization techniques for approximately solving MDPs, e.g., AMS (Adaptive Multi-stage Sampling) [3], NDP (Neuro-Dynamic Programming) [2], Q-learning [12], TD(λ) [11], etc.

This paper is organized as follows. In Section 2, we present the SAMW algorithm and analyze the convergence property of the algorithm. In Section 3, we discuss how to alleviate a possibly high computational cost of SAMW. We conclude the present paper in Section 4 with some remarks.

2 Simulated Annealing Multiplicative Weights Algorithm

2.1 Basic algorithm description

We assume that a *finite* policy space $\Pi \subset \tilde{\Pi}$ is given for applying SAMW. We denote π^* as an *optimal policy in* Π if for all $\pi \in \Pi$,

$$E[V^{\pi^*}(x_0)] \geq E[V^{\pi}(x_0)], \quad x_0 \sim \delta$$

and let $V^* := E[V^{\pi^*}(x_0)]$, and call V^* the *expected optimal value for* Π. The goal of SAMW is to approximate V^* for a given initial state distribution δ or to find an approximately optimal policy in Π. Note that even if Π is relatively small, we cannot obtain the exact value of $E[V^{\pi}(x_0)]$, in practice.

At each iteration $i = 1, ..., \infty$ of SAMW, we first sample an initial state x^i with respect to δ, and then we generate H random numbers $w_0, ..., w_{H-1}$, i.e., $U(0,1)$ i.i.d. and independent from

previously sampled $\{w_{t'}\}, t' < t$. We then "*simulate*" each policy $\pi \in \Pi$ with the sampled random numbers to obtain

$$V_i^\pi(x^i) := \sum_{t=0}^{H-1} \gamma^t R(x_t, \pi_t(x_t), w_t), x_0 = x^i.$$

Thus, $V_i^\pi(x^i)$ is independently generated from $V_{i'}^\pi(x^{i'})$ for all $i' \neq i$. These values will be used for updating a probability distribution over Π at each iteration i. That is, $V_i^\pi(x^i)$ is a sample value of following the policy π for a given random number sequence with a sampled initial state x^i with respect to δ. Note that $V_i^\pi(x^i)$ satisfies that from our assumption on R, $0 \leq V_i^\pi(x^i) \leq 1$ for any π and x^i. For T such generated samples, we let

$$V_T^\pi := \frac{1}{T} \sum_{i=1}^{T} V_i^\pi(x^i)$$

and let $V_T^* := V_T^{\pi^*}$. Thus, V_T^π is just a sample mean for approximating the true value of following the policy π with the initial state distribution δ.

SAMW starts with the uniform distribution ϕ^1 over Π that will be used for the first iteration. At each iteration $i > 1$, SAMW computes a new distribution ϕ^{i+1} by a simple multiplicative rule: for each $\pi \in \Pi$,

$$\phi^{i+1}(\pi) = \phi^i(\pi) \cdot \frac{\beta^{V_i^\pi(x^i)}}{Z^i},$$

where β is a parameter of the algorithm and normalization factor Z^i is given by

$$Z^i = \sum_{\pi \in \Pi} \phi^i(\pi) \cdot \beta^{V_i^\pi(x^i)}.$$

We remark that while applying SAMW, only the sample value of following each policy $\pi \in \Pi$ needs to be observed if this is possible. The next state function and the reward function themselves do not have to be known as in reinforcement learning [11] as long as we obtain the sample value. The need of obtaining a sample value for each policy in Π at each iteration can be cumbersome if the search space is large. We discuss this issue later in Section 3.

2.2 Convergence Analysis

We let M be the set of all possible probability distributions over Π. For $m \in M$ and $\pi \in \Pi$, $m(\pi)$ denotes the probability of choosing policy π, where the goal is to concentrate the probability on the best policies in Π. To analyze the performance of SAMW, we consider the following measure of "distance" between two probability distributions, called the *relative entropy* (also known as *Kullback-Leibler* entropy):

$$D(p, q) := \sum_{\pi \in \Pi} p(\pi) \ln \left(\frac{p(\pi)}{q(\pi)} \right), \quad p, q \in M.$$

Note that $D(p,q) \geq 0$ for any p and q, and $D(p,q) = 0$ if and only if $p = q$. (However, the measure is not symmetric, hence not a true distance metric.) We also let for a given $m \in M$ and $V_i^\pi(x^i)$ with a sampled initial state x^i,

$$\bar{V}_i(m)(x^i) = \sum_{\pi \in \Pi} V_i^\pi(x^i) m(\pi).$$

The following lemma provides an upper bound for a sample mean of the expected optimal value for Π via the probability distributions generated by SAMW, regardless of the state and the action space sizes.

Lemma 2.1 *Select the parameter β in $(1,\infty)$ and set $\phi^1(\pi) = \frac{1}{|\Pi|}$ for all $\pi \in \Pi$. Then the sequence of distributions $\phi^1, ..., \phi^T$ generated by SAMW satisfies*

$$V_T^* \leq \frac{\beta - 1}{\ln \beta} \cdot \frac{1}{T} \sum_{i=1}^{T} \bar{V}_i(\phi^i)(x^i) + \frac{\ln |\Pi|}{T \ln \beta}.$$

Proof: We first prove that for any Dirac distribution $m \in M$ such that for an optimal policy π^* in Π, $m(\pi^*) = 1$ and $m(\pi) = 0$ for all $\pi \in \Pi - \{\pi^*\}$, and for any iteration i where SAMW is used with $\beta \in (1,\infty)$, we have that

$$D(m, \phi^{i+1}) - D(m, \phi^i) \leq (-\ln \beta) V_i^{\pi^*}(x^i) + (\beta - 1) \bar{V}_i(\phi^i)(x^i).$$

From the definition of D and ϕ^i we have that

$$\begin{aligned}
D(m, \phi^{i+1}) - D(m, \phi^i) &= \sum_{\pi \in \Pi} m(\pi) \ln \left(\frac{\phi^i(\pi)}{\phi^{i+1}(\pi)} \right) \\
&= \sum_{\pi \in \Pi} m(\pi) \ln \frac{Z^i}{\beta^{V_i^\pi(x^i)}} \\
&= (-\ln \beta) \sum_{\pi \in \Pi} m(\pi) V_i^\pi(x^i) + \ln Z^i \\
&\leq (-\ln \beta) \bar{V}_i(m)(x^i) + \ln \left[\sum_{\pi \in \Pi} \phi^i(\pi) \left(1 + (\beta - 1) V_i^\pi(x^i)\right) \right] \\
&\quad \text{because } \beta^a \leq 1 + (\beta - 1)a \text{ for } \beta \geq 0, a \in [0,1] \\
&= (-\ln \beta) \bar{V}_i(m)(x^i) + \ln \left(1 + (\beta - 1) \bar{V}_i(\phi^i)(x^i)\right)
\end{aligned}$$

Because $\ln(1 + a) \leq a$ for $a > -1$, it implies that

$$D(m, \phi^{i+1}) - D(m, \phi^i) \leq (-\ln \beta) \bar{V}_i(m)(x^i) + (\beta - 1) \bar{V}_i(\phi^i)(x^i).$$

Now, summing this inequality over $i = 1, ..., T$, we have that

$$\sum_{i=1}^{T} \bar{V}_i(m)(x^i) \leq \frac{\beta - 1}{\ln \beta} \sum_{i=1}^{T} \bar{V}_i(\phi^i)(x^i) + \frac{1}{\ln \beta} \left(D(m, \phi^1) - D(m, \phi^{T+1}) \right)$$

$$\leq \frac{\beta-1}{\ln\beta}\sum_{i=1}^{T}\bar{V}_i(\phi^i)(x^i)+\frac{1}{\ln\beta}D(m,\phi^1)$$
$$\text{because } D(m,\phi^{T+1})\geq 0.$$

By the selection of $\phi^1(\pi)=1/|\Pi|$ for all π, we have that $D(m,\phi^1)\leq \ln|\Pi|$. Therefore, it follows that from the structure of m,

$$\sum_{i=1}^{T}V_i^{\pi^*}(x^i)\leq \frac{\beta-1}{\ln\beta}\sum_{i=1}^{T}\bar{V}_i(\phi^i)(x^i)+\frac{\ln|\Pi|}{\ln\beta}. \tag{1}$$

Dividing both sides of Equation (1) by T, we finally have that

$$V_T^*\leq \frac{\beta-1}{\ln\beta}\cdot\frac{1}{T}\sum_{i=1}^{T}\bar{V}_i(\phi^i)(x^i)+\frac{\ln|\Pi|}{T\ln\beta}.$$

∎

As β approaches 1, $\frac{\beta-1}{\ln\beta}$ approaches 1 while $\frac{\ln|\Pi|}{\ln\beta}$ increases to infinity. On the other hand, if we fix β and let T increase, $\frac{\ln|\Pi|}{\ln\beta}$ becomes negligible relative to T. This means that we need to choose β as a sequence β_T or a function of T such that β_T approaches 1 as $T\to\infty$ for "convergence".

The following lemma states that with a proper tuning or "*annealing*" of the β-parameter, the distributions generated by SAMW do not change by SAMW after a "long" iteration. In other words, for a suitable tuning of β_T, the distribution ϕ_T generated by SAMW converges to a stationary distribution over Π as $T\to\infty$.

Lemma 2.2 *Set $\beta_T=1+\sqrt{\frac{1}{T}}, T>0$. Apply SAMW with β_T for a selected T. Then for the distributions ϕ^T and ϕ^{T+k} generated by SAMW for any fixed integer $k\geq 1$, we have that*

$$\lim_{T\to\infty}D(\phi^T,\phi^{T+k})=\lim_{T\to\infty}D(\phi^{T+k},\phi^T)=0.$$

Proof: From the definition of D,

$$\begin{aligned}
D(\phi^T,\phi^{T+1}) &= \sum_{\pi\in\Pi}\phi^T(\pi)\ln\left(\frac{\phi^T(\pi)}{\phi^{T+1}(\pi)}\right)\\
&\leq \max_{\pi\in\Pi}\ln\left(\frac{\phi^T(\pi)}{\phi^{T+1}(\pi)}\right)\\
&= \max_{\pi\in\Pi}\ln\frac{Z^T}{\beta_T^{V_T^{\pi}(x^T)}}\\
&\leq \ln\beta_T.
\end{aligned}$$

Therefore,

$$D(\phi^T,\phi^{T+k})\leq\sum_{j=1}^{k}D(\phi^{T+j-1},\phi^{T+j})\leq k\ln\beta_T.$$

We know that $D(\phi^T, \phi^{T+k}) \geq 0$ for any k. It follows that $D(\phi^T, \phi^{T+k}) \to 0$ as $T \to \infty$ because $\beta_T \to 1$ as $T \to \infty$.

For the proof of the symmetry, we can simply show that

$$D(\phi^{T+1}, \phi^T) \leq (\ln \beta_T)\bar{V}_T(\phi^T)(x^T) - \ln Z^T \leq \ln \beta_T,$$

making the desired convergence with similar arguments as in the previous case. ∎

Theorem 2.1 *Set $\beta_T = 1+\sqrt{\frac{1}{T}}, T > 0$ and $\phi^1(\pi) = \frac{1}{|\Pi|}$ for all $\pi \in \Pi$. The sequence of distributions $\phi^1, ..., \phi^T$ generated by SAMW with β_T satisfies that as $T \to \infty$,*

$$\frac{\beta_T - 1}{\ln \beta_T} \cdot \frac{1}{T} \sum_{i=1}^{T} \bar{V}_i(\phi^i)(x^i) + \frac{\ln |\Pi|}{T \ln \beta_T} \to V^*.$$

Proof: Observe that $\beta_T - 1 \leq \beta_T \ln \beta_T$ for all $T > 0$. From the result of Lemma 2.1,

$$V_T^* \leq \frac{\beta_T - 1}{\ln \beta_T} \cdot \frac{1}{T} \sum_{i=1}^{T} \bar{V}_i(\phi^i)(x^i) + \frac{\ln |\Pi|}{T \ln \beta_T} \leq \beta_T \cdot \frac{1}{T} \sum_{i=1}^{T} \bar{V}_i(\phi^i)(x^i) + \frac{\ln |\Pi|}{T \frac{\beta_T - 1}{\beta_T}}. \tag{2}$$

Replacing β_T by $1 + \sqrt{\frac{1}{T}}$ and letting $T \to \infty$ makes the left hand side approach V^* by the Law of Large Numbers and the second term of the right hand side of Equation (2) vanishes to zero.

For the first term of the right hand side of Equation (2), from Lemma 2.2, for every $\epsilon > 0$, there exists $T' < \infty$ such that $D(\phi^{i+k}, \phi^i) \leq \epsilon$ for all $i > T'$ and any fixed integer $k \geq 1$. Then, with picking $T > T'$, we have that

$$\begin{aligned}
\frac{1}{T} \sum_{i=1}^{T} \bar{V}_i(\phi^i)(x^i) &\leq \frac{1}{T} \left[\sum_{i=1}^{T'} \bar{V}_i(\phi^i)(x^i) + \sum_{i=T'+1}^{T} \bar{V}_i(\phi^i)(x^i) \right] \\
&\leq \frac{1}{T} \sum_{i=1}^{T'} \bar{V}_i(\phi^i)(x^i) + \frac{1}{T} \sum_{i=T'+1}^{T} \bar{V}_i(\phi^{T'})(x^i) + \frac{1}{T} \sum_{i=T'+1}^{T} \sum_{\pi \in \Pi} (\phi^i(\pi) - \phi^{T'}(\pi)) V_i^\pi(x^i) \\
&\leq \frac{1}{T} \sum_{i=1}^{T'} \bar{V}_i(\phi^i)(x^i) + \frac{1}{T} \sum_{i=1}^{T} \bar{V}_i(\phi^{T'})(x^i) + \frac{1}{T} \sum_{i=T'+1}^{T} \sum_{\pi \in \Pi} (\phi^i(\pi) - \phi^{T'}(\pi)) V_i^\pi(x^i) \quad (3)
\end{aligned}$$

Because $D(\phi^{i+k}, \phi^i) \leq \epsilon$ for all $i > T'$, for such i, $\max_{\pi \in \Pi} \ln \left(\frac{\phi^{i+k}(\pi)}{\phi^i(\pi)} \right) \leq \epsilon$ (cf. the proof of Lemma 2.2), making $\phi^{i+k}(\pi) \leq \phi^i(\pi) e^\epsilon$ for all $\pi \in \Pi$. Therefore, the last term in Equation (3) is upper bounded by $O(1 - e^\epsilon)$. Letting $T \to \infty$, the first term of Equation (3) vanishes to zero and the second term is upper bounded by V^*. Because we can make ϵ arbitrarily close to zero, the desired convergence is obtained. ∎

From Theorem 2.1 with Lemma 2.2, the result below immediately follows: the convergent distribution generated by SAMW is concentrated only on the optimal policies in Π.

Corollary 2.1 *Set* $\beta_T = 1+\sqrt{\frac{1}{T}}, T > 0$. *Apply SAMW with* β_T *for a selected* T. *Then, as* $T \to \infty$, *the distribution* ϕ^T *generated by SAMW converges to* $m^* \in M$ *such that* $m(\pi) = 0$ *for all* π *such that* $V^\pi < V^*$.

Proof: From Theorem 2.1 with Lemma 2.2, ϕ^T converges to a distribution $m \in M$ and $V^* = \sum_{\pi\in\Pi} m(\pi)V^\pi$ where $V^\pi = E_\delta[V^\pi(x_0)]$. Let $\Omega = \{\pi | V^* = V^\pi, \pi \in \Pi\}$. It follows that

$$\begin{aligned} V^* &= \sum_{\pi\in\Pi} m(\pi)V^\pi \\ &= \sum_{\pi\in\Omega} m(\pi)V^\pi + \sum_{\pi\in\Pi-\Omega} m(\pi)V^\pi \\ &= V^* \sum_{\pi\in\Omega} m(\pi) + \sum_{\pi\in\Pi-\Omega} m(\pi)V^\pi \end{aligned}$$

Therefore, we have that

$$V^* \sum_{\pi\in\Pi-\Omega} m(\pi) = \sum_{\pi\in\Pi-\Omega} m(\pi)V^\pi.$$

Now, suppose that there exists $\pi \in \Pi - \Omega$ such that $m(\pi) \neq 0$. Rearranging the previous equation, we have that $V^* = \sum_{\pi\in\Pi-\Omega} m'(\pi)V^\pi$ with $m'(\pi) = \frac{m(\pi)}{\sum_{\pi\in\Pi-\Omega} m(\pi)}, \pi \in \Pi$. It follows that $V^* \leq \max_{\pi\in\Pi-\Omega} V^\pi$. But from the definition and nonemptiness of Ω, $V^* > V^\pi$ for all $\pi \in \Pi - \Omega$, which makes a contradiction. Therefore, such m must have the structure of m^*. ∎

2.3 Convergence with probability one

Suppose that we use the distributions generated by SAMW to actually select a policy in Π (at random) at each iteration and apply the policy to the system. In other words, we apply sequentially a policy to the system at each iteration where the policy is selected at random with respect to the distribution generated by SAMW at each iteration. With the uniform distribution as the initial distribution for SAMW, we have shown that the *expected* per-iteration performance of SAMW approaches the expected optimal value as $T \to \infty$ if we appropriately tune the parameter β as a function of T.

In this subsection, we want to show that the actual per-iteration performance also converges to the expected optimal value with probability one. We note that a related result is proven by Freund and Schapire within the context of solving two-player zero-sum bimatrix repeated game [4].

Theorem 2.2 *Let* T *be given such that* $T = \sum_{k=1}^{l} T_k$ *with* $T_k = k^2$. *For all iterations* $i \in I_k = [1-k^2+\sum_{j=1}^{k} j^2, \sum_{j=1}^{k} j^2], k = 1, ..., l$, *set* $\beta_i = 1+\frac{1}{\sqrt{T_k}}$. *Let* $\phi^i \in M$ *and* $\theta^i \in \Pi$ *be the distribution generated by SAMW and a policy selected at random with respect to* ϕ^i *at the iteration* i, *respectively. Reset* $\phi^i = \frac{1}{|\Pi|}$ *if* $i = 1-k^2+\sum_{j=1}^{k} j^2, k = 1, ..., l$ *while applying SAMW. Then, with probability*

one, as $T \to \infty$,

$$\frac{1}{T}\sum_{i=1}^{T} V_i^{\theta^i}(x^i) \to V^*.$$

Proof: For each $I_k, k = 1, ..., l$, select $\epsilon_k = 2\frac{\sqrt{\ln k}}{k}$. Observe that the sequence of random variables $\kappa_i = V_i^{\theta^i}(x^i) - \bar{V}_i(\phi^i)(x^i)$ is a martingale difference sequence with $|\kappa_i| \leq 1$ because for all i, $E[\kappa_i|\kappa_1, ..., \kappa_{i-1}] = 0$ (note that in this case κ_i is a random variable with respect to only θ^i). Therefore, applying Azuma's inequality [9, p.309], we have that for every $\epsilon_k > 0$,

$$\Pr\left[\frac{1}{T_k}\left|\sum_{i\in I_k}\left(V_i^{\theta^i}(x^i) - \bar{V}_i(\phi^i)(x^i)\right)\right| > \epsilon_k\right] \leq 2e^{-0.5T_k\epsilon_k^2} = \frac{2}{k^2}. \tag{4}$$

The sum of the probability bound in Equation (4) over all k from 1 to ∞ is finite. Therefore, by the Borel-Cantelli lemma, with probability one all but a finite number of I_k's ($k = 1, ..., \infty$) satisfy that

$$\sum_{i\in I_k} \bar{V}_i(\phi^i)(x^i) \leq \sum_{i\in I_k} V_i^{\theta^i}(x^i) + T_k\epsilon_k. \tag{5}$$

This means that we can ignore the influence of the "bad" intervals such that the previous inequality is violated.

From Lemma 2.1 with the definition of β_i, for all $i \in I_k$,

$$\begin{aligned} T_k V_{T_k}^* \leq \frac{\beta_i - 1}{\ln \beta_i}\sum_{i\in I_k}\bar{V}_i(\phi^i)(x^i) + \frac{\ln|\Pi|}{\ln\beta_i} &\leq \beta_i \sum_{i\in I_k}\bar{V}_i(\phi^i)(x^i) + \frac{\ln|\Pi|}{\frac{\beta_i - 1}{\beta_i}} \\ &\leq \sum_{i=1}^{T_k}\bar{V}_i(\phi^i)(x^i) + \sqrt{T_k}(\ln|\Pi| + 1) + \ln|\Pi|. \end{aligned} \tag{6}$$

Combining Equation (5) and (6), we have that

$$T_k V_{T_k}^* \leq \sum_{i\in I_k} V_i^{\theta^i}(x^i) + T_k\epsilon_k + \sqrt{T_k}(\ln|\Pi| + 1) + \ln|\Pi|.$$

It follows that

$$TV_T^* \leq \sum_{i\in I_1\cup\cdots\cup I_l} V_i^{\theta^i}(x^i) + \sum_{k=1}^{l}\left[2k\sqrt{\ln k} + k(\ln|\Pi| + 1) + \ln|\Pi|\right].$$

Dividing both sides by T, we finally have that

$$V_T^* \leq \frac{1}{T}\sum_{i\in I_1\cup\cdots\cup I_l} V_i^{\theta^i}(x^i) + \frac{1}{T}\sum_{k=1}^{l}\left[2k\sqrt{\ln k} + k(\ln|\Pi| + 1) + \ln|\Pi|\right]. \tag{7}$$

Because $T = \sum_{k=1}^{l} k^2 = O(l^3)$, the error term in the right side of Equation (7) vanishes to zero as $T \to \infty$.

From Corollary 2.1, ϕ^i converges to m^* such that $m^*(\pi) = 0$ for all non-optimal policies $\pi \in \Pi$ as $i \to \infty$ or $T \to \infty$. Therefore, for every $\epsilon > 0$, there exists $T' < \infty$ such that $D(\phi^{T'+k}, m^*) \leq \epsilon$ for any fixed integer $k \geq 1$. This implies that with the similar arguments in the proof of Theorem 2.1, $\frac{1}{T}\sum_{i \in I_1 \cup \cdots \cup I_l} V_i^{\theta^i}(x^i)$ is upper bounded by V^* as $T \to \infty$, which provides the desired convergence result. ∎

3 Extensions to SAMW

3.1 Parallelization

SAMW can be naturally parallelized to speed up its computational cost. We partition the given policy space Π by $\Delta_j, j = 1, ..., N$ such that $\Delta_j \cap \Delta_{j'} = \emptyset$ for all j, j' and $\bigcup_{j=1}^N \Delta_j = \Pi$. SAMW is applied in parallel to each Δ_j. If SAMW is applied over T iterations for each Δ_j with a fixed value of β, we have the following finite-time bound from Lemma 2.1:

$$V_T^* \leq \max_{j=1,...,N} \left\{ \frac{\beta - 1}{\ln \beta} \cdot \frac{1}{T} \sum_{i=1}^{T} \bar{V}_i(\phi_j^i)(x_j^i) + \frac{\ln |\Delta_j|}{T \ln \beta} \right\}$$

where ϕ_j^i is the distribution generated by SAMW for Δ_j at the iteration i and x_j^i is the sampled initial state at the iteration i by SAMW for Δ_j.

3.2 ϵ-cut SAMW

While applying SAMW, at each iteration, we need to have a sample value of following each policy in Π. If Π is large, this causes a high computational cost. We have shown that the sequence of the distributions generated by SAMW converges to a distribution concentrated on the optimal policies in Π. This means that as the number of iterations increases, the contributions from non-optimal policies get smaller and smaller on the approximation of the true value of following an optimal policy in Π. Based on this fact, we can heuristically adjust the amount of sampling as the application of SAMW proceeds. Obviously, there would be many ways of achieving this. We discuss here a particular simple way by modifying the basic algorithmic procedure of SAMW and call the resulting version as ϵ-cut SAMW.

As before, ϵ-cut SAMW starts with the uniform distribution ρ^1 over Π that will be used for the first iteration. At each iteration $i > 1$, ϵ-cut SAMW computes a new distribution ρ^{i+1} by a simple multiplicative rule: for each $\pi \in \Pi$,

$$\rho^{i+1}(\pi) = \rho^i(\pi) \cdot \frac{\beta^{V_i^\pi(x^i)}}{Z^i},$$

where $V_i^\pi(x^i) = V_{i-1}^\pi(x^{i-1})$ if $\rho^i(\pi) \leq \epsilon$ for a fixed $\epsilon < \frac{1}{|\Pi|}$ and Z^i is again given by

$$Z^i = \sum_{\pi \in \Pi} \rho^i(\pi) \cdot \beta^{V_i^\pi(x^i)}.$$

In other words, we *reuse* the previously generated sample value of following a policy $\pi \in \Pi$ at the iteration i if $\rho^i(\pi)$ is small with respect to the preassigned value of ϵ.

To see why ϵ-cut SAMW works in an analytical way, let $S_i \subset \Pi$ be the set of policies for which we obtained a new sample value at the iteration i, that is, $S_i = \{\pi | \rho^i(\pi) > \epsilon, \pi \in \Pi\}$, and let $V^\pi_{i,n}(x^i)$ denote a newly sampled value of following a policy $\pi \in \Pi - S_i$ at the iteration i. With the similar reasoning of the proof of Lemma 2.1, for any Dirac distribution $m \in M$ such that for an optimal policy π^* in Π, $m(\pi^*) = 1$ and $m(\pi) = 0$ for all $\pi \in \Pi - \{\pi^*\}$, and for any iteration i where ϵ-cut SAMW is used with $\beta \in (1, \infty)$, we have that for $i > 1$,

$$\begin{aligned}
D(m, \rho^{i+1}) - D(m, \rho^i) &\leq (-\ln \beta) \left[\sum_{\pi \in S_i} m(\pi) V^\pi_i(x^i) + \sum_{\pi \in \Pi - S_i} m(\pi) V^\pi_{i-1}(x^{i-1}) \right] + (\beta - 1) \bar{V}_i(\rho^i)(x^i) \\
&\leq (-\ln \beta) \left[\sum_{\pi \in S_i} m(\pi) V^\pi_i(x^i) + \sum_{\pi \in \Pi - S_i} m(\pi) V^\pi_{i,n}(x^i) + m(\pi)[V^\pi_{i-1}(x^{i-1}) - V^\pi_{i,n}(x^i)] \right] \\
&\quad + (\beta - 1) \bar{V}_i(\rho^i)(x^i) \\
&\leq (-\ln \beta) \left[\sum_{\pi \in S_i} m(\pi) V^\pi_i(x^i) + \sum_{\pi \in \Pi - S_i} m(\pi) V^\pi_{i,n}(x^i) \right] + (\ln \beta) \sum_{\pi \in \Pi - S_i} m(\pi) \\
&\quad + (\beta - 1) \bar{V}_i(\rho^i)(x^i) \\
&\leq (-\ln \beta) \left[\sum_{\pi \in S_i} m(\pi) V^\pi_i(x^i) + \sum_{\pi \in \Pi - S_i} m(\pi) V^\pi_{i,n}(x^i) \right] + (\ln \beta) \Pr\{\rho^i(\pi^*) \leq \epsilon\} \\
&\quad + (\beta - 1) \bar{V}_i(\rho^i)(x^i).
\end{aligned}$$

Summing this inequality over $i = 1, ..., T$, and rearranging terms, and dividing both sides by T, we have that

$$V^*_T \leq \frac{\beta - 1}{\ln \beta} \cdot \frac{1}{T} \sum_{i=1}^{T} \bar{V}_i(\rho^i)(x^i) + \frac{\ln |\Pi|}{T \ln \beta} + \frac{1}{T} \sum_{i=1}^{T} \Pr\{\rho^i(\pi^*) \leq \epsilon\}.$$

We can see that the performance of ϵ-cut SAMW will be close to the original SAMW by choosing a proper value of ϵ.

3.3 Sample path optimization

Suppose that we (independently) sample T initial states $x^i, i = 1, ..., T$ with respect to δ and T random number sequences of the length H, $W_i = \{w_{i,t}, t = 0, ..., H-1\}, i = 1, ..., T$, where each $w_{i,t}$ is independently and uniformly sampled from [0,1]. Define the following deterministic problem: for a given finite set $\Pi \subset \tilde{\Pi}$, we want to obtain

$$J^*_T := \max_{\pi \in \Pi} \left[\frac{1}{T} \sum_{i=1}^{T} \left(\sum_{t=0}^{H-1} \gamma^t R(x_{i,t}, \pi(x_{i,t}), w_{i,t}) \right) \middle| x_{i,0} = x^i, i = 1, ..., T \right],$$

where the state dynamics are described by the next state function f such that for all i,

$$x_{i,t+1} = f(x_{i,t}, \pi_t(x_{i,t}), w_{i,t}), t = 0, ..., H-1.$$

We can then try to solve this deterministic problem to approximate the expected optimal value V^* for Π. The very idea of solving the deterministic problem after realizing a set of random number sequences and random initial states in order to optimize a given expected value function has been called "sample path optimization" or "sample average approximation" in sochastic (discrete) optimization literature (see, e.g., [10]). In particular, it has been shown that the exact solution of the deterministic problem converges to exponentially fast on the sample size T under some conditions [7]. However, solving the deterministic problem exactly is often difficult due to the size of the search space and more difficult if T is relatively large. The usual approach is to use an iterative method, e.g., SA, for approximately solving the deterministic problem, which will also require a non-trivial tuning process to optimize SA itself and the evaluation of a particular candidate solution with respect to the large sample set.

We can use the ϵ-cut SAMW to approximately solve the deterministic problem with the presampled initial states and presampled random number sequences. Furthermore, in theory, SAMW provides an upper bound for the optimal value J_T^* of the deterministic problem. With the similar reasoning in the proof of Lemma 2.1, we can show that with a selected value of β in $(1, \infty)$ and setting $\phi^1(\pi) = \frac{1}{|\Pi|}$ for all $\pi \in \Pi$, the sequence of distributions $\phi^1, ..., \phi^T$ generated by SAMW satisfies that

$$J_T^* \leq \frac{\beta - 1}{\ln \beta} \cdot \frac{1}{T} \sum_{i=1}^{T} \bar{V}_i(\phi^i)(x^i) + \frac{\ln |\Pi|}{T \ln \beta},$$

where $\bar{V}_i(\phi^i)(x^i)$ for SAMW is computed with the random number sequence W_i.

4 Concluding Remarks

The cooling schedule we presented in this paper is not the only way of controlling the parameter β. Finding an optimal tuning of β would be difficult but is a good future research topic.

Even though we presented the SAMW algorithm within the context of solving finite horizon stochastic dynamic programming problems, it can be used for more general situation. We have a probability space $(\Omega, \mathcal{F}, P)$, and a finite solution space S, e.g., a sequence of functions or a subset of $\Re^n$, etc., and a (measurable) evaluation function $g : S \times \Omega \to \Re$ with the assumption that g is bounded. If we wish to solve the problem of $\max_{s \in S} \int_\Omega g(s, \omega) P(d\omega)$ approximately, SAMW is a candidate approach.

References

[1] D. P. Bertsekas, *Dynamic Programming and Optimal Control, Volumes 1 and 2.* Athena Scientific, 1995.

[2] D. P. Bertsekas and J. Tsitsiklis, *Neuro-Dynamic Programming.* Athena Scientific, 1996.

[3] H. S. Chang, M. Fu, and S. Marcus, "An adaptive sampling algorithm for solving Markov decision processes," submitted to *Operations Research* (TR 2002-19, ISR, Univ. of Maryland, 2002).

[4] Y. Freund and R. Schapire, "Adaptive game playing using multiplicative weights," *Games and Economic Behavior*, vol. 29, pp. 79–103, 1999.

[5] O. Hernández-Lerma and J. B. Lasserre, *Discrete-Time Markov Control Processes: basic optimality criteria*, Springer, 1996.

[6] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by Simulated Annealing," *Science*, vol. 220, pp. 45–54, 1983.

[7] A. J. Kleywegt, A. Shapiro, and R. Homem-de-Mello, "The sample average approximation method for stochastic discrete optimization," *SIAM J. on Control and Optimization*, vol. 12, no. 2, pp. 479–502, 2001.

[8] N. Littlestone and M. K. Warmnuth, "The weighted majority algorithm," *Information and Computation*, vol. 108, pp. 212–261, 1994.

[9] S. M. Ross, *Stochastic Processes*, Second Edition, John Wiley & Sons, 1996.

[10] R. Y. Rubinstein and A. Shapiro, *Discrete Event Systems: Sensitivity Analysis and Stochastic Optimization by the Score Function Method*, John Wiley & Sons, 1993.

[11] R. Sutton and A. Barto, *Reinforcement Learning: An Introduction*, MIT Press, Cambridge, Massachusetts, 1998.

[12] C. J. C. H. Watkins, "Q-learning," *Machine Learning*, vol. 8, no. 3, pp. 279–292, 1992.