# An Incremental Fast Policy Search Using a Single Sample Path

Ajin George Joseph and Shalabh Bhatnagar(✉)

Indian Institute of Science, Bangalore, India
{ajin,shalabh}@iisc.ac.in

**Abstract.** In this paper, we consider the control problem in a reinforcement learning setting with large state and action spaces. The control problem most commonly addressed in the contemporary literature is to find an optimal policy which optimizes the long run $\gamma$-discounted transition costs, where $\gamma \in [0, 1)$. They also assume access to a generative model/simulator of the underlying MDP with the hidden premise that realization of the system dynamics of the MDP for arbitrary policies in the form of sample paths can be obtained with ease from the model. In this paper, we consider a cost function which is the expectation of a approximate value function w.r.t. the steady state distribution of the Markov chain induced by the policy, without having access to the generative model. We assume that a single sample path generated using a priori chosen behaviour policy is made available. In this information restricted setting, we solve the generalized control problem using the incremental cross entropy method. The proposed algorithm is shown to converge to the solution which is globally optimal relative to the behaviour policy.

## 1 Introduction

In this paper, we consider a reinforcement learning setting with the underlying Markov decision process (MDP) defined by the 4-tuple $(\mathbb{S}, \mathbb{A}, \mathrm{R}, \mathrm{P})$, where the finite sets $\mathbb{S}$ and $\mathbb{A}$ are referred to as the *state space* and *action space* respectively. Also, $\mathrm{R} : \mathbb{S} \times \mathbb{A} \times \mathbb{S} \to \mathbb{R}$ is the *reward function* which defines the state transition costs and $\mathrm{P} : \mathbb{S} \times \mathbb{A} \times \mathbb{A} \to [0, 1]$ is the *transition probability function*. A stationary randomized policy (SRP) $\pi$ is a probability mass function over the actions conditioned on the state space, *i.e.*, for $s \in \mathbb{S}$, we have $\pi(\cdot|s) \in [0, 1]^{|\mathbb{A}|}$ and $\sum_{a \in \mathbb{A}} \pi(a|s) = 1$. A policy determines the action to be taken at each discrete time step of an arbitrary realization of the MDP. In this paper, we employ a parametrized class of SRPs $\{\pi_w | w \in \mathbb{W} \subset \mathbb{R}^{k_2}\}$. We assume that $\mathbb{W}$ is compact.

By complying to a policy $\pi_w$, the behaviour of the MDP reduces to a Markov chain defined by the transition probabilities $\mathrm{P}_w(s, s') = \sum_{a \in \mathbb{A}} \pi_w(a|s)\mathrm{P}(s, a, s')$. The performance of a policy is usually quantified by a value function which is defined as the long-run $\gamma$-discounted transition costs ($\gamma \in [0, 1)$) incurred by the MDP while following the policy. The value function is formally defined as follows: $V^w(s) = \mathbb{E}_w \left[\sum_{t \in \mathbb{N}} \gamma^t \mathrm{R}(\mathbf{s}_t, \mathbf{a}_t, \mathbf{s}_{t+1}) | \mathbf{s}_0 = s\right], s \in \mathbb{S}$, where $\mathbb{E}_w[\cdot]$ is the expectation w.r.t. the probability distribution of the Markov chain induced by


B.U. Shankar et al. (Eds.): PReMI 2017, LNCS 10597, pp. 3–10, 2017.
https://doi.org/10.1007/978-3-319-69900-4_1

the policy $\pi_w$. And the primary goal in an RL setting is to find the optimal policy which solves $\arg\max_{w \in \mathbb{W}} V^w$ without any knowledge of the model parameters P and R. However, observations in the form of sample paths which are realizations of the MDP under any arbitrary policy are made available.

Classical approaches have complexities which scale polynomial in the cardinality of the state space and hence are intractable. This is commonly referred to as the curse of dimensionality. Hence, one has to resort to approximation techniques in order to achieve tractability. In this paper, for value function estimation, we consider the linear function approximation. Here, for a given policy $\pi_w$, we approximate its value function $V^w$ by projecting it on to the subspace $\{\Phi x | x \in \mathbb{R}^{k_1}\}$, where $\Phi = (\phi_1, \phi_2, \ldots, \phi_{k_1})^\top$, $k_1 \ll |\mathbb{S}|$ and $\phi_i \in \mathbb{R}^{|\mathbb{S}|}$, $1 \leq i \leq k_1$ called the *prediction features* are chosen a priori. In order for the projection to be well-defined, we require the following assumption:

**(A1):** *For each $w \in \mathbb{W}$, the Markov chain induced by the policy $\pi_w$ is ergodic.*

Under this assumption, one can define the weighted norm $\|\cdot\|_\nu$ as follows: For $V \in \mathbb{R}^{|\mathbb{S}|}$, $\|V\|_\nu = (\sum_{s \in \mathbb{S}} V^2(s)\nu(s))^{\frac{1}{2}}$, where $\nu$ is the limiting distribution (steady state distribution) of the given sample path. If the sample path is generated using a policy $\pi_{w_b}$, $w_b \in \mathbb{W}$ (referred to as the *behaviour policy*), then the limiting distribution is the stationary distribution $\nu_{w_b}$ of the Markov chain induced by the behaviour policy $\pi_{w_b}$. Therefore, the linear function approximation of the value function $V^w$ is defined as follows:

$$h_{w|w_b} \triangleq \arg\min_{x \in \mathbb{R}^{k_1}} \|\Phi x - V^w\|_{\nu_{w_b}} \tag{1}$$

$$\textit{In this paper, we solve the following problem: } w^* = \arg\max_{w \in \mathbb{W}} \mathbb{E}_{\nu_w}\left[h_{w|w}\right], \tag{2}$$

*where we assume that an infinitely long sample path $\{\mathbf{s}_0, \mathbf{a}_0, \mathbf{r}_0, \mathbf{s}_1, \mathbf{a}_1, \mathbf{r}_1, \mathbf{s}_2, \ldots\}$ generated by the behaviour policy $\pi_{w_b}$ ($w_b \in \mathbb{W} \subseteq \mathbb{R}^{k_2}$) is available.*

**(A2):** *The behaviour policy $\pi_{w_b}$, where $w_b \in \mathbb{W}$, satisfies the following condition:* $\pi_{w_b}(a|s) > 0$, $\forall s \in \mathbb{S}, \forall a \in \mathbb{A}$.

## 2 Proposed Algorithm

Our proposed approach has two components:

1. A stochastic approximation (SA) version of the cross entropy (CE) method to solve the control problem (2). The SA version of the CE method is a zero-order, incremental, adaptive and stable global optimization method.
2. A variation of the off-policy LSTD ($\lambda$) to compute the objective function values (i.e., $\mathbb{E}_{\nu_w}\left[h_{w|w}\right]$).

We describe the above two components in detail here.

### 2.1 Stochastic Approximation Version of the Cross Entropy Method

Cross entropy method [4,9] solves global optimization problems where the objective function does not possess good structural properties, *i.e.*, those of the kind: Find $x^* = \arg\max_{x \in \mathbb{X} \subset \mathbb{R}^d} J(x)$, where $J : \mathbb{X} \to \mathbb{R}$ is a bounded Borel measurable function ($J_l < J(x) < J_u$, $\forall x$). CE is a zero-order method which implies that the algorithm does not require the gradient or higher-order derivatives of the objective function in order to seek the optimal solution. CE method has found successful application in diverse domains which include continuous multi-extremal optimization [7], reinforcement learning [5,6] and several NP-hard problems [7,8].

CE method generates a sequence of model parameters $\{\theta_t\}_{t \in \mathbb{N}}$, $\theta_t \in \Theta$ (assumed to be compact) and a sequence of thresholds $\{\gamma_t \in \mathbb{R}\}_{t \in \mathbb{N}}$, $J_l \leq \gamma_t \leq J_u$ and the algorithm attempts to direct the sequence $\{\theta_t\}$ towards the degenerate distribution concentrated at the global optimum $x^*$ and $\gamma_t$ towards $J(x^*)$. The threshold $\gamma_{t+1}$ is usually taken as the $(1-\rho)$-quantile of $J$ w.r.t the PDF $f_{\theta_t}$. (For $\theta \in \Theta$, we denote by $\gamma_\rho(J, \theta)$ the $(1-\rho)$-quantile of $J$ w.r.t the PDF $f_\theta$). Henceforth, $\gamma_{t+1} = \gamma_\rho(J, \theta_t)$. And the model parameter $\theta_{t+1}$ is generated by projecting (w.r.t. the Kullback-Leibler (KL) divergence) on to the family of distributions $\mathcal{F} \triangleq \{f_\theta | \theta \in \Theta\}$, the zero-variance distribution concentrated in the region $\{J(x) \geq \gamma_{t+1}\}$ with respect to the PDF $f_{\theta_t}$. Thus,

$$\theta_{t+1} = \arg\min_{\theta \in \Theta} KL(f_\theta, g_t), \text{ where } g_t(x) = \frac{S(J(x)) f_{\theta_t}(x) \mathbb{I}_{\{J(x) \geq \gamma_{t+1}\}}}{\mathbb{E}_{\theta_t}\left[S(J(\mathsf{X})) \mathbb{I}_{\{J(\mathsf{X}) \geq \gamma_{t+1}\}}\right]} \tag{3}$$

with $S : \mathbb{R} \to \mathbb{R}_+$ is a positive, monotonically increasing function.

In this paper, we consider the Gaussian distribution as the family of distributions $\mathcal{F}$. In this case, the PDF is being parametrized as $\theta = (\mu, \Sigma)^\top$, where $\mu$ and $\Sigma$ are the mean and the covariance of the Gaussian distribution respectively. Also, one can solve the optimization problem (3) analytically to obtain the following update rule:

$$\left.\begin{aligned} \mu_{t+1} &= \frac{\mathbb{E}_{\theta_t}\left[\mathsf{g}_1(J(\mathsf{X}), \mathsf{X}, \gamma_{t+1})\right]}{\mathbb{E}_{\theta_t}\left[\mathsf{g}_0(J(\mathsf{X}), \gamma_{t+1})\right]} \triangleq \Upsilon_1(\theta_t, \gamma_{t+1}), \\ \Sigma_{t+1} &= \frac{\mathbb{E}_{\theta_t}\left[\mathsf{g}_2(J(\mathsf{X}), \mathsf{X}, \gamma_{t+1}, \mu_{t+1})\right]}{\mathbb{E}_{\theta_t}\left[\mathsf{g}_0(J(\mathsf{X}), \gamma_{t+1})\right]} \triangleq \Upsilon_2(\theta_t, \gamma_{t+1}). \end{aligned}\right\} \tag{4}$$

$$\left.\begin{aligned} \text{where } \mathsf{g}_0(J(x), \gamma) &\triangleq S(J(x)) \mathbb{I}_{\{J(x) \geq \gamma\}}, \\ \mathsf{g}_1(J(x), x, \gamma) &\triangleq S(J(x)) \mathbb{I}_{\{J(x) \geq \gamma\}} x, \\ \mathsf{g}_2(J(x), x, \gamma, \mu) &\triangleq S(J(x)) \mathbb{I}_{\{J(x) \geq \gamma\}} (x - \mu)(x - \mu)^\top. \end{aligned}\right\} \tag{5}$$

Thus the CE algorithm can be expressed as $\theta_{t+1} = (\Upsilon_1(\theta_t, \gamma_{t+1}), \Upsilon_2(\theta_t, \gamma_{t+1}))^\top$. However, this is the ideal scenario which is intractable due to the inability to compute the quantities $\mathbb{E}_{\theta_t}[\cdot]$ and $\gamma_\rho(\cdot, \cdot)$. There are multiple ways one can track the ideal CE method. In this paper, we consider the efficient tracking of the ideal CE method using the stochastic approximation (SA) framework proposed

in [1,2]. The SA version of the CE method consists of three stochastic recursions which are defined as follows:

$$\begin{aligned}\gamma_{t+1} =& \gamma_t - \beta_t \Delta\gamma_t(J(\mathsf{X}_{t+1})),\\ &\text{where } \Delta\gamma_t(y) \triangleq -(1-\rho)\mathbb{I}_{\{y\geq\gamma_t\}} + \rho\mathbb{I}_{\{y\leq\gamma_t\}}.\end{aligned} \tag{6}$$

$$\begin{aligned}\xi_{t+1}^{(0)} =& \xi_t^{(0)} + \beta_t \Delta\xi_t^{(0)}(\mathsf{X}_{t+1}, J(\mathsf{X}_{t+1})),\\ &\text{where } \Delta\xi_t^{(0)}(x,y) \triangleq \mathsf{g}_1(y,x,\gamma_t) - \xi_t^{(0)}\mathsf{g}_0(y,\gamma_t).\end{aligned} \tag{7}$$

$$\begin{aligned}\xi_{t+1}^{(1)} =& \xi_t^{(1)} + \beta_t \Delta\xi_t^{(1)}(\mathsf{X}_{t+1}, J(\mathsf{X}_{t+1})),\\ &\text{where } \Delta\xi_{j+1}^{(1)}(x,y) \triangleq \mathsf{g}_2(y,x,\gamma_t,\xi_t^{(0)}) - \xi_t^{(1)}\mathsf{g}_0(y,\gamma_t).\end{aligned} \tag{8}$$

Here, $\beta_t > 0$ and $\mathsf{X}_{t+1} \sim \widehat{f}_{\theta_t}$, where the mixture PDF $\widehat{f}_{\theta_t}$ is defined as $\widehat{f}_{\theta_t} \triangleq (1-\zeta)f_{\theta_t} + \zeta f_{\theta_0}$, $\zeta \in (0,1)$, $f_{\theta_0}$ is the initial PDF. The mixture approach facilitates extensive exploration of the solution space and prevents the model iterates from getting stranded in suboptimal solutions.

### 2.2 Computing the Objective Function $\mathbb{E}_{\boldsymbol{\nu_w}}\left[\boldsymbol{h_{w|w}}\right]$

In this paper, we employ the off-policy LSTD ($\lambda$) to approximate $h_{w|w}$ for a given policy parameter $w \in \mathbb{W}$. The procedure to estimate the objective function $\mathbb{E}_{\nu_w}\left[h_{w|w}\right]$ is formally defined in Algorithm 1. The *Predict* procedure in Algorithm 1 is almost the same as the off-policy LSTD algorithm. The recursion (step 9) attempts to estimate the objective function $\mathbb{E}_{\nu_w}\left[h_{w|w}\right]$ as follows:

$$\ell_{k+1}^w = \ell_k^w + \frac{1}{k}\Big(\mathbf{x}_k^\top \phi(\mathbf{s}_{k+1}) - \ell_k^w\Big), \tag{9}$$

**Algorithm 1.** Predict Function

1 **Input parameters:** $w \in \mathbb{W}$, $N \in \mathbb{N}$ ▶ *Input policy vector, Trajectory length*;
2 **Data**: *A priori chosen sample trajectory* $\{\mathbf{s}_0, \mathbf{a}_0, \mathbf{r}_0, \mathbf{s}_1, \mathbf{a}_1, \mathbf{r}_1, \mathbf{s}_2, \dots\}$ *generated using the behaviour policy* $\pi_{w_b}$;
3 $k = 0$;
4 **while** $k < N$ **do**
5 $\quad \mathbf{e}_{k+1} = \gamma\lambda\rho_k\mathbf{e}_k + \phi(\mathbf{s}_k)$; ▶ $\rho_k$ *is the sampling ratio,* $\rho_k = \frac{\pi_w(\mathbf{a}_k|\mathbf{s}_k)}{\pi_{w_b}(\mathbf{a}_k|\mathbf{s}_k)}$;
6 $\quad \mathbf{A}_{k+1} = \mathbf{A}_k + \frac{1}{k}\left(\mathbf{e}_k(\phi(\mathbf{s}_k) - \gamma\rho_k\phi(\mathbf{s}_{k+1}))^\top - \mathbf{A}_k\right)$;
7 $\quad \mathbf{b}_{k+1} = \mathbf{b}_k + \frac{1}{k}(\rho_k\mathbf{r}_k\mathbf{e}_k - \mathbf{b}_k)$;
8 $\quad \mathbf{x}_{k+1} = \mathbf{A}_{k+1}^{-1}\mathbf{b}_{k+1}$; ▶ *Prediction vector*;
9 $\quad \ell_{k+1}^w = \ell_k^w + \frac{1}{k}\Big(\mathbf{x}_k^\top\phi(\mathbf{s}_{k+1}) - \ell_k^w\Big)$; ▶ *Objective function estimation*;
10 $\quad k = k+1$;
11 **return** $\ell_N^w$; ▶ *Outputs after N iterations*;

For a given $w \in \mathbb{W}$, $\ell_k^w$ attempts to find an approximate value of the objective function $J(w)$. The following lemma formally characterizes the limiting behaviour of the iterates $\ell_k^w$.

**Lemma 1.** *For a given* $w \in \mathbb{W}, \ell_k^w \to \ell_*^w = \mathbb{E}_{\nu_{w_b}}\left[x_{w|w_b}^\top \phi(\mathbf{s})\right]$ *as* $k \to \infty$ *w.p. 1, where*

$$\begin{aligned} x_{w|w_b} =& A_{w|w_b}^{-1} b_{w|w_b} \text{ with } A_{w|w_b} = \Phi^\top D^{\nu_{w_b}}(\mathbb{I} - \gamma\lambda \mathrm{P}_w)^{-1}(\mathbb{I} - \gamma \mathrm{P}_w)\Phi \\ & \text{and } b_{w|w_b} = \Phi^\top D^{\nu_{w_b}}(\mathbb{I} - \gamma\lambda \mathrm{P}_w)^{-1}\mathrm{R}^w. \end{aligned} \tag{10}$$

*Here* $D^{\nu_{w_b}}$ *is the diagonal matrix with* $D_{ii}^{\nu_{w_b}} = \nu_{w_b}(i)$, $1 \le i \le |\mathbb{S}|$, *where* $\nu_{w_b}$ *is the stationary distribution of the Markov chain* $\mathrm{P}_{w_b}$ *induced by the behavior policy* $\pi_{w_b}$ *and* $\mathrm{R}^w \in \mathbb{R}^{|\mathbb{S}|}$ *with* $\mathrm{R}^w(s) \triangleq \Sigma_{s' \in \mathbb{S}, a \in \mathbb{A}} \pi_w(a|s)\mathrm{P}(s,a,s')\mathrm{R}(s,a,s')$.

*Remark 1.* By the above lemma, for a given $w \in \mathbb{W}$, the quantity $\ell_k^w$ tracks $J_b(w) \triangleq \mathbb{E}_{\nu_{w_b}}\left[x_{w|w_b}^\top \phi(\mathbf{s})\right]$. This is however different from the true objective function value $J(w) = \mathbb{E}_{\nu_w}\left[h_{w|w}\right]$, when $w \neq w_b$. This additional approximation error incurred is the extra cost one has to pay for the discrepancy in the policy which generated the sample path.

### 2.3 Proposed Algorithm

Our algorithm to solve the control problem (2) is illustrated in Algorithm 2. The following theorem shows that the model sequence $\{\theta_t\}$ and the averaged sequence $\{\bar{\theta}_t\}$ generated by Algorithm 2 converge to the degenerate distribution concentrated on the global maximum of the objective function $J_b$.

**Theorem 1.** *Let* $S(x) = exp(rx), r \in \mathbb{R}$. *Let* $\rho, \zeta \in (0,1)$ *and* $\theta_0 = (\mu_0, q\mathbb{I}_{k_2 \times k_2})^\top$, *where* $q \in \mathbb{R}_+$. *Also, let* $c_t \to 0$ *as* $t \to \infty$.. *Let the learning rates* $\bar{\beta}_t$ *and* $\beta_t$ *satisfy* $\sum_t \beta_t = \sum_t \bar{\beta}_t = \infty$, $\sum_t \beta_t^2 + \bar{\beta}_t^2 < \infty$. *Let* $\{\theta_t = (\mu_t, \Sigma_t)\}_{t \in \mathbb{N}}$ *and* $\{\bar{\theta}_t = (\bar{\mu}_t, \bar{\Sigma}_t)\}_{t \in \mathbb{N}}$ *be the sequences generated by Algorithm 2 and also assume* $\theta_t \in \Theta$, $\forall t \in \mathbb{N}$. *Let* $\bar{\beta}_t = o(\beta_t)$. *Let* $w_b \in \mathbb{W}$ *be the chosen behaviour policy vector. Also, let the assumptions (A1–A2) hold. Then, there exists* $q^* \in \mathbb{R}_+$ *and* $r^* \in \mathbb{R}_+$ *s.t.* $\forall q > q^*$ *and* $\forall r > r^*$,

$$\theta_t \to (w^{b*}, 0_{k_2 \times k_2})^\top, \bar{\theta}_t \to (w^{b*}, 0_{k_2 \times k_2})^\top \quad as\ t \to \infty, \quad w.p.1, \tag{13}$$

*where* $w^{b*} \in \arg\max_{w \in \mathbb{W}} J_b(w)$ *with* $J_b(w) \triangleq \mathbb{E}_{\nu_{w_b}}\left[x_{w|w_b}^\top \phi(\mathbf{s})\right]$.

## 3 Experimental Illustrations

The performance of our algorithm is evaluated on the chain walk MDP setting. This particular setting which is being proposed in [3] demonstrates the scenario where policy iteration is non-convergent when approximate value functions are

**Algorithm 2.** Proposed Algorithm

**Input parameters:** $\epsilon, \rho \in (0,1), \bar{\beta}_t, \beta_t, \zeta, c_t \in (0,1), c_t \to 0$, $\theta_0 = (\mu_0, \Sigma_0)^\top$, $\{N_t, t \in \mathbb{N}\}$ ▶ *Trajectory length rule chosen a priori*;

**Initialization:** $t = 0, \gamma_0 = 0$, $\xi_0^{(0)} = 0_{k_2 \times 1}$, $\xi_0^{(1)} = 0_{k_2 \times k_2}$, $T_0 = 0$, $\theta^p = NULL$, $\gamma_0^p = -\infty$;

**while** *stopping criteria not satisfied* **do**

  **Sample generation** : $\mathsf{W}_{t+1} \sim \widehat{f}_{\theta_t}(\cdot)$, where $\widehat{f}_{\theta_t} = (1-\zeta) f_{\theta_t} + \zeta f_{\theta_0}$;

  **Objective function estimation:** $\hat{J}(\mathsf{W}_{t+1}) = Predict(\mathsf{W}_{t+1}, N_{t+1})$;

  **Tracking** $\gamma_\rho(J_b, \widehat{\theta}_t)$**:** $\gamma_{t+1} = \gamma_t - \beta_t \Delta\gamma_t(\hat{J}(\mathsf{W}_{t+1}))$;

  **Tracking** $\Upsilon_1(\widehat{\theta}_t, \gamma_\rho(J_b, \widehat{\theta}_t))$**:** $\xi_{t+1}^{(0)} = \xi_t^{(0)} + \beta_t \Delta\xi_t^{(0)}(\mathsf{W}_{t+1}, \hat{J}(\mathsf{W}_{t+1}))$;

  **Tracking** $\Upsilon_2(\widehat{\theta}_t, \gamma_\rho(J_b, \widehat{\theta}_t))$**:** $\xi_{t+1}^{(1)} = \xi_t^{(1)} + \beta_t \Delta\xi_t^{(1)}(\mathsf{W}_{t+1}, \hat{J}(\mathsf{W}_{t+1}))$;

  **if** $\theta^p \neq NULL$ **then**

    $\mathsf{W}_{t+1}^p \sim \widehat{f}_{\theta^p} = (1-\zeta) f_{\theta^p} + \zeta f_{\theta_0}$;

    $\gamma_{t+1}^p = \gamma_t^p - \beta_t \Delta\gamma_t^p(\hat{J}(\mathsf{W}_{t+1}^p))$;

  **Threshold Comparison:** $T_{t+1} = T_t + c\left(\mathbb{I}_{\{\gamma_{t+1} \geq \gamma_t^p\}} - \mathbb{I}_{\{\gamma_{t+1} < \gamma_t^p\}} - T_t\right)$;

  **if** $T_{t+1} > \epsilon$ **then**

    **Save previous model** : $\theta^p = \theta_t$; $\gamma_{t+1}^p = \gamma_t$;

    **Update model:** $\theta_{t+1} = \theta_t + \beta_t \left((\xi_t^{(0)}, \xi_t^{(1)})^\top - \theta_t\right)$; (11)

    **Polyak averaging:** $\bar{\theta}_{t+1} = \bar{\theta}_t + \bar{\beta}_t \left(\theta_{t+1} - \bar{\theta}_t\right)$; (12)

  **else**

    $\gamma_{t+1}^p = \gamma_t^p$; $\theta_{t+1} = \theta_t$;

  $t = t + 1$;

employed instead of true ones. Here $|\mathbb{S}| = 300$, $\mathbb{A} = \{L, R\}$, $k_1 = 5$, $k_2 = 10$ and the discount factor $\gamma = 0.99$. The reward function $R(\cdot, \cdot, 100) = R(\cdot, \cdot, 200) = 1.0$ and zero for all other transitions. The transition probability kernel is given by

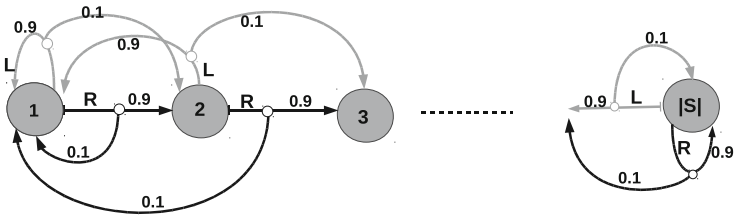


**Fig. 1.** Chain walk MDP

$P(s, L, s+1) = 0.1$, $P(s, L, s-1) = 0.9$, $P(s, R, s+1) = 0.9$ and $P(s, R, s-1) = 0.1$. We choose the behaviour policy vector $w_b = (0, 0, \ldots, 0)^\top$. We employ the Gibbs "soft-max" class of policies: $\pi_w(a|s) = \frac{e^{(w^\top \psi(s,a)/\tau)}}{\sum_{b \in \mathbb{A}} e^{(w^\top \psi(s,b)/\tau)}}$, where $\{\psi(s,a) \in \mathbb{R}^{k_2} | s \in \mathbb{S}, a \in \mathbb{A}\}$ is a given *policy feature set* and $\tau \in \mathbb{R}_+$ is fixed *a priori*. We employ radial basis functions (RBF) as both policy and prediction features, *i.e.*,

**Policy features**

$$\psi(s,a) = \begin{pmatrix} I_{\{a=L\}} e^{-\frac{(s-m_1)^2}{2.0v_1^2}} \\ \vdots \\ I_{\{a=L\}} e^{-\frac{(s-m_5)^2}{2.0v_5^2}} \\ I_{\{a=R\}} e^{-\frac{(s-m_1)^2}{2.0v_1^2}} \\ \vdots \\ I_{\{a=R\}} e^{-\frac{(s-m_5)^2}{2.0v_5^2}} \end{pmatrix}.$$

**Prediction features**

$$\phi_i(s) = e^{-\frac{(s-m_i)^2}{2.0v_i^2}},$$

where $m_i = 5 + 10(i-1), v_i = 5$, $1 \le i \le 5$ (Fig. 1).

The results are shown in Fig. 2.

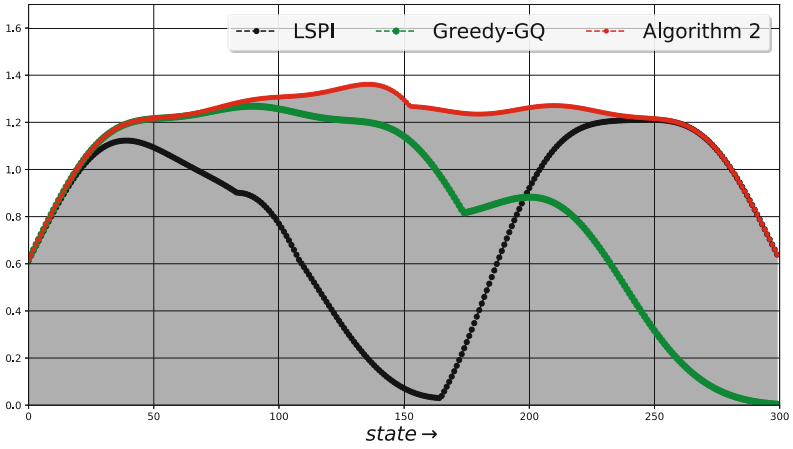


**Fig. 2.** The plot of the respective optimal value functions contrived by LSPI, Greedy-GQ and Algorithm 2 for the chain walk MDP setting. The optimal solutions of various algorithms are being developed by averaging over 10 independent trials. For Algorithm 2, we averaged the various optimal solutions obtained for different sample trajectories generated using the same behaviour policy, but with different initial states which are chosen randomly. Our approach (Algorithm 2) literally surpassed other algorithms in terms of its quality. The random choice of the initial state ineffectively favoured sufficient exploration of the state space which directly assisted in generating high quality solutions.

## 4 Conclusion

We propose an adaptation of the cross entropy method to solve the control problem in reinforcement learning under an information restricted setting, where only a single sample path generated using a priori chosen behaviour policy is available. The proposed algorithm is shown to converge to the solution which is globally optimal relative to the behaviour policy.

**Acknowledgement.** This work was supported in part by the Robert Bosch Centre for Cyber-Physical Systems, Indian Institute of Science, Bangalore.

## References

1. Joseph, A.G., Bhatnagar, S.: A randomized algorithm for continuous optimization. In: Winter Simulation Conference, WSC 2016, Washington, DC, USA, 11–14 December 2016, pp. 907–918 (2016)
2. Joseph, A.G., Bhatnagar, S.: Revisiting the cross entropy method with applications in stochastic global optimization and reinforcement learning. In: Frontiers in Artificial Intelligence and Applications, (ECAI 2016), vol. 285, pp. 1026–1034 (2016)
3. Koller, D., Parr, R.: Policy iteration for factored MDPS. In: Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence, pp. 326–334. Morgan Kaufmann Publishers Inc. (2000)
4. Kroese, D.P., Porotsky, S., Rubinstein, R.Y.: The cross-entropy method for continuous multi-extremal optimization. Methodol. Comput. Appl. Probab. **8**(3), 383–407 (2006)
5. Mannor, S., Rubinstein, R.Y., Gat, Y.: The cross entropy method for fast policy search. In: ICML, pp. 512–519 (2003)
6. Menache, I., Mannor, S., Shimkin, N.: Basis function adaptation in temporal difference reinforcement learning. Ann. Oper. Res. **134**(1), 215–238 (2005)
7. Rubinstein, R.: The cross-entropy method for combinatorial and continuous optimization. Methodol. Comput. Appl. Probab. **1**(2), 127–190 (1999)
8. Rubinstein, R.Y.: Cross-entropy and rare events for maximal cut and partition problems. ACM Trans. Model. Comput. Simul. (TOMACS) **12**(1), 27–53 (2002)
9. Rubinstein, R.Y., Kroese, D.P.: The Cross-Entropy Method: A Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation and Machine Learning. Springer, New York (2013)