A Convergent Recursive Least Squares Approximate Policy Iteration Algorithm for Multi-Dimensional Markov Decision Process with Continuous State and Action Spaces

Jun Ma and Warren B. Powell

Abstract— **In this paper, we present a recursive least squares approximate policy iteration (RLSAPI) algorithm for infinite-horizon multi-dimensional Markov decision process in continuous state and action spaces. Under certain problem structure assumptions on value functions and policy spaces, the approximate policy iteration algorithm is provably convergent in the mean. That is to say the mean absolute deviation of the approximate policy value function from the optimal value function goes to zero as successive approximation improves.**

I. INTRODUCTION

CURRENT theory for dynamic programming (DP) based algorithms focuses on finite state, finite action Markov decision process (MDP) problems. However, discrete representations of state and action spaces often suffer from the curse of dimensionality, typically limiting their application to problems with a relatively small number of dimensions. In addition, there are plenty of real world problems with continuous state and action spaces, to which algorithms developed for discrete problems cannot be applied. As a result, continuous function approximation is introduced to overcome the problem. Nevertheless, in contrast to the mature and elegant convergence theory supporting discrete MDP's [1], there is a paucity of theoretical results regarding the convergence of DP algorithms with function approximation applied to continuous state problems.

This paper is a synopsis of [2]. In this paper, we describe an approximate policy iteration algorithm with recursive least squares function approximation for infinite horizon Markov decision process problems with continuous state and action spaces. The algorithm can handle curses of dimensionality i.e. the state and decision variables are high dimensional continuous vectors and the expectation is hard to compute. It is provably convergent by imposing the significant assumption that the true value function is continuous and linearly parameterized with finitely many known basis functions. Extensions can be made to apply the convergence results to continuously differentiable value functions. The idea of policy iteration has been around for a while. [3] states that policy iteration runs into two difficulties for problems with infinite state and action spaces: (1) the existence of a sequence of policies generated by the algorithm, (2) the feasibility of obtaining accurate policy value functions in a computationally implementable way. Least squares updating combined with policy iteration is proposed in [4] to conquer the second difficulty for problems with discrete state space. The assumption we make on the policy value functions allows us to overcome the first difficulty and to provide a sound convergence theory of the algorithm applied to problems with continuous state and action spaces.

Jun Ma and Warren B. Powell are with the Department of Operations Research and Financial Engineering, Princeton University, Princeton, New Jersey (email: {junma, powell}@princeton.edu).

	S	A	R	E	Noise	Policy	Conv.
[10]	D	D	G	Y	S	VI	Y
[11]	D	D	G	Y	S	FP	Y
[12]	D	D	G	N	S	FP	Y
[13]	D	D	G	N	S	FP	Y
[4]	D	D	G	N	S	API	N
[14]	C	D	G	N	S	FP	Y
[15]	D	D	G	N	S	API	Y
[16]	NA	NA	G	Y	S	VI	Y
[17]	D	D	G	N	S	VI	Y
[18]	D	D	G	N	D	VI	Y
[19]	C	C	Q	N	D	API	Y
[21]	C	C	Q	N	D	VI/API	Y
[22]	C	C	Q	N	S(G)	VI	Y
[23]	C	D	G	Y	S	EPI	Y
[24]	C	D	G	N	S	VI	B
[25]	C	C	G	N	S	API	B
[26], [27]	C	D	G	N	S	API	B
[28]	C	D	G	N	S	VI	Y

Fig. 1. Table of some existing continuous function approximation algorithms and related convergence results

The rest of the paper is organized as follows: Section II briefly reviews the literature on continuous function approximation applied to MDP problems. In section III, we illustrate the details of a RLSAPI algorithm and go over the basics about MDP's and related preliminaries. In section IV, we present the convergence theorems of the algorithm and provide a sketch of proofs. Section V extends the convergence results to the more general case of unknown basis functions. Section VI provides the numerical experiment of the algorithm applied to a simple linear quadratic problem. Finally, section VII concludes and discusses future research directions.

II. LITERATURE REVIEW

Heuristic approaches of using compact representations in value function approximation have been around since the inception of DP to deal with the curses of dimensionality in

large-scale stochastic dynamic programming problems [5], [6], [7], [8], [9]. It was not until the early nineties that convergence theory of continuous function approximation was introduced. The literature on continuous function approximations and related convergence theories can be divided into the following three categories: 1) continuous function approximation of discrete problems, 2) approximation of continuous problems 3) examples of divergence for value iteration type algorithms.

A. Continuous function approximation of discrete problems

[10] first sets up a rigorous framework combining dynamic programming and compact representations using feature-based (linear approximation with basis functions) value iteration algorithms. [11], [12], [13], [14] prove convergence results for a variety of temporal difference learning TD(λ) algorithms with linear approximation for a constant policy, which reduces the problem to a discrete Markov chain. With some technical assumptions on the policy updating operator, [15] proves convergence of an API algorithm with Sarsa updating on linear state-action value function approximation for policy evaluation. [16] provides a convergence proof for fitted temporal difference learning (value iteration type) algorithm with function approximations that are contraction or expansion mappings (excluding linear regression and neural network). [17] presents a weaker convergence result that the SARSA(0) and V(0) (value iteration) algorithms with linear approximation converge to a bounded region almost surely. Residual gradient algorithm in [18] performs gradient descent on the mean squared Bellman residual to overcome the instability of updating value function or Q-function directly, and its convergence is guaranteed only for deterministic case.

B. Approximations of continuous problems

Papers that deal with function approximation algorithms directly applied to continuous problems is comparatively fewer specifically due to the instability feature of function approximations inherent in the continuous state space. Convergence results can only be found for the problem class of linear quadratic regulation (LQR) [19], [20], [21] (deterministic) and [22] (stochastic with Gaussian noise), average cost optimal control [23], batch reinforcement learning [24], [25], [26], [27] and the non-parametric approach with kernel regression [28].

C. Examples of divergence

Value iteration type algorithms with linear function approximation frequently fails to converge due to the exaggeration feature of the algorithm that small local changes at each iteration can lead to large global shifts of the approximation. [10], [29] describe counter-examples of TD(0) algorithms with linear approximation that either produces poor approximation or diverges even when the optimal value function can be perfectly represented by the linear approximator. [30] illustrates divergent behavior of value iteration algorithms with a variety of function approximation techniques such as polynomial regression, backpropagation and local weighted regression when the algorithm is applied to simple nonlinear problems.

The table in figure 1 is a brief summary of the algorithms categorized by their characteristics such as whether the state and action spaces are discrete or continuous (D/C), whether the contribution/reward function is quadratic or general (Q/G), whether the expectation can be computed exactly (Y/N), whether the problem is deterministic or stochastic (Gaussian noise) (D/S(G)), the type of algorithms including value iteration (VI), fixed policy (FP), exact/approximate policy iteration (E/API), and whether there is a convergence guarantee for the algorithm (Y/N) or performance bound (B). More details of the algorithms and their convergence properties are discussed in [2].

III. ALGORITHM AND RELATED PRELIMINARIES

In this section, we present some related preliminary concepts that are crucial in problem formulation and the convergence proof of the RLSAPI algorithm. We start with a brief review of the Markov decision process, followed by corresponding max operators and post-decision state variable and value function. Then, we turn to some useful properties of Markov chain with continuous state space and conclude with a detailed description of the algorithm.

A. Markov decision process

A Markov decision process is a sequential optimization problem where the goal is to find a policy that maximizes the sum of expected infinite-horizon discounted rewards. Let $S_t \in \mathcal{S}$ (state space) be the state of the system at time t, x_t be a vector-valued continuous decision (control) vector, $X^\pi(S_t)$ be a decision function corresponding to a policy $\pi \in \Pi$ (policy space), $C(S_t, x_t)$ be a contribution/reward function, and γ be a discount factor between 0 and 1. We shall, by convenient abuse of notation, denote π to be the policy function also and use it interchangeably with the corresponding decision function X^π. The system evolves according to some state transition function:

$$S_{t+1} = S^M(S_t, x_t, W_{t+1}), \tag{1}$$

where W_{t+1} represents the exogenous information that arrives during the time interval from t to $t+1$. The problem is to find the policy that solves

$$\sup_{\pi \in \Pi} \mathbb{E}\left\{ \sum_{t=0}^{\infty} \gamma^t C(S_t, X_t^\pi(S_t)) \right\}. \tag{2}$$

Since solving the objective function (2) directly is computationally intractable, Bellman's equation is introduced so that the optimal control can be computed recursively:

$$V(s) = \max_{x \in \mathcal{X}} \left\{ C(s,x) + \gamma \mathbb{E}\left[V(S^M(s,x,W)) | s \right] \right\}, \tag{3}$$

where $V(s)$ is the value function representing the value of being in state s. It is worth noting that the contribution function in (3) is often stochastic, in which case the Bellman's

equation becomes

$$V(s) = \max_{x \in \mathcal{X}} \left\{ \mathbb{E} \left[\hat{C}(s,x,W) + \gamma V(S^M(s,x,W)) | s \right] \right\}. \quad (4)$$

To work with continuous problems, we make some assumptions regarding the state, action and outcome spaces and the transition kernel function. Assume that the state space $\mathcal{S}$ is a convex and compact subset of $\mathbb{R}^m$, the action space $\mathcal{X}$ a compact subset of $\mathbb{R}^n$, the outcome space $\mathcal{W}$ a compact subset of $\mathbb{R}^l$ and $Q : \mathcal{S} \times \mathcal{X} \times \mathcal{W} \to \mathbb{R}$ is a continuous probability transition function. Let $\mathcal{C}^b(\mathcal{S})$ denote the space of all bounded continuous functions from $\mathcal{S}$ to $\mathbb{R}$, and $\mathcal{C}^b(\mathcal{S})$ is a complete metric space.

B. Contraction operators

In this section, we describe the contraction operators associated with Markov decision processes and their elementary properties [3] that are crucial in the convergence proofs of RLSAPI.

Definition 3.1 (Max operator): Let M be the max operator such that for all $s \in \mathcal{S}$ and $V \in \mathbf{C}^b(\mathcal{S})$,

$$MV(s) = \sup_{x \in \mathcal{X}} \{ C(s,x) + \gamma \int_{\mathcal{W}} Q(s,x,dw) V(S^M(s,x,w)) \},$$

where Q has the Feller Property (that is M maps $\mathbf{C}^b(\mathcal{S})$ into itself) and $S^M : \mathcal{S} \times \mathcal{X} \times \mathcal{W} \to \mathcal{S}$ is the continuous state transition function.

Definition 3.2 (Operator for a fixed policy π): Let M_π be the operator for a fixed policy π such that for all $s \in \mathcal{S}$,

$$M_\pi V(s) = C(s, X^\pi(s)) + \gamma \int_{\mathcal{W}} Q(s, X^\pi(s), dw) V(S^M(s, X^\pi(s), w))$$

where Q and S^M have the same property as in definition 3.1.

Proposition 3.1 (Monotonicity of M and M_π): For any $V_1, V_2 \in \mathcal{C}^b(\mathcal{S})$, if $V_1(s) \leq V_2(s)$ for all $s \in \mathcal{S}$, then for all $k \in \mathbb{N}$ and $s \in \mathcal{S}$

$$M^k V_1(s) \leq M^k V_2(s),$$

$$M_\pi^k V_1(s) \leq M_\pi^k V_2(s).$$

Proposition 3.2 (Fixed point of M and M_π): For any $V \in \mathcal{C}^b(\mathcal{S})$, $\lim_{k \to \infty} M^k V = V^*$ where V^* is the optimal value function, and V^* is the only solution to the equation $V = MV$. Similarly, for any $V \in \mathcal{C}^b(\mathcal{S})$, $\lim_{k \to \infty} M_\pi^k V = V^\pi$ where V^π is the value function by following policy π, and V^π is the only solution to the equation $V = M_\pi V$.

C. Post-decision state variable

When the underlying distribution of the evolution of the stochastic system is unknown, computing the expectation within the max operator M or M_π explicitly is impossible. Moreover, computation of the expectation may be intractable if the exogenous information W is a vector with known distribution. However, the difficulties can be circumvented by using the post-decision state variable [31], [32], [33]. To illustrate the idea, we first break the original transition function (1) into the two steps

$$S_t^x = S^{M,x}(S_t, x_t),$$

$$S_{t+1} = S^{M,W}(S_t^x, W_{t+1}).$$

Then, let $V^x(S_t^x)$ be the value of being in state S_t^x immediately after we made a decision. There is a simple relationship between the pre-decision value function $V(S_t)$ and post-decision value function $V^x(S_t^x)$ that is summarized as follows:

$$V(S_t) = \max_{x_t \in \mathcal{X}} \{ C(S_t, x_t) + \gamma V^x(S_t^x) \}, \quad (5)$$

$$V^x(S_t^x) = \mathbb{E}\{ V(S_{t+1}) | S_t^x \}. \quad (6)$$

By substituting 5 into 6, we obtain Bellman's equation for post-decision value function

$$V^x(S_t^x) = \mathbb{E}\{ \max_{x_{t+1} \in \mathcal{X}} \{ C(S_{t+1}, x_{t+1}) + \gamma V^x(S_{t+1}^x) \} | S_t^x \}. \quad (7)$$

RLSAPI works with the post-decision value functions of following a constant policy. By the assumption that the post-decision value functions are continuous and of the linear form $V^x(s^x|\theta) = \phi(s^x)^T \theta$ (where $\phi(s^x)$ is the vector of known basis functions and θ is the vector of linear parameters), it is enough to just estimate the linear parameters instead of the whole post-decision value function in the algorithm.

D. Markov chains with continuous state space

Continuous MDP problems require the use of Markov chains with continuous state space. The following definitions of irreducibility, invariant measure, recurrence and positivity [34] that all have familiar counterparts in discrete chains.

Definition 3.3 (ψ-Irreducibility for general space chains): We call a Markov chain Φ on state space $\mathcal{B}(\mathcal{S})$ φ-irreducible if there exists a measure φ on $\mathcal{B}(\mathcal{S})$ such that whenever $\varphi(A) > 0$ for $A \in \mathcal{B}(\mathcal{S})$, we have

$$\mathbb{P}_s\{\Phi \text{ ever enters } A\} > 0, \forall s \in \mathcal{S}$$

where $\mathbb{P}_s$ denotes the conditional probability on the event that the chain starts in state s. Let ψ be the maximal irreducibility measure among such measures.

Definition 3.4 (Harris recurrence): The set $A \in \mathcal{B}(\mathcal{S})$ is called Harris recurrent if

$$\mathbb{P}_s\{\Phi \in A \text{ infinitely often}\} = 1, \forall s \in \mathcal{S}.$$

.

A chain Φ is called Harris (recurrent) if it is ψ-irreducible and every set in

$$\mathcal{B}^+(\mathcal{S}) = \{A \in \mathcal{B}(\mathcal{S}) : \psi(A) > 0\}$$

is Harris recurrent.

Step 0: Initialization:

Step 0a Set the initial values of the value function parameters $\hat{\theta}^0$.

Step 0b Set the initial policy

$$\pi_0(s) = \arg\max_{x \in \Gamma(s)} \{C(s,x) + \gamma\phi(s^x)'\hat{\theta}^0\}.$$

Step 0c Set the iteration counter $n = 1$.

Step 0d Set the initial State S_0^0.

Step 1: Do for $n = 1, \ldots, N$,

Step 2: Do for $m = 1, \ldots, M$:

Step 3: Initialize $\hat{v}^m = 0$.

Step 4: Choose one step sample realization ω.

Step 5: Do the following:

Step 5a Set $x_m^n = \pi_{n-1}(S_m^n)$.

Step 5b Compute $S_m^{n,x} = S^{M,x}(S_m^n, x_m^n)$, $S_{m+1}^n = S^M(S_m^{n,x}, W_{m+1}(\omega))$ and $S_{m+1}^{n,x} = S^{M,x}(S_{m+1}^n, x_{m+1}^n)$

Step 5c Compute the corresponding basis function value in the general form $\phi(s_m^{n,x}) - \gamma\phi(s_{m+1}^{n,x})$.

Step 6: Do the following:

Step 6a Compute $\hat{v}_m = C(S_m^{n,x}, S_{m+1}^{n,x})$

Step 6b Update parameters $\hat{\theta}^{n,m}$ with LS/RLS method

Step 7: Update the parameter and the policy:

$$\hat{\theta}^n = \hat{\theta}^{n,M},$$

$$\pi_n(s) = \arg\max_{x \in \Gamma(s)} \{C(s,x) + \gamma\phi(s^x)'\hat{\theta}^n\}.$$

Step 8: Return the policy π_N and parameters $\hat{\theta}^N$.

Fig. 2. Infinite-horizon approximate policy iteration algorithm with recursive least squares method

Definition 3.5 (Invariant measure): Let $P(\cdot,\cdot)$ be the transition kernel of a chain Φ on the state space $\mathcal{S}$. A σ-finite measure μ on $\mathcal{B}(\mathcal{S})$ with the property

$$\mu(A) = \int_{\mathcal{S}} \mu(ds)P(s,A), \forall A \in \mathcal{B}(\mathcal{S})$$

will be called invariant.

Definition 3.6 (Positive chains): Suppose a chain Φ is ψ-irreducible and admits an invariant probability measure μ. Then Φ is called a positive chain.

E. Algorithm details

We summarize the RLSAPI algorithm in Figure 2. We make the following assumption and remark about the $\arg\max$ function at the end of the algorithm. This step is usually a multi-variable nonlinear programming problem that updates the policy from the post-decision value function of the previous inner iteration. The updated policy function returns a decision of vector form given any fixed input of the state variable. We assume that there is a tie-breaking rule or algorithm i.e. the nonlinear proximal point algorithm [35] that determines a unique solution to the $\arg\max$ function for all $f \in \mathcal{C}^b(\mathcal{S})$. As a result, the policy functions π_n's in the algorithms are well-defined single-valued continuous functions. Moreover, special problem structures such as strict concavity and differentiability of the value functions can significantly reduce the computational difficulty and complexity.

IV. CONVERGENCE RESULTS AND SKETCH OF PROOFS

We observe that there are both inner and outer loops in the RLSAPI algorithm. The inner loop is the policy evaluation step, while the outer loop corresponds to policy updates. We first consider an exact policy iteration algorithm, in which case both the inner and outer loop iteration counters goes to infinity. This conceptual algorithm is provably convergent almost surely (for details see [2]). To prove convergence of the exact algorithm, it is necessary to prove convergence of policy evaluation step in the inner loop, and the convergence result turns out to be useful in proving the mean convergence of RLSAPI.

A. Almost sure convergence for a fixed policy

For a fixed policy, a Markov decision problem is a Markov chain and Bellman's equation for the post-decision value function of following the fixed policy π (see (7)) becomes:

$$V^\pi(s) = \int_{\mathcal{S}^x} P^\pi(s,ds')(C^\pi(s,s') + \gamma V^\pi(s')), \tag{8}$$

where s, s' are post-decision state variables (we drop the superscript x for simplicity), $\mathcal{S}^x$ is the post-decision state space (compact since $\mathcal{S}$ and $\mathcal{X}$ are compact and the transition function is continuous), $P^\pi(\cdot,\cdot)$ is the transition probability function of the chain and $C^\pi(\cdot,\cdot)$ is the stochastic contribution/reward function. We further assume that the chain is a positive Harris chain with invariant measure μ_π (see definition 3.4, 3.5, 3.6) to ensure convergence. This strong assumption on the policy space is similar to the condition of having an ergodic Markov chain or an absorbing chain in the discrete case. If the problem structure does not guarantee this assumption, an exploration step such as adding a random exploration component to the policy function is necessary in an actual implementation of the algorithm.

Suppose $V^\pi(s) = (\phi^\pi(s))^T\theta^{*,\pi}$ where $\phi^\pi(s) = \left[\cdots, \phi_f^\pi(s), \cdots\right]$ is the vector of basis functions of dimension $F = |\mathcal{F}|$ (number of basis functions) and $f \in \mathcal{F}$ ($\mathcal{F}$ denotes the set of features). Bellman's equation (8) gives us

$$(\phi^\pi(s))^T\theta^{*,\pi} = \int_{\mathcal{S}^x} P^\pi(s,ds')[C^\pi(s,u) + \gamma(\phi^\pi(s'))^T\theta^{*,\pi}].$$

We can rewrite the recursion as

$$C^\pi(s,s') = (\phi^\pi(s) - \gamma\int_{\mathcal{S}^x} P^\pi(s,ds')\phi^\pi(s'))^T\theta^{*,\pi} + C^\pi(s,s') - \int_{\mathcal{S}^x} P^\pi(s,ds')C^\pi(s,s').$$

The integral is taken componentwise for the vector $\phi^\pi(s')$, so it feeds back a vector. It is similar in the situation when we put an integral sign in front of a matrix.

The model is known as the errors-in-variable model [36], since it has both observation and input noises. $C^\pi(s,s')$ is the observation, and $C^\pi(s,s') - \int_{\mathcal{S}^x} P^\pi(s,ds')C^\pi(s,s')$ is

the observation error. $\phi^\pi(s) - \gamma \int_{\mathcal{S}^x} P^\pi(s, ds')\phi^\pi(s')$ can be viewed as the input variable. Since the transition kernel function may be unknown, at iteration m, instead of having the exact input variable, we can only observe an unbiased sample $\phi_m^\pi - \gamma\phi_{m+1}^\pi$. Therefore, the errors in the input variables introduce bias into the regular formula of linear regression for estimating $\theta^{*,\pi}$. To eliminate the asymptotic bias, an instrumental variable ρ^π is introduced. Then, the m-th estimate of $\theta^{*,\pi}$ becomes

$$\theta_m^\pi = \left[\frac{1}{m}\sum_{i=1}^{m} \rho_i^\pi(\phi_i^\pi - \gamma\phi_{i+1}^\pi)'\right]^{-1}\left[\frac{1}{m}\sum_{i=1}^{m}\rho_i^\pi C_i^\pi\right], \quad (9)$$

where C_i^π is the i-th observation of the contribution. [36] proves convergence of the parameter estimates using (9) to the true values by assuming that the instrumental variable is correlated with the true input variable but uncorrelated with the observation error term and the correlation matrix between the instrumental variable and input variable is nonsingular and finite.

[13] chooses $\phi^\pi(s)$ as the instrumental variable ρ^π, and $\phi^\pi(s)$ turns out to be a good candidate for the continuous case as well. It is easy to check that it is uncorrelated with the observation error term in both cases. By assuming that the number of basis functions is the same as the number of states, [13] ensures that the correlation matrix between the input and instrumental variables is invertible. However, this assumption defeats the purpose of using continuous function approximation to overcome the curse of dimensionality. In the continuous case, the correlation matrix

$$\int_{\mathcal{S}^x} \mu_\pi(ds)\phi^\pi(s)(\phi^\pi(s) - \gamma\int_{\mathcal{S}^x} P^\pi(s, ds')\phi^\pi(s'))^T$$

is invertible, if we have orthonormal basis functions and $\gamma < \frac{1}{F}$ where F is the number of basis functions (for proof see [2]). Without loss of generality, we can assume $\gamma < \frac{1}{F}$. Since $\gamma \in (0,1)$, there exists $k \in \mathbb{N}$ such that $\gamma^{k-1} < \frac{1}{F}$ and the following recursion must also be satisfied if we keep substituting the Bellman's equation (8) back into itself $k-2$ times:

$$V^\pi(s_1) = \int_{\mathcal{S}^x\times\ldots\times\mathcal{S}^x}\prod_{i=1}^{k-1} P^\pi(s_i, ds_{i+1})\{\sum_{i=1}^{k-1}\gamma^{i-1}C^\pi(s_i, s_{i+1}) + \gamma^{k-1}V^\pi(s_k)\}.$$

Let $\tilde{P}^\pi(s_1, ds_k) = \prod_{i=1}^{k-1} P^\pi(s_i, ds_{i+1})$, $\tilde{C}^\pi(s_1, s_k) = \sum_{i=1}^{k-1}\gamma^{i-1}C^\pi(s_i, s_{i+1})$ and $\tilde{\gamma} = \gamma^{k-1}$. It is easy to check that μ_π is also an invariant measure for $\tilde{P}^\pi$. As a result, we have

$$V^\pi(s_1) = \int_{\mathcal{S}^x}\tilde{P}^\pi(s_1, ds_k)(\tilde{C}^\pi(s_1, s_k) + \tilde{\gamma}V^\pi(s_k)),$$

which is of the same form as in (8). Hence, we can make a minor modification to the algorithm to ensure the non-singularity of the correlation matrix by collapsing $k-1$ transitions ($s_1 \to s_k$) into 1 transition.

To compute estimates of $\theta^{*,\pi}$ more efficiently, we use recursive least squares to obtain the following well-known updating formulas:

$$\begin{aligned}
\epsilon_m^\pi &= C_m^\pi - (\phi_m^\pi - \gamma\phi_{m+1}^\pi)'\theta_{m-1}^\pi,\\
B_m^\pi &= B_{m-1}^\pi - \frac{B_{m-1}^\pi\phi_m^\pi(\phi_m^\pi - \gamma\phi_{m+1}^\pi)'B_{m-1}^\pi}{1 + (\phi_m^\pi - \gamma\phi_{m+1}^\pi)'B_{m-1}^\pi\phi_m^\pi},\\
\theta_m^\pi &= \theta_{m-1}^\pi + \frac{\epsilon_m^\pi B_{m-1}^\pi\phi_m^\pi}{1 + (\phi_m^\pi - \gamma\phi_{m+1}^\pi)'B_{m-1}^\pi\phi_m^\pi},
\end{aligned}$$

where θ_0^π and B_0^π must be specified before running the algorithm. θ_0^π can be any finite vector, while B_0^π is usually chosen to be βI for some small positive constant β.

The following convergence theorem of the RLS estimate of $\theta^{*,\pi}$ is an extension of the convergence theorem of LSTD(0) algorithm for a discrete chain in [13] applied to a Markov chain with continuous state space.

Theorem 4.1 (Almost sure convergence for a fixed policy): Suppose that for a fixed policy $\pi \in \Pi$, the infinite-horizon MDP problem can be reduced to a positive Harris chain Φ_π with invariant measure μ_π and the corresponding policy value function (post-decision) is continuous and linear in the parameters $\theta^{*,\pi}$. Further assume that the basis functions are continuous and orthonormal and that $1 + (\phi_m^\pi - \gamma\phi_{m+1}^\pi)'B_{m-1}\phi_m^\pi \neq 0$ for all m, then $\theta_m^\pi \to \theta^{*,\pi}$ almost surely.

Applying the law of large numbers for a positive Harris chain [35], the proof only involves simple calculations and manipulation of equation (8), so it is omitted here (for details see [2]).

B. Mean convergence of RLSAPI

In the RLSAPI algorithm, the estimated value function of the approximate policy in the policy evaluation step is random due to its dependence on both the Markov chain samples and iteration counter M of the inner loop. In other words, for fixed $s \in \mathcal{S}$, $\hat{V}^{\hat{\pi}_n}(s)$ is a random variable. Assuming that the state space (post-decision) $\mathcal{S}^x$ is compact and the norm is the sup norm $||\cdot||_\infty$ for continuous functions, we can show that RLSAPI converges in the mean. That is to say, the mean of the norm of the difference between the optimal value function and the estimated policy value function using approximate policy iteration converges to 0 as the successive approximations improves. We first consider a theorem regarding the mean convergence of a general API algorithm. Then, the mean convergence of RLSAPI is just a corollary from the generalization.

Theorem 4.2 (Mean convergence of an API algorithm): Let $\hat{\pi}_0$, $\hat{\pi}_1, \ldots, \hat{\pi}_n$ be the sequence of policies generated by some approximate policy iteration algorithm and let $\hat{V}^{\hat{\pi}_0}, \hat{V}^{\hat{\pi}_1}, \ldots, \hat{V}^{\hat{\pi}_n}$ be the corresponding approximate value functions. Further assume that, for each fixed policy $\hat{\pi}_n$, the MDP is reduced to a positive Harris chain that admits an invariant probability measure $\mu_{\hat{\pi}_n}$. Let $\{\epsilon_n\}$ be positive scalars that bound the mean errors between the approximate

and the true value functions over all iterations so that, for all $n \in \mathbb{N}$,

$$\mathbb{E}_{\mu_{\hat{\pi}_n}} ||\hat{V}^{\hat{\pi}_n} - V^{\hat{\pi}_n}||_\infty \leq \epsilon_n.$$

Suppose $\epsilon_n \to 0$ and $\lim_{n\to\infty} \sum_{i=0}^{n-1} \gamma^{n-1-i}\epsilon_i = 0$, e.g. $\epsilon_i = \gamma^i$. Then, this sequence eventually produces policies whose performance converges to the optimal performance in the mean:

$$\lim_{n\to\infty} \mathbb{E}_{\mu_{\hat{\pi}_n}} ||\hat{V}^{\hat{\pi}_n} - V^*||_\infty = 0.$$

The proof follows the same line as the proof of error bounds for approximate policy iteration with discrete and deterministic (or in the almost sure sense) value function approximations in [37]. In a general API algorithm presented in [37], error can accumulate in both policy evaluation and policy updating steps. In our case, policy updates are performed exactly, while policy evaluations are performed within certain error tolerances that diminish to 0 in the mean at a certain rate.

Corollary 4.1 (Mean convergence of RLSAPI): Theorem 4.2 holds for the RLSAPI algorithm.

It suffices to show that the mean error of policy evaluation in each inner loop can be made arbitrarily small. Since $\mathcal{S}^x$ is compact and the basis functions are continuous, $||\phi^{\pi_n}(s^x)||_2 \leq c$ for some finite positive constant c for all post-decision states. By our linear assumption on the post-decision value functions and relation (5) between pre- and post-decision value functions,

$$\mathbb{E}_{\mu_{\hat{\pi}_n}} ||\hat{V}^{\hat{\pi}_n} - V^{\hat{\pi}_n}||_\infty = \gamma \mathbb{E}_{\mu_{\hat{\pi}_n}} ||\phi(s^x)^T(\theta_{n,M} - \theta_n^*)||_\infty$$

By Cauchy-Schwartz inequality, we have

$$\mathbb{E}_{\mu_{\hat{\pi}_n}} ||\phi(s^x)^T(\theta_{n,M} - \theta_n^*)||_\infty \leq c\mathbb{E}_{\mu_{\hat{\pi}_n}} ||\theta_M^{\pi_n} - \theta^{*,\pi_n}||_2.$$

Recall that $\theta_M^{\pi_n}$ is a random vector of dimension F for fixed M, so

$$\mathbb{E}_{\mu_{\hat{\pi}_n}} ||\theta_M^{\pi_n} - \theta^{*,\pi_n}||_2 \leq \sum_{i=1}^{F} \mathbb{E}_{\mu_{\hat{\pi}_n}} |\theta_{M,i}^{\pi_n} - \theta_i^{*,\pi_n}|.$$

Since $\theta_M^{\pi_n} \to \theta^{*,\pi_n}$ almost surely for each i as $M \to \infty$ by theorem 4.1 and the parameter estimates are bounded due to the compact post-decision state space, $\theta_{M,i}^{\pi_n} \to \theta_i^{*,\pi_n}$ in L_1. As a result, in the policy evaluation step we can uniformly bound the mean difference between the approximate value function and the true function of the approximate policy. That is to say, for any $\epsilon_n > 0$, there exists $M_n \in \mathbb{N}$ such that

$$\mathbb{E}_{\mu_{\hat{\pi}_n}} ||\hat{V}^{\hat{\pi}_n} - V^{\hat{\pi}_n}||_\infty \leq \gamma c(\sum_{i=1}^{F} \mathbb{E}_{\mu_{\hat{\pi}_n}} |\theta_{M_n,i}^{\pi_n} - \theta_i^{*,\pi_n}|) \leq \epsilon_n.$$

Hence, we conclude that the RLSAPI algorithm converges in the mean by theorem 4.2.

It is worth making two remarks here. First, the number M_n is different after each policy update step (outer loop), since the underlying Markov chain changes as policy gets updated. In addition, the mean absolute deviation of a parameter from the true value is bounded by the sum of standard deviation and absolute bias of the parameter estimate, i.e.

$$\mathbb{E}_{\mu^{\hat{\pi}_n}} |\theta_{M,i}^{\pi_n} - \theta_i^{*,\pi_n}| \leq \sqrt{Var_{\mu^{\hat{\pi}_n}}(\theta_{M,i}^{\pi_n})} + |\mathbb{E}_{\mu^{\hat{\pi}_n}} \theta_{M,i}^{\pi_n} - \theta_i^{*,\pi_n}|.$$

Both the variance and bias converge to 0, and we can use them to determine a stopping rule for the inner loop. However, the variance and bias of the parameter estimate depend on the unknown true parameter and sample variance. Hence, we use the estimated variance and bias from samples in an actual implementation of the algorithm.

V. EXTENSION TO UNKNOWN BASIS FUNCTIONS

The mean convergence of RLSAPI requires the assumption that the value function of all policies in the policy space (including the optimal policy) are spanned by some finite set of known basis functions. We can relax the condition and extend the convergence result to countably infinite set of unknown basis functions by assuming that the value function of all policies are in some specific but general function spaces. For simplicity, we focus on 1-dimensional state space of a closed interval $[a, b]$. The function space we consider is $C^k[a, b]$, the set of all continuous function with up to k-th derivative.

A. Orthogonal polynomials

We first develop the concept of orthogonal polynomials by defining an inner product with respect to a weighting function w on the function space to be

$$< f, g >_w = \int_a^b f(s)g(s)w(s)ds.$$

This inner product defines a quadratic semi-norm $||f||_w^2 =< f, f >_w$. Let $G^w = \{g_n^w\}_{n=1}^\infty$ be a set of orthogonal basis functions with respect to w and $G_N^w = \{g_n^w\}_{n=1}^N$ be the finite subset of order N. Let $\mathcal{G}^w$ and $\mathcal{G}_N^w$ denote the function spaces spanned by G^w and G_N^w respectively. Given any $f \in C^k[a, b]$, the best least square approximation of f with respect to w onto G_N^w (denote it by f_N^w) is the solution to the following optimization problem

$$\inf_{g\in\mathcal{G}_N^w} \int_a^b (f(s) - g(s))^2 w(s)ds.$$

It is easy to check that $f_N^w = \sum_{n=1}^N \frac{<f,g_n^w>_w}{||g_n^w||_w^2} g_n^w$.

B. Chebyshev polynomials

We further restrict ourselves to one specific weighting function: Chebyshev weighting function $c(s) = \frac{1}{\sqrt{1-s^2}}$ on $[-1, 1]$. Chebyshev polynomials are desirable for approximating non-periodic smooth functions due to its uniform convergence property [38]. The weighting functions can be easily extended to an arbitrary closed interval $[a, b]$, e.g. the general Chebyshev weighting function is $c^g(s) = (1 - (\frac{2x-a-b}{b-a})^2)^{-\frac{1}{2}}$.

The family of Chebyshev polynomials $T = \{\tilde{t}_n\}_{n=0}^\infty$ is defined as $\tilde{t}_n(s) = \cos(n \cos^{-1} s)$ [38]. It can also be recursively defined as

$$\tilde{t}_0(s) = 1, \tilde{t}_1(s) = s,$$

$$\tilde{t}_{i+1}(s) = 2s\tilde{t}_i(s) - \tilde{t}_{i-1}(s), i \geq 1.$$

After normalizing, we have $t_0 = \frac{\tilde{t}_0}{\pi}$ and $t_i = \frac{2\tilde{t}_i}{\pi}$ for all $i \geq 1$. Let $\mathcal{C}_N$ be the function space spanned by the orthonormal basis set $T_N = \{t_n\}_{n=0}^N$.

C. Construction of basis functions from Chebyshev polynomials

Suppose μ^π is the invariant measure of a Markov chain of following a fixed policy π and $V^\pi \in C^k[-1,1]$. Further assume the invariant measure μ^π has a continuous density function f^π(i.e. $\mu^\pi(ds) = f^\pi(s)ds$), which is k-th differentiable and strictly positive on $[-1,1]$.

We first consider the following Chebyshev least square approximation problem for $\tilde{V}^\pi = V^\pi\sqrt{\frac{f^\pi}{c}}$,

$$\inf_{g \in \mathcal{C}_N} (\tilde{V}(s) - g(s))^2 c(s) ds.$$

The solution to this problem is the N-th degree Chebyshev least squares approximation

$$C_N(s) = \sum_{j=0}^{N} c_j t_j(s),$$

where $c_j = \int_{-1}^{1} \tilde{V}^\pi(s) t_j(s) c(s) ds$.

Let $T^\pi = \{t_n\sqrt{\frac{c}{f^\pi}}\}_{n=0}^\infty$. It is easy to check that T^π is an orthonormal basis set with respect to f^π. The least squares approximation problem for the value function of policy π onto the basis set $\mathcal{T}_N^\pi$ is

$$\inf_{g \in \mathcal{T}_N^\pi} \int_{-1}^{1} (V^\pi(s) - g(s))^2 f^\pi(s) ds.$$

The solution is easily verified to be $V_N^\pi = C_N\sqrt{\frac{c}{f^\pi}}$.

The parameter estimates for the constructed basis fucntions are not asymptotically unbiased anymore. However, it can be shown that given a desired approximation error ϵ in the inner loop of the algorithm, we can determine a finite set of orthonormal basis functions of order $N+1$ such that $E_{\mu_{\hat{\pi}_n}} ||\hat{V}_N^\pi - V^\pi||_\infty \leq \epsilon$, where $\hat{V}_N^\pi$ is a statistical estimate of V_N^π (for details of the proof, see [2]). Hence, corollary 4.1 applies.

However, there are two technical difficulties in practice. First, the exact invariant density function f^π is required to construct the basis set and to determine the order of the set. In addition, we need to know the bound of the k-th derivative of $\tilde{V}^\pi$ to determine the order of the basis set. In an actual implementation of the algorithm, we need to (1) call another procedure to produce a smooth approximation of f^π and (2) fix the bound of $(\tilde{V}^\pi)^{(k)}$ at some large number before evaluating the value function of the corresponding policy.

VI. Experiments

In this section, we consider a simple one-dimensional stochastic linear quadratic problem with Gaussian noise for numerical experiment, because it matches our assumption of known basis functions and we can compute the optimal solution analytically. Instead of focusing on compact state, action and outcome spaces, we let $\mathcal{S} = \mathcal{X} = \mathcal{W} = \mathbb{R}$. Let the transition function S^M be

$$S^M(s,x,w) = \rho(s+x) + w$$

where $0 < \rho < 1$ and $w \sim \mathcal{N}(0,1)$ and the contribution function C be

$$C(s,x) = -s^2 - x^2$$

Since we have a concave quadratic contribution function, it is well-known that the value function is also concave quadratic, i.e.

$$V(s) = \theta_0 + \theta_1 s + \theta_2 s^2$$

where $\theta_2 < 0$, and the optimal decision/control function is of linear form: $x^* = a + bs$.

In the numerical example, we fix the discount factor $\gamma = 0.95$ and $\rho = 0.9$, and we take 1000 iterations for each policy evaluation step and 100 iterations for policy updating. The following figures plot the deviation of RLSAPI estimated values of control function intercept a and slope b from their true values against the policy updating iteration counter N. From the figure, we clearly observe convergence behavior of the RLSAPI algorithm for the stochastic linear quadratic problem.

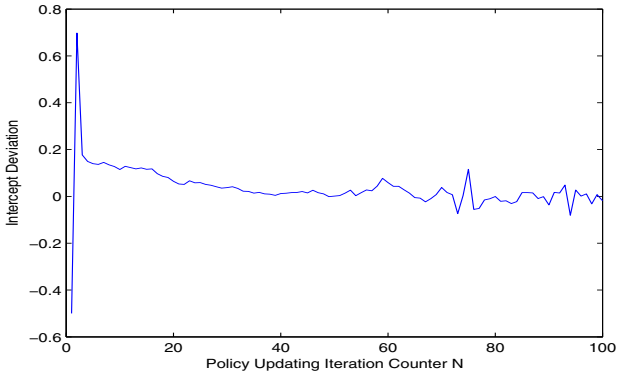

Fig. 3. Deviation of the intercept of the control from its true value 0.

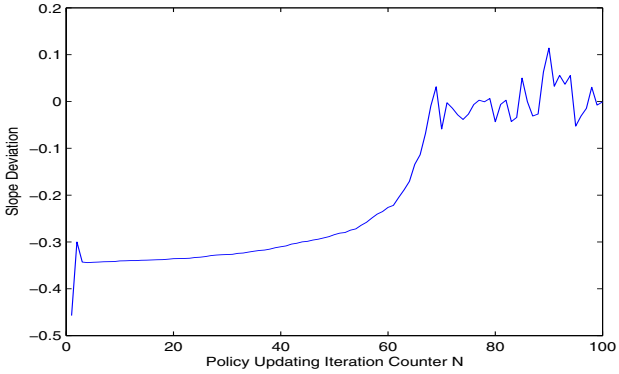

Fig. 4. Deviation of the slope of the control from its true value -0.5428.

VII. Conclusion and Discussions

An approximate policy iteration algorithm with recursive least squares updating is developed for infinite-horizon continuous Markov decision process problems with continuous state and action spaces. The algorithm is provably convergent in the mean under the assumptions that the stochastic system evolves according to a positive Harris chain for any fixed policy in the policy space and the true post-decision value functions of policies are continuous and spanned by a finite set of known basis functions. If the true policy value functions are not spanned by the set of basis functions, the algorithm may not converge to the optimal value function but a similar error bound between the approximate value function using the RLSAPI and the optimal value function can be derived as in [37]. Furthermore, the convergence results still applies if the true value functions are in some special function spaces e.g. $C^k[a,b]$ such that we can construct a finite set of orthonormal basis functions at each policy evaluation step. Our next step will be tackling the technical difficulties mentioned in section V. More specifically, we search for a procedure that incorporates the approximation step of the invariant density function into our algorithm without compromising the convergence property. In addition, convergent algorithms using other function approximators (parametric or non-parametric) including neural networks [37], local polynomial regression [39] and kernel-based reinforcement learning [28] will be explored to treat MDP problems with value functions of unknown form.

References

[1] M. L. Puterman, *Markov Decision Processes.* John Wiley & Sons, New York, 1994.

[2] J. Ma and W.B. Powell, "Convergence Proofs for Least Squares Policy Iteration Algorithm of High-Dimensional Infinite Horizon Markov Decision Process Problems," *working paper, http://castlelab.princeton.edu/Papers.html*, 2008.

[3] D.P. Bertsekas and S.E. Shreve, *Stochastic Optimal Control: The Discrete-Time Case.* Academic Press, Inc. Orlando, FL, USA, 1978.

[4] M.G. Lagoudakis and R. Parr, "Least-Squares Policy Iteration," *Journal of Machine Learning Research*, vol. 4, no. 6, pp 1107-1149, 2003.

[5] R. Bellman and I. Glicksberg and O. Gross, "On the optimal inventory equation," *Management Science*, vol. 2, no. 1, pp 83-104, 1955.

[6] R. Bellman and R. Kalaba and B. Kotkin, "Polynomial approximation—a new computational technique in dynamic programming: Allocation processes," *Mathematics of Computation*, vol. 17, no. 82, pp. 155-161, Mar. 1965.

[7] D. Reetz, "Approximate Solutions of a Discounted Markovian Decision Process," *Bonner Mathematische Schriften*, vol. 98, pp. 77-92, 1977.

[8] W. Whitt, "Approximations of Dynamic Programs, I," *Mathematics of Operations Research*, vol. 3, no. 3, pp. 231-243, 1978.

[9] P.J. Schweitzer and A. Seidmann, "Generalized polynomial approximations in Markovian decision processes," *Journal of mathematical analysis and applications*, vol. 110, no. 2, pp 568-582, 1985.

[10] J.N. Tsitsiklis and B. Van Roy, "Feature-based methods for large scale dynamic programming," *Machine Learning*, vol. 22, no. 1, pp 59-94, 1996.

[11] J.N. Tsitsiklis and B. Van Roy, "An analysis of temporal-difference learning with function approximation," *IEEE Transactions on Automatic Control*, vol. 42, no. 5, pp. 674-690, 1997.

[12] V.A. Papavassiliou and S.J. Russell, "Convergence of Reinforcement Learning with General Function Approximators," *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence*, pp. 748-757, 1999.

[13] S.J. Bradtke and A.G. Barto, "Linear Least-Squares algorithms for temporal difference learning," *Machine Learning*, vol. 22, no. 1, pp. 33-57, 1996.

[14] F.S. Melo and P. Lisboa and M.I. Ribeiro, "Convergence of Q-learning with linear function approximation," *Proceedings of the European Control Conference 2007*, 2007.

[15] T.J. Perkins and D. Precup, "A Convergent Form of Approximate Policy Iteration," *Advances In Neural Information Processing Systems*, pp. 1627-1634, 2003.

[16] G.J. Gordon, "Stable function approximation in dynamic programming" *Proceedings of the Twelfth International Conference on Machine Learning*, vol. 131, pp. 132-143, 1995.

[17] G.J. Gordon, "Reinforcement learning with function approximation converges to a region," *Advances in Neural Information Processing Systems*, vol. 13, pp. 1040-1046, 2001.

[18] L.C.Baird, "Residual algorithms: Reinforcement learning with function approximation," *Proceedings of the Twelfth International Conference on Machine Learning*, pp. 30-37, 1995.

[19] S.J. Bradtke and B.E. Ydstie and A.G. Barto, "Adaptive linear quadratic control using policy iteration," *American Control Conference*, 1994.

[20] S.J. Bradtke, "Reinforcement Learning Applied to Linear Quadratic Regulation," *Advances In Neural Information Processing Systems*, pp. 295-302, 1993.

[21] T. Landelius and H. Knutsson, "Greedy Adaptive Critics for LQR Problems: Convergence Proofs," *Neural Computation*, 1997.

[22] I. Szita, "Rewarding Excursions: Extending Reinforcement Learning to Complex Domains," *Ph.D. Thesis*, 2007.

[23] S.P. Meyn, "The policy iteration algorithm for average reward Markov decisionprocesses with general state space," *Automatic Control, IEEE Transactions on*, vol. 42, no. 12, pp. 1663-1680, 1997.

[24] R. Munos and C. Szepesvári, "Finite time bounds for sampling based fitted value iteration," *Proceedings of the 22nd international conference on Machine learning*, pp. 880-887, 2005.

[25] A. Antos and C. Szepesvári and R. Munos, "Value-Iteration Based Fitted Policy Iteration: Learning with a Single Trajectory," *Approximate Dynamic Programming and Reinforcement Learning, 2007. ADPRL 2007. IEEE International Symposium on*, pp. 330-337, 2007.

[26] A. Antos and R. Munos and C.Szepesvári, "Fitted Q-iteration in continuous action-space MDPs," *Proceedings of Neural Information Processing Systems Conference (NIPS), Vancouver, Canada*, 2007.

[27] A. Antos and C. Szepesvári and R. Munos, "Learning near-optimal policies with Bellman-residual minimization based fitted policy iteration and a single sample path," *Machine Learning*, vol. 71, no. 1, pp. 89-129, 2008.

[28] D. Ormoneit and Ś. Sen, "Kernel-Based Reinforcement Learning," *Machine Learning*, vol. 49, no. 2, pp. 161-178, 2002.

[29] D.P. Bertsekas, "A Counterexample to Temporal Differences Learning," *Neural Computation*, vol. 7, no. 2, pp. 270-279, 1995.

[30] J. Boyan and A.W. Moore, "Generalization in Reinforcement Learning: Safely Approximating the Value Function," *Advances In Neural Information Processing Systems*, pp. 369-376, 1995.

[31] B. Van Roy and D.P. Bertsekas and Y. Lee and J.N. Tsitsiklis, "A Neuro-Dynamic Programming Approach to Retailer Inventory Management," *Decision and Control, 1997., Proceedings of the 36th IEEE Conference on* vol. 4, 1997.

[32] R.S. Sutton and R.S.Barto, *Reinforcement Learning: An Introduction.* MIT Press, 1998.

[33] W.B. Powell, *Approximate Dynamic Programming: Solving the curses of dimensionality.* John Wiley and Sons, New York, 2007.

[34] S.P. Meyn and R.L. Tweedie, *Markov chains and stochastic stability.* Springer, 1993.

[35] J. Luque, *The nonlinear proximal point algorithm.* Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, 1986.

[36] P. Young, *Recursive estimation and time-series analysis: an introduction.* Springer-Verlag New York, Inc. New York, NY, USA, 1984.

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

[38] K.L. Judd, *Numerical Methods in Economics.* MIT Press, 1998.

[39] J. Fan and I. Gijbels, *Local Polynomial Modelling and Its Applications.* Chapman & Hall/CRCs, 1996.