An Asymptotically Efficient Simulation-Based Algorithm for Finite Horizon Stochastic Dynamic Programming

Hyeong Soo Chang, Michael C. Fu, Jiaqiao Hu, and Steven I. Marcus

***Abstract*— We present a simulation-based algorithm called "Simulated Annealing Multiplicative Weights" (SAMW) for solving large finite-horizon stochastic dynamic programming problems. At each iteration of the algorithm, a probability distribution over candidate policies is updated by a simple multiplicative weight rule, and with proper annealing of a control parameter, the generated sequence of distributions converges to a distribution concentrated only on the best policies. The algorithm is "asymptotically efficient," in the sense that for the goal of estimating the value of an optimal policy, a provably convergent finite-time upper bound for the sample mean is obtained.**

***Index Terms*— stochastic dynamic programming, Markov decision processes, simulation, learning algorithms, simulated annealing**

I. Introduction

Consider a discrete-time system with a finite horizon H: $x_{t+1} = f(x_t, a_t, w_t)$ for $t = 0, 1, ..., H-1$, where x_t is the state at time t – ranging over a (possibly infinite) set X, a_t is the action at time t – to be chosen from a nonempty subset $A(x_t)$ of a given (possibly infinite) set of available actions A 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, and $f : X \times A(X) \times [0, 1] \to X$ is a next-state function. Throughout, we assume the initial state x_0 is given, but this is without loss of generality, as the results in the paper carry through for the case where x_0 follows a given distribution.

Define a nonstationary (non-randomized) policy $\pi = \{\pi_t | \pi_t : X \to A(X), t = 0, 1, ..., H-1\}$, and its corresponding finite-horizon discount *value function* given by

$$V^{\pi} = E_{w_0,\ldots,w_{H-1}}\left[\sum_{t=0}^{H-1} \gamma^t R(x_t, \pi_t(x_t), w_t)\right], \qquad (1)$$

with discount factor $\gamma \in (0, 1]$ and one-period reward function $R : X \times A(X) \times [0, 1] \to \mathcal{R}^+$. We suppress explicit dependence of the horizon H on V. The function f, together with X, A, and R, comprise a stochastic dynamic programming problem or a Markov decision process (MDP) [1] [8]. We assume throughout that the one-period reward function is bounded. For simplicity, but without loss of generality, we take the bound to be $1/H$, i.e., $\sup_{x\in X, a\in A, w\in[0,1]} R(x, a, w) \le 1/H$, so $0 \le V^{\pi} \le 1$.

The problem we consider is estimating the *optimal value* over a *given finite set* of policies Π:

$$V^* := \max_{\pi\in\Pi} V^{\pi}. \qquad (2)$$

This work was supported in part by the National Science Foundation under Grant DMI-0323220, in part by the Air Force Office of Scientific Research under Grant FA95500410210, and in part by the Department of Defense. The work of H.S. Chang was also supported by the Sogang University research grants in 2006.

H.S. Chang is with the Department of Computer Science and Engineering at Sogang University, Seoul 121-742, Korea. (e-mail:hschang@sogang.ac.kr).

M.C. Fu is with the Robert H. Smith School of Business and the Institute for Systems Research at the University of Maryland, College Park. (e-mail:mfu@rhsmith.umd.edu).

J. Hu is with the Department of Applied Mathematics & Statistics, SUNY, Stony Brook. (e-mail:jqhu@xx.xx.edu).

S.I. Marcus is with the Department of Electrical & Computer Engineering and the Institute for Systems Research at the University of Maryland, College Park. (e-mail:marcus@eng.umd.edu).

Preliminary portions of this paper appeared in the *Proceedings of the 42nd IEEE Conference on Decision and Control*, 2003.

Any policy π^* that achieves V^* is called an *optimal policy*. Our setting is that in which *explicit* forms for f and R are not available, but both can be *simulated*, i.e., sample paths for the states and rewards can be generated from a given random number sequence $\{w_0, \ldots, w_{H-1}\}$.

We present a simulation-based algorithm called "*Simulated Annealing Multiplicative Weights*" (SAMW) for solving (2), based on the "weighted majority algorithm" of [12]. Specifically, we exploit the recent work of the "multiplicative weights" algorithm studied by Freund and Schapire [7] in a completely different context: noncooperative repeated two-player bimatrix zero-sum games. At each iteration, the algorithm updates a probability distribution over Π by a multiplicative weight rule using the estimated (from simulation) value functions for all policies in Π, requiring $|\Pi|$ sample paths. With a proper "annealing" of the control parameter associated with the algorithm as in Simulated Annealing (SA) [10], the sequence of distributions generated by the multiplicative weight rule converges to a distribution concentrated only on policies that achieve V^*, motivating our choice of SAMW for the name of the algorithm.

The algorithm is "asymptotically efficient," in the sense that a *finite-time upper bound* is obtained for the sample mean of the value of an optimal policy, and the upper bound converges to V^* with rate $O(1/\sqrt{T})$, where T is the number of iterations. A sampling version of the algorithm that does not enumerate all policies in Π at each iteration, but instead samples from the sequence of generated distributions, is also shown to converge to V^*. The sampling version can be used as an *on-line* simulation-based control in the context of planning.

SAMW differs from the usual SA in that it does not perform any local search; rather, it directly updates a probability distribution over Π 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 stochastic dynamic programming. The use of probability distribution on the search space is a fundamentally different approach from existing "simulation-based" optimization techniques for solving MDPs, such as (basis-function based) neuro-dynamic programming [2], model-free approaches of Q-learning [19] and TD(λ)-learning [17], and (bandit-theory based) adaptive multi-stage sampling [4]. Updating a probability distribution over the search space is similar to the learning automata approach for stochastic optimization [13], but SAMW is based on a different multiplicative weight rule. Ordinal comparison [9] that simply chooses the current best $\pi \in \Pi$ from the sample mean of V^{π} does not provide a deterministic upper-bound even if a probabilistic bound is possible (see, e.g., Theorem 1 in [6] with letting each arm of the bandit into a policy). Furthermore, it is not clear how to design a variant of ordinal comparison that does not enumerate all policies in Π. This is also true for the recently proposed on-line control algorithms, parallel rollout and policy switching [5], for MDPs.

This paper is organized as follows. In Section II, we present the SAMW algorithm and in Sections III and IV, we analyze its convergence properties. We conclude in Section VI with some remarks.

II. Basic Algorithm Description

Let Φ be the set of all probability distributions over Π. For $\phi \in \Phi$ and $\pi \in \Pi$, let $\phi(\pi)$ denote the probability for policy π. The goal is to concentrate the probability on the optimal policies π^* in Π. The SAMW algorithm iteratively generates a sequence of distributions, where ϕ_i denotes the distribution at iteration i. Each iteration of SAMW requires H random numbers $w_0, ..., w_{H-1}$, i.e., i.i.d. $U(0, 1)$ and independent from previous iterations. Each policy $\pi \in \Pi$ is then *simulated* using the same sequence of random numbers for that

iteration (different random number sequences can also be used for each policy, and all of the results still hold) in order to obtain a sample path estimate of the value function (1):

$$V_i^\pi := \sum_{t=0}^{H-1} \gamma^t R(x_t, \pi_t(x_t), w_t), \tag{3}$$

where the subscript i denotes the iteration count, which has been omitted for notational simplicity in the quantities x_t and w_t. The estimates $\{V_i^\pi, \pi \in \Pi\}$ are used for updating a probability distribution over Π at each iteration i. Note that $0 \leq V_i^\pi \leq 1$ (a.s.) by the boundedness assumption. Note also that the size of Π in the worst case can be quite large, i.e., $|A|^{|X||H|}$ so that we assume here that Π is relatively small. In Section IV, we study the convergence property of a sampling version of SAMW that does not enumerate all policies in Π at each iteration.

The iterative updating to compute the new distribution ϕ^{i+1} from ϕ^i and $\{V_i^\pi\}$ uses a simple multiplicative rule:

$$\phi^{i+1}(\pi) = \phi^i(\pi)\frac{\beta_i^{V_i^\pi}}{Z^i}, \ \forall \pi \in \Pi, \tag{4}$$

where $\beta_i > 1$ is a parameter of the algorithm, the normalization factor Z^i is given by $Z^i = \sum_{\pi\in\Pi} \phi^i(\pi)\beta_i^{V_i^\pi}$, and the initial distribution ϕ^1 is the uniform distribution, i.e., $\phi^1(\pi) = 1/|\Pi| \ \forall \pi \in \Pi$.

III. Convergence Analysis

For $\phi \in \Phi$, define

$$\bar{V}_i(\phi) = \sum_{\pi\in\Pi} V_i^\pi \phi(\pi),$$

$$\Psi_T^\pi := \frac{1}{T}\sum_{i=1}^{T} V_i^\pi,$$

where Ψ_T^π is the sample mean estimate for the value function of policy π. Again, note that (a.s.) $0 \leq \bar{V}_i(\phi) \leq 1$ for all $\phi \in \Phi$. We remark that $\bar{V}_i(\phi)$ represents an *expected* reward for each fixed (iteration) experiment i, where the expectation is w.r.t. the distribution of the policy.

The following lemma provides a finite-time upper bound for the sample mean of the value function of an optimal policy in terms of the probability distributions generated by SAMW via (4).

Lemma 3.1: For $\beta_i = \beta > 1, \ i = 1, ..., T$, the sequence of distributions $\phi^1, ..., \phi^T$ generated by SAMW via (4) satisfies (a.s.)

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

for any optimal policy π^*.

Proof: The proof idea follows that of Theorem 1 in [7], for which it is convenient to introduce the following measure of "distance" between two probability distributions, called the *relative entropy* (also known as *Kullback-Leibler* entropy):

$$D(p, q) := \sum_{\pi\in\Pi} p(\pi) \ln\left(\frac{p(\pi)}{q(\pi)}\right), \quad p, q \in \Phi. \tag{5}$$

Although $D(p, q) \geq 0$ for any p and q, and $D(p, q) = 0$ if and only if $p = q$, the measure is not symmetric, hence not a true metric.

Consider any Dirac distribution $\phi^* \in \Phi$ such that for an optimal policy π^* in Π, $\phi^*(\pi^*) = 1$ and $\phi^*(\pi) = 0$ for all $\pi \in \Pi - \{\pi^*\}$. We first prove that

$$V_i^{\pi^*} \leq \frac{(\beta - 1)\bar{V}_i(\phi^i) + D(\phi^*, \phi^i) - D(\phi^*, \phi^{i+1})}{\ln \beta}, \tag{6}$$

where ϕ^i and ϕ^{i+1} are generated by SAMW via (4) and $\beta_i > 1$.

From the definition of D given by (5),

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

where the first inequality follows from the property $\beta^a \leq 1 + (\beta - 1)a$ for $\beta \geq 0, a \in [0, 1]$, and the last inequality follows from the property $\ln(1 + a) \leq a$ for $a > -1$. Solving for $V_i^{\pi^*}$ (recall $\beta > 1$) yields (6).

Summing the inequality (6) over $i = 1, ..., T$,

$$\begin{aligned}
\sum_{i=1}^{T} V_i^{\pi^*} &\leq \frac{\beta - 1}{\ln\beta}\sum_{i=1}^{T} \bar{V}_i(\phi^i) + \frac{1}{\ln\beta}\left(D(\phi^*, \phi^1) - D(\phi^*, \phi^{T+1})\right) \\
&\leq \frac{\beta - 1}{\ln\beta}\sum_{i=1}^{T} \bar{V}_i(\phi^i) + \frac{1}{\ln\beta} D(\phi^*, \phi^1) \\
&\leq \frac{\beta - 1}{\ln\beta}\sum_{i=1}^{T} \bar{V}_i(\phi^i) + \frac{\ln|\Pi|}{\ln\beta},
\end{aligned}$$

where the second inequality follows from $D(\phi^*, \phi^{T+1}) \geq 0$, and the last inequality uses the uniform distribution property that

$$\phi^1(\pi) = \frac{1}{|\Pi|} \ \ \forall \pi \quad \Longrightarrow \quad D(\phi^*, \phi^1) \leq \ln|\Pi|.$$

Dividing both sides by T yields the desired result. ■

If $(\beta - 1)/\ln\beta$ is very close to 1 and at the same time $\ln|\Pi|/(T\ln\beta)$ is very close to 0, then the above inequality implies that the *expected* per-iteration performance of SAMW is very close to the optimal value. However, letting $\beta \to 1$, $\ln|\Pi|/\ln\beta \to \infty$. On the other hand, for fixed β and T increasing, $\ln|\Pi|/\ln\beta$ becomes negligible relative to T. Thus, from the form of the bound, it is clear that the sequence β_T should be chosen as a function of T such that $\beta_T \to 1$ and $T\ln\beta_T \to \infty$ in order to achieve convergence.

Define the total variation distance for probability distributions p and q by $d_T(p, q) := \sum_{\pi\in\Lambda} |p(\pi) - q(\pi)|$. The following lemma states that the sequence of distributions generated by SAMW converges to a stationary distribution, with a proper tuning or annealing of the β-parameter.

Lemma 3.2: Let $\{\psi(T)\}$ be a decreasing sequence such that $\psi(T) > 1 \ \forall T$ and $\lim_{T\to\infty} \psi(T) = 1$. For $\beta_i = \psi(T), \ i = 1, ..., T+k, \ k \geq 1$, the sequence of distributions $\phi^1, ..., \phi^T$ generated by SAMW via (4) satisfies (a.s.)

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

Proof: From the definition of D given by (5),

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

since $V_T^\pi \leq 1$ and $Z^T \geq 1$ for all π and any T.

Applying Pinsker's inequality [18],

$$d_T(\phi^T, \phi^{T+1}) \leq \sqrt{2D(\phi^T, \phi^{T+1})} \leq \sqrt{2 \ln \psi(T)}.$$

Therefore,

$$d_T(\phi^T, \phi^{T+k}) \leq \sum_{j=1}^{k} d_T(\phi^{T+j-1}, \phi^{T+j}) \leq \sum_{j=0}^{k-1} \sqrt{2 \ln \psi(T+j)}.$$

Because $d_T(\phi^T, \phi^{T+k}) \geq 0$ for any k and $\sum_{j=0}^{k-1} \sqrt{2 \ln \psi(T+j)} \to 0$ as $T \to \infty$, $d_T(\phi^T, \phi^{T+k}) \to 0$ as $T \to \infty$. ∎

Theorem 3.1: Let $\{\psi(T)\}$ be a decreasing sequence such that $\psi(T) > 1 \ \forall T$, $\lim_{T\to\infty} \psi(T) = 1$, and $\lim_{T\to\infty} T \ln \psi(T) = \infty$. For $\beta_i = \psi(T),\ i = 1, ..., T$, the sequence of distributions $\phi^1, ..., \phi^T$ generated by SAMW via (4) satisfies (a.s.)

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

and $\phi_i \to \phi^* \in \Phi$, where $\phi^*(\pi) = 0$ for all π such that $V^\pi < V^*$.

Proof: Using $x - 1 \leq x \ln x$ for all $x \geq 1$ and Lemma 3.1,

$$\begin{aligned} \Psi_T^{\pi^*} &\leq \frac{\psi(T) - 1}{\ln \psi(T)} \cdot \frac{1}{T} \sum_{i=1}^{T} \bar{V}_i(\phi^i) + \frac{\ln |\Pi|}{T \ln \psi(T)} \\ &\leq \psi(T) \cdot \frac{1}{T} \sum_{i=1}^{T} \bar{V}_i(\phi^i) + \frac{\ln |\Pi|}{T \ln \psi(T)}. \end{aligned} \tag{7}$$

In the limit as $T \to \infty$, the lefthand side converges to V^* by the law of large numbers, and in the rightmost expression in (7), $\psi(T) \to 1$ and the second term vanishes, so it suffices to show that $T^{-1} \sum_{i=1}^{T} \bar{V}_i(\phi^i)$ is bounded from above by V^* (in the limit).

From Lemma 3.2, for every $\epsilon > 0$, there exists $T' < \infty$ such that $d_T(\phi^i, \phi^{i+k}) \leq \epsilon$ for all $i > T'$ and any integer $k \geq 1$. Then, for $T > T'$, we have (a.s.)

$$\begin{aligned} \frac{1}{T} \sum_{i=1}^{T} \bar{V}_i(\phi^i) &= \frac{1}{T} \left[\sum_{i=1}^{T'} \bar{V}_i(\phi^i) + \sum_{i=T'+1}^{T} \bar{V}_i(\phi^i) \right] \\ &\leq \frac{1}{T} \sum_{i=1}^{T'} \bar{V}_i(\phi^i) + \frac{1}{T} \sum_{i=T'+1}^{T} \bar{V}_i(\phi^{T'}) \\ &\quad + \frac{1}{T} \sum_{i=T'+1}^{T} \sum_{\pi \in \Pi} |\phi^i(\pi) - \phi^{T'}(\pi)| V_i^\pi \\ &\leq \frac{1}{T} \sum_{i=1}^{T'} \bar{V}_i(\phi^i) + \frac{1}{T} \sum_{i=1}^{T} \bar{V}_i(\phi^{T'}) \\ &\quad + \frac{1}{T} \sum_{i=T'+1}^{T} \sum_{\pi \in \Pi} |\phi^i(\pi) - \phi^{T'}(\pi)| V_i^\pi \\ &\leq \frac{1}{T} \sum_{i=1}^{T'} \bar{V}_i(\phi^i) + \frac{1}{T} \sum_{i=1}^{T} \bar{V}_i(\phi^{T'}) + \frac{1}{T} \sum_{i=T'+1}^{T} |\Pi| \epsilon, \end{aligned} \tag{8}$$

the last inequality following from $\max_{\pi \in \Pi} |\phi^{i+k}(\pi) - \phi^i(\pi)| \leq \epsilon$ and $V_i^\pi \leq 1 \ \forall i > T' \ \forall k \geq 1, \pi \in \Pi$. As $T \to \infty$, the first term of (8) vanishes, and the second term converges by the law of large numbers to $V^\pi, \pi \sim \phi^{T'}$, which is bounded from above by V^*. Since ϵ can be chosen arbitrarily close to zero, the desired convergence follows.

The second part of the theorem follows directly from the first part with Lemma 3.2, with a proof obtained in a straightforward manner by assuming there exists a $\pi \in \Pi$ such that $\phi^*(\pi) \neq 0$ and $V^\pi < V^*$, leading to a contradiction. We skip the details. ∎

An example of a decreasing sequence $\{\psi(T)\}, T = 1, 2, ...,$ that satisfies the condition of Theorem 3.1 is $\psi(T) = 1 + \sqrt{1/T}, T > 0$.

IV. Convergence of the Sampling Version of the Algorithm

Instead of estimating the value functions for every policy in Π according to (3), which requires simulating all policies in Π, a *sampling* version of the algorithm would sample a subset of the policies in Π at each iteration i according to ϕ^i and simulate only those policies (and estimate their corresponding value functions). In this context, Theorem 3.1 essentially establishes that the *expected* per-iteration performance of SAMW approaches the optimal value as $T \to \infty$ for appropriately selected tuning sequence $\{\beta_i\}$. Here, we show that the actual (distribution sampled) per-iteration performance also converges to the optimal value using a particular annealing schedule of the parameter β. For simplicity, we assume that a *single* policy is sampled at each iteration (i.e., subset is a singleton). A related result is proven by Freund and Schapire within the context of solving two-player zero-sum bimatrix repeated game [7], and the proof of the following theorem is based on theirs.

Theorem 4.1: Let $T_k = \sum_{j=1}^{k} j^2$. For $\beta_i = 1 + 1/k, T_{k-1} < i \leq T_k$, let $\{\phi^i\}$ denote the sequence of distributions generated by SAMW via (4), with "resetting" of $\phi^i(\pi) = 1/|\Pi| \ \forall \pi$ at each $i = T_k$. Let $\hat{\pi}(\phi^i)$ denote the policy sampled from ϕ^i (at iteration i). Then (a.s. as $k \to \infty$),

$$\frac{1}{T_k} \sum_{i=1}^{T_k} V_i^{\hat{\pi}(\phi^i)} \to V^*.$$

Proof: The sequence of random variables $\kappa_i = V_i^{\hat{\pi}(\phi^i)} - \bar{V}_i(\phi^i)$ forms a martingale difference sequence with $|\kappa_i| \leq 1$, since $E[\kappa_i | \kappa_1, ..., \kappa_{i-1}] = 0$ for all i. Let $\epsilon_k = 2\sqrt{\ln k / k}$ and $I_k = [T_{k-1} + 1, T_k]$. Applying Azuma's inequality [14, p.309], we have that for every $\epsilon_k > 0$,

$$P\left(\frac{1}{k^2} \left| \sum_{i \in I_k} \left(V_i^{\hat{\pi}(\phi^i)} - \bar{V}_i(\phi^i) \right) \right| > \epsilon_k \right) \leq 2e^{-0.5 k^2 \epsilon_k^2} = \frac{2}{k^2}. \tag{9}$$

The sum of the probability bound in (9) over all k from 1 to ∞ is finite. Therefore, by the Borel-Cantelli lemma, (a.s.) all but a finite number of I_k's ($k = 1, ..., \infty$) satisfy

$$\sum_{i \in I_k} \bar{V}_i(\phi^i) \leq \sum_{i \in I_k} V_i^{\hat{\pi}(\phi^i)} + k^2 \epsilon_k, \tag{10}$$

so those I_k that violate (9) can be ignored (a.s.).

From Lemma 3.1 with the definition of β_i, for all $i \in I_k$,

$$\begin{aligned} k^2 \Psi_{k^2}^{\pi^*} &\leq \sum_{i \in I_k} \frac{\beta_i - 1}{\ln \beta_i} \bar{V}_i(\phi^i) + \frac{\ln |\Pi|}{\ln \beta_i} \\ &\leq \sum_{i \in I_k} \beta_i \bar{V}_i(\phi^i) + \frac{\ln |\Pi|}{\frac{\beta_i - 1}{\beta_i}} \\ &= \sum_{i \in I_k} \left(1 + \frac{1}{k} \right) \bar{V}_i(\phi^i) + \ln |\Pi| (k+1) \\ &\leq \sum_{i \in I_k} \bar{V}_i(\phi^i) + k + \ln |\Pi| (k+1), \end{aligned} \tag{11}$$

where the last inequality follows from $\bar{V}_i(\phi) \leq 1 \ \forall \phi \in \Phi$ and $|I_k| = k^2$.

Combining (10) and (11) and summing, we have

$$\begin{aligned} T_k \Psi_{T_k}^{\pi^*} \leq & \sum_{i \in I_1 \cup \cdots \cup I_k} V_i^{\hat{\pi}(\phi^i)} \\ & + \sum_{j=1}^{k} \left[2j \sqrt{\ln j} + j(\ln |\Pi| + 1) + \ln |\Pi| \right], \text{ so} \end{aligned}$$

$$\Psi_{T_k}^{\pi^*} \leq \frac{1}{T_k} \sum_{i \in I_1 \cup \cdots \cup I_k} V_i^{\hat{\pi}(\phi^i)} + \frac{1}{T_k} \sum_{j=1}^{k} \left[2j\sqrt{\ln j} + j(\ln|\Pi| + 1) + \ln|\Pi| \right]. \quad (12)$$

Because T_k is $O(k^3)$, the term on the righthand side of (12) vanishes as $k \to \infty$. Therefore, for every $\epsilon > 0$, (a.s.) for all but a finite number of values of T_k,

$$\Psi_{T_k}^{\pi^*} \leq \frac{1}{T_k} \sum_{i=1}^{T_k} V_i^{\hat{\pi}(\phi^i)} + \epsilon.$$

We now argue that $\{\phi^i\}$ converges to a fixed distribution as $k \to \infty$, so that eventually the term $T_k^{-1} \sum_{i=1}^{T_k} V_i^{\hat{\pi}(\phi^i)}$ is bounded from above by V^*. With similar reasoning as in the proof of Lemma 3.2, for every $\epsilon > 0$, there exists $T' \in I_k$ for some $k > 1$ such that for all $i > T'$ with $i + j \in I_k$, $j \geq 1$, $d_T(\phi^i, \phi^{i+j}) \leq \epsilon$. Taking $T > T'$ with $T \in I_k$,

$$\begin{aligned}
\frac{1}{T} \sum_{i=1}^{T} V_i^{\hat{\pi}(\phi^i)} &= \frac{1}{T} \left[\sum_{i=1}^{T'} V_i^{\hat{\pi}(\phi^i)} + \sum_{i=T'+1}^{T} V_i^{\hat{\pi}(\phi^i)} \right] \\
&\leq \frac{1}{T} \sum_{i=1}^{T'} V_i^{\hat{\pi}(\phi^i)} + \frac{1}{T} \sum_{i=T'+1}^{T} V_i^{\hat{\pi}(\phi^{T'})} \\
&\quad + \frac{1}{T} \sum_{i=T'+1}^{T} V_i^{\hat{\pi}(\phi^i)} - V_i^{\hat{\pi}(\phi^{T'})} \\
&\leq \frac{1}{T} \sum_{i=1}^{T'} V_i^{\hat{\pi}(\phi^i)} + \frac{1}{T} \sum_{i=1}^{T} V_i^{\hat{\pi}(\phi^{T'})} \\
&\quad + \frac{1}{T} \sum_{i=T'+1}^{T} \left(V_i^{\hat{\pi}(\phi^i)} - V_i^{\hat{\pi}(\phi^{T'})} \right). \quad (13)
\end{aligned}$$

Letting $T \to \infty$, the first term on the righthand side of (13) vanishes, and the second term is bounded from above by V^*, because the second term converges to $V^{\pi}, \pi \sim \phi^{T'}$, from the law of large numbers. We know that for all $i > T'$ in I_k, $-\epsilon + \phi^i(\pi) \leq \phi^{i+j}(\pi) \leq \phi^i(\pi) + \epsilon$ for all $\pi \in \Pi$ and any j. Therefore, as ϵ can be chosen arbitrarily close to zero, $\{\phi^i\}$ converge to the distribution $\phi^{T'}$, making the last term vanish (once each policy is sampled from the same distribution over Π, the simulated value would be the same for the same random numbers), which provides the desired convergence result. ■

V. A Numerical Example

To illustrate the performance of SAMW, we consider a simple finite-horizon inventory control problem with lost sales, zero order lead time, and linear holding and shortage costs. Given an inventory level, orders are placed and received, demand is realized, and the new inventory level is calculated. Formally, we let D_t, a discrete random variable, denote the demand in period t, x_t the inventory level at period t, a_t the order amount at period t, p the per period per unit demand lost penalty cost, h the per period per unit inventory holding cost, and M the inventory capacity. Thus, the inventory level evolves according to the following dynamics: $x_{t+1} = \max\{0, x_t + a_t - D_t\}$. The goal is to minimize, over a given set of (nonstationary) policies Π, the expected total cost over the entire horizon from a given initial inventory level x_0, i.e., $\min_{\pi \in \Pi} E[\sum_{t=0}^{H-1} [h \max\{0, x_t + \pi_t(x_t) - D_t\} + p \max\{0, D_t - x_t - \pi_t(x_t)\}] | x_0 = x]$.

The following set of parameters is used in our experiments: $M = 20$, $H = 3$, $h = 0.003$, $p = 0.012$, $x_0 = 5$ and $x_t \in \{0, 5, 10, 15, 20\}$ for $t = 1, \ldots, H$, $a_t \in \{0, 5, 10, 15, 20\}$ for all $t = 0, \ldots, H-1$, and D_t is a discrete uniformly distributed random variable taking values in $\{0, 5, 10, 15, 20\}$. The values of h and p are chosen so as to satisfy the one-period reward bound assumed in the SAMW convergence results. Note that since we are ignoring the setup cost (i.e., no fixed order cost), it is easy to see that the optimal order policy follows a threshold rule, in which an order is placed at period t if the inventory level x_t is below a certain threshold S_t, and the amount to order is equal to the difference $\max\{0, S_t - x_t\}$. Thus, by taking advantage of this structure, in actual implementation of SAMW, we restrict the search of the algorithm to the set of threshold policies, i.e., $\Pi = (S_0, S_1, S_2),\ S_t \in \{0, 5, 10, 15, 20\},\ t = 0, 1, 2$, rather than the set of all admissible policies.

We implemented two versions of SAMW, i.e., the fully sampled version of SAMW, which constructs the optimal value function estimate by enumerating all policies in Π and using all value function estimates, and the single sampling version of SAMW introduced in Theorem 4.1, which uses just one sampled policy in each iteration to update the optimal value function estimate; however, updating ϕ^i requires value function estimates for all policies in Π. For numerical comparison, we also applied the adaptive multi-stage sampling (AMS) algorithm [4] and a non-adaptive multi-stage sampling (NMS) algorithm. These two algorithms are simulation-tree based methods, where each node in the tree represents a state, with the root node being the initial state, and each edge signifies the sampling of an action. They both use forward search to generate sample path from the initial state to the final state, and update the value function backwards only at those visited states. The difference between the AMS and NMS algorithms is in the way actions are sampled at each decision period: AMS samples actions in an adaptive manner according to some performance index, whereas NMS simply samples each action for a fixed number of times. A detailed description of these approaches can be found in [4].

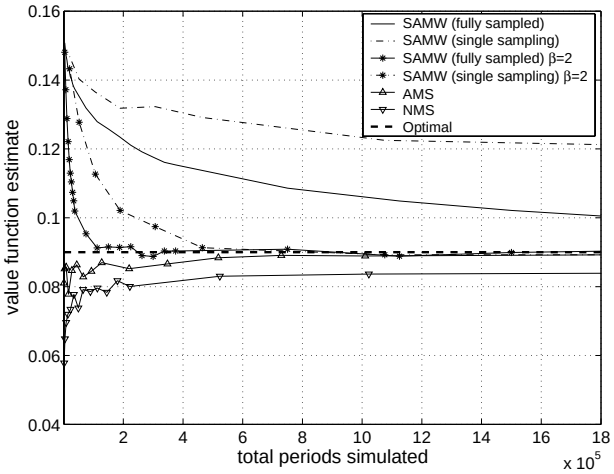

Fig. 1. Average performance (mean of 25 simulation replications, resulting in confidence half-widths within 5% of estimated mean) of SAMW, AMS, and NMS on the inventory control problem ($h = 0.003$, $p = 0.012$).

Figure 1 shows the performance of these algorithms as a function of the total number of periods simulated, based on 25 independent replications. The results indicate convergence of both versions of SAMW; however, the two alternative benchmark algorithms AMS and NMS seem to provide superior empirical performance over SAMW. We believe this is because the annealing schedule for β used in SAMW is too conservative for this problem, thus leading to slow convergence. To improve the empirical performance of SAMW, we also implemented both versions of the algorithm with β being

held constant throughout the search, i.e., independent of T. The $\beta = 2$ case is included in Figure 1, which shows significantly improved performance. Experimentation with the SAMW algorithm also revealed that it performed even better for cost parameters values in the inventory control problem that do not satisfy the strict reward bound. One such example is shown in Figure 2, for the case $h = 3$ and $p = 12$ (all other parameter values unchanged).

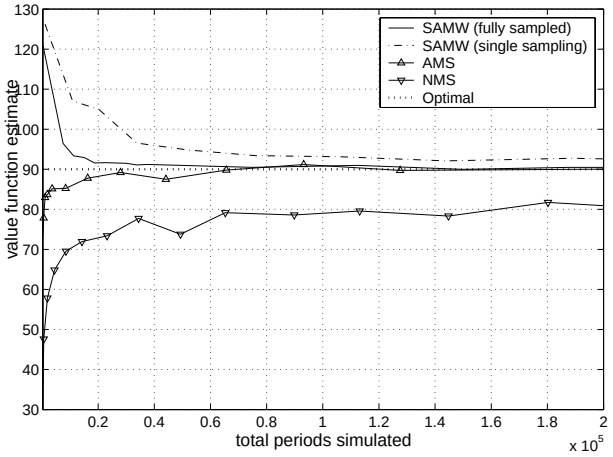

Fig. 2. Average performance (mean of 25 simulation replications, resulting in confidence half-widths within 5% of estimated mean) of SAMW, AMS, and NMS on the inventory control problem ($h = 3$, $p = 12$).

VI. Concluding Remarks

SAMW can be naturally parallelized to speed up its computational cost. Partition the given policy space Π into $\{\Delta_j\}$ such that $\Delta_j \cap \Delta_{j'} = \emptyset$ for all $j \neq j'$ and $\bigcup_j \Delta_j = \Pi$, and apply the algorithm in parallel for T iterations on each Δ_j. For a fixed value of $\beta > 1$, we have the following finite-time bound from Lemma 3.1:

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

where ϕ_j^i is the distribution generated for Δ_j at iteration i.

The original version of SAMW recalculates an estimate of the value function for all policies in Π at each iteration, requiring each policy to be simulated. If Π is large, this may not be practical, and the sampling version of SAMW given by Theorem 4.1 also requires each value function estimate in order to update the ϕ^i at each iteration. One simple alternative is to use the prior value function estimates for updating ϕ^i, except for the single sampled one; thus, only one simulation per iteration would be required. Specifically, $V_i^\pi := V_{i-1}^\pi$ if π not sampled at iteration i; else obtain a new estimate of V_i^π via (3). An extension of this is to use a threshold on ϕ^i to determine which policies will be simulated. Since the sequence of the distributions generated by SAMW converges to a distribution concentrated on the optimal policies in Π, as the number of iterations increases, the contributions from non-optimal policies get smaller and smaller, so these policies need not be resimulated (and value function estimates updated) very often. Specifically, $V_i^\pi := V_{i-1}^\pi$ if $\phi^i(\pi) \leq \epsilon$; else obtain a new estimate of V_i^π via (3).

The cooling schedule presented in Theorem 4.1 is just one way of controlling the parameter β. Characterizing properties of good schedules is critical to effective implementation, as the numerical experiments showed. The numerical experiments also demonstrated that the algorithm may work well outside the boundaries of the assumptions under which theoretical convergence is proved, specifically the bound on the one-period reward function and the value of the cooling parameter β. We suspect this has something to do with the scaling of the algorithm, but more investigation into this phenomena is clearly warranted.

Finally, we presented SAMW in the MDP framework for optimization of the sequential decision making processes. Even though the idea is general, the actual algorithm depends on the sequential structure of MDPs. The general problem given by (2) takes the form of a general stochastic optimization problem, so SAMW can also be adapted to serve as a global stochastic optimization algorithm for bounded expected value objective functions.

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. C. Fu, and S. I. Marcus, "An asymptotically efficient algorithm for finite horizon stochastic dynamic programming problems," in *Proc. of the 42nd IEEE Conf. Decision and Control*, 2003, pp. 3818–3823.

[4] H. S. Chang, M. C. Fu, J. Hu, and S. I. Marcus, "An adaptive sampling algorithm for solving Markov decision processes," *Operations Research*, vol. 53, no. 1, pp. 126-139, 2005.

[5] H. S. Chang, R. Givan, and E. K. P. Chong, "Parallel rollout for online solution of partially observable Markov decision processes," *Discrete Event Dynamic Systems: Theory and Application*, vol. 14, no. 3, pp. 309–341, 2004.

[6] E. Even-Dar, S. Mannor, and Y. Mansour, "PAC bounds for multi-armed bandit and Markov decision processes," in *Proc. of the 15th Annual Conf. on Computational Learning Theory*, 2002, pp. 255–270.

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

[8] O. Hernández-Lerma and J. B. Lasserre, *Discrete-Time Markov Control Processes: Basic Optimality Criteria*, Springer, 1996.

[9] Y. C. Ho, C. Cassandras, C-H. Chen, and L. Dai, "Ordinal optimization and simulation," *J. of Operations Research Society*, vol. 21, pp. 490–500, 2000.

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

[11] 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.

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

[13] A. S. Poznyak and K. Najim, *Learning Automata and Stochastic Optimization*, Springer-Verlag, 1997.

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

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

[16] N. Shimkin and A. Shwartz, "Guaranteed performance regions in Markovian systems with competing decision makers," *IEEE Trans. on Automatic Control*, vol. 38, no. 1, pp 84–95, 1993.

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

[18] F. Topsoe, "Bounds for entropy and divergence for distributions over a two-element set," *J. of Inequalities in Pure and Applied Mathematics*, vol. 2, issue 2, Article 25, 2001.

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