Sampling Techniques for Zero-sum, Discounted Markov Games

Uday Savagaonkar, Edwin K. P. Chong, and Robert L. Givan

Abstract

In this paper, we first present a key approximation result for zero-sum, discounted Markov games, providing bounds on the state-wise loss and the loss in the sup norm resulting from using approximate Q-functions. Then we extend the policy rollout technique for MDPs to Markov games. Using our key approximation result, we prove that, under certain conditions, the rollout technique gives rise to a policy that is closer to the Nash equilibrium than the base policy. We also use our key result to provide an alternative analysis of a second sampling approach to Markov games known as sparse sampling. Our analysis implies the (already known) result that, under certain conditions, the policy generated by the sparse-sampling algorithm is close to the Nash equilibrium. We prove that the amount of sampling that guarantees these results is independent of the state-space–size of the Markov game.

I. Introduction

Previous work on Markov Decision Processes (MDPs) has provided many algorithms, such as value iteration and policy iteration, for finding (approximately) optimal policies for MDPs [1], and extensions of these algorithms for large state spaces [2], [3], [4], [5], [6], [7], [8].

Markov games are a natural multi-controller extension of MDPs for which the notion of a Nash equilibrium [9], where each player's policy is a "best response" to the other players' policies, is a widely accepted notion of optimality.

Patek and Bertsekas have established the convergence of value iteration and policy iteration for stochastic shortest-path games [10]. These algorithms require time at least polynomial in the state-space cardinality, and hence are impractical for Markov games with large state spaces. Previous online algorithms from the reinforcement-learning community are also impractical in large state spaces [11], [12], [13], [14].

Large state space MDP approaches include structure exploitation, value-function approximation, and sampling methods. Here, we focus on sampling algorithms, but for Markov games. The sampling algorithms we consider involve drawing random samples to estimate, for each possible initial action pair, the value of taking that initial action pair, and then either acting optimally, or following some given policy pair. We call this act "estimating the Q-function" (for the policy pair, if any). The resulting Q-function estimate defines a matrix game for the current state, and a current action is then chosen (possibly stochastically) by finding a (possibly mixed) Nash equilibrium for this game.

Our aim is to evaluate the policies that are formed using this "Nash look-ahead" technique. We present a key approximation result for discounted zero-sum games that provides bounds on the loss of the Nash look-ahead policy constructed using a sampled Q-function estimation. A similar result was given by Singh and Yee [15] for MDPs, using a different technique, but, as we will elaborate later, our result is more general, even as applied to MDPs, and can be used to infer a variant of the Singh and Yee result.

We then present two particular sampling algorithms. The first algorithm is an extension of the policy-rollout algorithm developed by Bertsekas and Castanon [4]. This algorithm starts

Uday Savagaonkar and Robert Givan are with the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907-1285. Email: {savagaon, givan}@ecn.purdue.edu.

Edwin K. P. Chong is with the Dept. of Electrical and Computer Engineering, Colorado State University, Fort Collins, CO 80523; Tel: 970-491-7858, Fax: 970-491-2249, Email: echong@engr.colostate.edu.

This research is supported in part by DARPA/ITO under contract F30602-00-2-0552 and by NSF grants 9977981-IIS, 0093100-IIS, 0098089-ECS, and 0099137-ANI. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of DARPA, NSF, or the U.S. Government.

with a pair of base policies, and produces a new policy by rolling out the base-policy pair. Using our approximation result, we establish that, under certain conditions, with finitely many finite-horizon samples, the new policy is closer to the Nash-optimal policy than the original policy. The number of samples required is independent of state-space size—even the MDP specialization of this result is new.

The second algorithm we present is the sparse-sampling Mrkov-game algorithm presented by Kearns et al. [16]. Kearns et al. proved that one can perform an amount of sampling that is independent of the size of the state space to obtain a near-Nash stochastic policy for a Markov game. We provide an alternate analysis of this result, leaveraging our main approximation theorem again.

II. Definitions, Notation, and Technical Background

In the rest of the paper, we use $\Pi(\mathbb{S})$ to denote the probability measures over the a $\mathbb{S}$. We use bold-face fonts to indicate random variables. For real functions f and g on domain $\mathbb{D}$, we write $f \leq g$ to indicate that $f(x) \leq g(x)$ for every x in $\mathbb{D}$. We write $|f|_\infty$ for $\sup_{x \in \mathbb{D}} |f(x)|$.

A zero-sum, discounted Markov game between players A and B with a discount factor γ is a tuple $\langle \mathbb{X}, \mathbb{A}, \mathbb{B}, \mathfrak{T}, R, x_0 \rangle$, where $\mathbb{X}$ is the (countable) state space, $\mathbb{A}$ ($\mathbb{B}$) is the finite action space for player A (B), $\mathfrak{T} : \mathbb{X} \times \mathbb{A} \times \mathbb{B} \to \Pi(\mathbb{X})$ is the transition function, $R : \mathbb{X} \times \mathbb{A} \times \mathbb{B} \to \mathbb{R}$ is the reward function, and $x_0 \in \mathbb{X}$ is the initial state. The aim of A (B) is to maximize (minimize) the γ-discounted cumulative reward. Let $\boldsymbol{f}(x, a, b)$ be a random state resulting from taking action pair $\langle a, b \rangle$ in state x, as specified by $\mathfrak{T}$.

An A-policy π^A for player A is a sequence of maps $\mu_i^{\pi^A} : \mathbb{X} \to \Pi(\mathbb{A})$, $i \geq 0$, specifying the probability distribution with which actions are chosen by A at time i. If $\mu_i^{\pi^A} = \mu_0^{\pi^A}$ for all i, then the policy is said to be stationary. We similarly define B-policies. Given map $\mu^A : \mathbb{X} \to \Pi(\mathbb{A})$, we use the bold-face notation $\boldsymbol{\mu}^A(x)$ to denote a random variable that has distribution $\mu^A(x)$. Given a policy π for either player, we use the notation μ_k^π to denote the k'th member of the sequence π. When the policy π is stationary, we will omit the subscript k.

Given a pair of policies $\langle \pi^A, \pi^B \rangle$ and state x, we define the value function $V_{\pi^A, \pi^B}(x)$ as expected value of the reward sum $\sum_{k=0}^\infty \gamma^k R(\boldsymbol{x}_k, \boldsymbol{\mu}_k^{\pi^A}(\boldsymbol{x}_k), \boldsymbol{\mu}_k^{\pi^B}(\boldsymbol{x}_k))$, where $\boldsymbol{x}_0 = x$, and $\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k, \boldsymbol{\mu}_k^{\pi^A}(\boldsymbol{x}_k), \boldsymbol{\mu}_k^{\pi^B}(\boldsymbol{x}_k))$. The space of value functions $V : \mathbb{X} \to \mathbb{R}$ is denoted by $\mathbb{V}$.

Given an A-policy π^A, a corresponding best-response policy for B is defined as a B-policy π^B that minimizes value $V_{\pi^A, \pi^B}(x)$ of the game in each state x, given that A plays policy π^A. We denote the set of all the best-response policies for B by $\mathrm{brB}(\pi^A)$. Similarly, we define $\mathrm{brA}(\pi^B)$. A pair of policies $\langle \pi^A, \pi^B \rangle$ is said to constitute a Nash equilibrium, if $\pi^B \in \mathrm{brB}(\pi^A)$ and $\pi^A \in \mathrm{brA}(\pi^B)$, in which case we write V^* for V_{π^A, π^B}. We call such policies *Nash policies* for the respective players.

For any A-policy π^A (or B-policy π^B), we define the security level $V^s_{\pi^A}$ (or $V^s_{\pi^B}$) as the minimum over all $\tilde{\pi}^B$ of $V_{\pi^A, \tilde{\pi}^B}(x)$ (or $\max_{\tilde{\pi}^A} V_{\tilde{\pi}^A, \pi^B}(x)$, respectively). We compare policies via security levels, so that A-policy $\tilde{\pi}^A$ is better than A-policy π^A in state x, if $V^s_{\tilde{\pi}^A}(x) \geq V^s_{\pi^A}(x)$, and is state-wise better if $V^s_{\tilde{\pi}^A} \geq V^s_{\pi^A}$. We say that policy $\tilde{\pi}^A$ is better than policy π^A in the sup norm if $\left| V^* - V^s_{\tilde{\pi}^A} \right|_\infty \leq \left| V^* - V^s_{\pi^A} \right|_\infty$.

For $V : \mathbb{X} \to \mathbb{R}$, and policies $\langle \pi_A, \pi_B \rangle$, define T, $T_{\mu_k^{\pi^A}, \mu_k^{\pi^B}}$, and $T^k_{\pi^A, \pi^B}$.

$$
\begin{aligned}
(TV)(x) &= \max_{z \in \Pi(\mathbb{A})} \min_{b \in \mathbb{B}} E\left[R(x, \boldsymbol{z}, b) + \gamma V(\boldsymbol{f}(x, \boldsymbol{z}, b)) \right]. \\
(T_{\mu_k^{\pi^A}, \mu_k^{\pi^B}} V)(x) &= E\left[R(x, \boldsymbol{\mu}_k^{\pi^A}(x), \boldsymbol{\mu}_k^{\pi^B}(x)) + \gamma V(\boldsymbol{f}(x, \boldsymbol{\mu}_k^{\pi^A}(x), \boldsymbol{\mu}_k^{\pi^B}(x))) \right], \text{ so that} \\
(T^0_{\pi^A, \pi^B} V)(x) &= V(x), \text{ and } (T^k_{\pi^A, \pi^B} V)(x) = (T^{k-1}_{\pi^A, \pi^B}(T_{\mu_k^{\pi^A}, \mu_k^{\pi^B}} V))(x).
\end{aligned}
$$

For a stationary policy pair $\langle \pi^A, \pi^B \rangle$, we will use the short-hand notation $T_{\pi^A, \pi^B} \triangleq T^1_{\pi^A, \pi^B}$.

The Q-function $Q_{\pi^A \pi^B} : \mathbb{X} \times \mathbb{A} \times \mathbb{B} \rightarrow \mathbb{R}$ for a pair of stationary policies $\langle \pi^A, \pi^B \rangle$, is defined as $Q_{\pi^A,\pi^B}(x,a,b) = E\left[R(x,a,b) + \gamma V_{\pi^A,\pi^B}(\boldsymbol{f}(x,a,b))\right]$, and we write $\mathbb{Q}$ for the space of functions $Q : \mathbb{X} \times \mathbb{A} \times \mathbb{B} \rightarrow \mathbb{R}$. Also, for Nash $\langle \pi^A, \pi^B \rangle$, we denote $Q_{\pi^A,\pi^B}(\cdot,\cdot,\cdot)$ by $Q^*(\cdot,\cdot,\cdot)$. We call a random variable that takes values in $\mathbb{Q}$ a *stochastic Q-function*.

Let $\mathrm{Nash}(M(\cdot,\cdot))$ be an operator that for any matrix $M(\cdot,\cdot) \in \mathbb{R}^{|\mathbb{A}| \times |\mathbb{B}|}$ computes a probability distribution pair $\langle \mu^A, \mu^B \rangle$ that achieves a Nash equilibrium for the zero-sum matrix game [9] described by $M(\cdot,\cdot)$. The operator $\mathrm{NashVal}(M(\cdot,\cdot))$ returns the value of the game when the $\mathrm{Nash}(M(\cdot,\cdot))$ distributions are used by the players to play the matrix game $M(\cdot,\cdot)$. Nash and $\mathrm{NashVal}$ can be computed in time polynomial in their argument sizes. Let $\mathrm{Nash}_A(M)$ and $\mathrm{Nash}_B(M)$ compute probability distributions used by the respective players at an equilibrium.

Throughout this paper, we assume that the reward function is bounded, i.e., $|R(x,a,b)| \leq R_{\max}$ for some $R_{\max} \in \mathbb{R}$ and all $x \in \mathbb{X}$, $a \in \mathbb{A}$, and $b \in \mathbb{B}$. The value function for any $\langle \pi^A, \pi^B \rangle$ satisfies, at any $x \in \mathbb{X}$, $|V_{\pi^A,\pi^B}(x)| \leq V_{\max} \triangleq R_{\max}/(1-\gamma)$. We also write e for the value function whose value is 1 for every state.

We prove the following technical background propositions in the full paper, following analogous proofs for MDPs by others [1], [17]. Here, $V(\cdot)$ and $V'(\cdot)$ are arbitrary value functions, and $\langle \pi^A, \pi^B \rangle$ is an arbitrary policy pair.

Proposition 1. Suppose $V(x) \leq V'(x)$ for all $x \in \mathbb{X}$. We then have $(T^k V)(x) \leq (T^k V')(x)$ for all $x \in \mathbb{X}$. Also, we have $(T^k_{\pi^A,\pi^B} V)(x) \leq (T^k_{\pi^A,\pi^B} V')(x)$, for all $x \in \mathbb{X}$.

Proposition 2. For any $r \in \mathbb{R}$, and e the unit value function defined above,

$(T^K(V + re))(x) = (T^K V)(x) + \gamma^K r$, and $(T^K_{\pi^A,\pi^B}(V + re))(x) = (T^K_{\pi^A,\pi^B} V)(x) + \gamma^K r$.

Proposition 3. $\sup_{\pi^A} \inf_{\pi^B} (T^K_{\pi^A,\pi^B} V)(x) = (T^K V)(x)$.

Proposition 4. (Value iteration converges) $\lim_{N\rightarrow\infty} (T^N V)(x_0) = V^*(x_0)$.

Proposition 5. $\lim_{N\rightarrow\infty} (T^N_{\pi^A,\pi^B} V)(x_0) = V_{\pi^A,\pi^B}(x_0)$.

Proposition 6. For stationary π^A and π^B, $V_{\pi^A,\pi^B} = T_{\pi^A,\pi^B} V_{\pi^A,\pi^B}$.

Proposition 7. The Nash value of the game satisfies Bellman's equation, $V^* = TV^*$.

Proposition 8. Suppose V and V' are bounded. For all $k \in \mathbb{N}$, we then have that $\max_{x\in\mathbb{X}} \left|(T^k V)(x) - (T^k V')(x)\right| \leq \gamma^k \max_{x\in\mathbb{X}} |V(x) - V'(x)|$.

Proposition 9. Let $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_N$ be i.i.d. random variables satisfying $|\boldsymbol{X}_i| \leq X_{\max}$ and $E\boldsymbol{X}_i = \mu$. Then, $P\left[\left|\frac{1}{N}\sum_{i=1}^N \boldsymbol{X}_i - \mu\right| \leq \lambda\right] \geq 1 - 4e^{-\lambda^2 N/8X^2_{\max}}$.

III. An Approximation Result for Markov Games

A. The Concept of Look-ahead

Policy selection in MDP problems typically critically involves a process of one-step look-ahead relative to a given (possibly estimated) value function, followed by a maximization to select the best action. This process selects the action that will perform the best if future value is given by the specified value function.

The corresponding process in Markov games is more complicated. One step look-ahead can integrate the system dynamics to convert a given value function into a Q-function; however, maximization is insufficient for action selection, as the opponent action is unknown. Our approach to analyzing Markov games leverages the idea that the Q function defines a matrix game in the current state. This matrix game can be solved to get an "equilibrium action distribution" analogous to the maximal-valued action in the MDP case (e.g., see [10]). Here, we are concerned with evaluating the effect of sampling error on the resulting policy improvement—we have a stochastic Q-function rather than a Q-function. Analysis of the effects of sampling is critical for large state-space games, where exact policy improvement and/or policy iteration simply cannot be carried out in practice.

We now formally define the important concept of one-step look-ahead for Markov games in the presence of sampling; this look-ahead function converts a given distribution F over Q-functions (typically, from a sampling algorithm) into a distribution over actions. Given a distribution F over $\mathbb{Q}$, the *one-step look-ahead policy* $\mathrm{lookaheadA}(F)$ for A is the policy that chooses actions in state x according to the probability distribution $E\ \mathrm{Nash}_A(\boldsymbol{Q}(x,\cdot,\cdot))$, where $\boldsymbol{Q}$ is a $\mathbb{Q}$-valued random variable with distribution F.

The stochastically described matrix $\boldsymbol{Q}(x,\cdot,\cdot)$ can be viewed as a matrix-game encapsulation of the expected future following each available action pair, and the $\mathrm{lookaheadA}(F)$ policy chooses its actions by solving this game. Now suppose that we have an algorithm that takes as an input the current state x, and outputs a random matrix $\boldsymbol{M} \in \mathbb{R}^{|\mathbb{A}|\times|\mathbb{B}|}$ distributed as specified by F in state x. Then the policy $\tilde{\pi}^A$ (and $\tilde{\pi}^B$) can be generated as follows. At every decision epoch, observe the current state x and generate a random matrix $\boldsymbol{M}$ using the given algorithm, and compute the distribution $\mathrm{Nash}_A(\boldsymbol{M})$, and choose an action a according this distribution. The sampling algorithms we consider use this technique for generating policies.

B. *The Main Theorem*

Our main theorem provides bounds on the state-wise loss suffered when following a policy selected by Nash look-ahead using sampled Q-functions. When the sampled Q-functions are generated from a particular base policy pair (rather than by estimating the optimal value directly), as in policy rollout, the theorem also gives bounds on the change in loss in the sup norm, relative to that base policy pair. Our result for discounted Markov games is more general than the MDP result proven by Singh and Yee [15] in four aspects:

1. We extend their MDP result to Markov games.
2. We relax their finite–state-space restriction to allow countable state-spaces[1].
3. Our result applies to approximating Q-functions of arbitrary policies, not just Q^*.
4. Their proof does not support any bound on the sup-norm loss of the look-ahead policy.

Because of aspects 3 and 4, we use a different proof method, by non-trivially extending the techniques of [10] to include bounds on the effects of Q-function approximation.

Theorem 1. Let π^A be a stationary policy for A, and let π^B be a best-response policy for π^A. Let F be a Q-function distribution such that any $\boldsymbol{Q}$ distributed according to F satisfies $\left|Q_{\pi^A,\pi^B}(x,\cdot,\cdot) - \boldsymbol{Q}(x,\cdot,\cdot)\right|_\infty < \epsilon$, for any $x \in \mathbb{X}$, with probability at least $1-\delta$, and is a.s. bounded by $V_{\max}$, i.e., $P[|\boldsymbol{Q}|_\infty \leq V_{\max}] = 1$. Let $\tilde{\pi}^A = \mathrm{lookaheadA}(F)$. Then

1. $V^s_{\pi^A}(x)$ is bounded above by $V^s_{\tilde{\pi}^A}(x) + 2(\epsilon + 2\delta V_{\max})/(1-\gamma)$ for all $x \in \mathbb{X}$, and
2. $|V^* - V^s_{\tilde{\pi}^A}|_\infty$ is bounded above by $\gamma(|V^* - V^s_{\pi^A}|_\infty) + 2(\epsilon + 2\delta V_{\max})/(1-\gamma)$.

Proof. Let $\tilde{\pi}^B \in \mathrm{brB}(\tilde{\pi}^A)$ be a stationary best-response policy to $\tilde{\pi}^A$. Let us denote by $\boldsymbol{Q}$ a stochastic Q-function having distribution F. We wish to compare the security level of π^A, i.e., V_{π^A,π^B}, with the security level achieved by $\tilde{\pi}^A$, i.e., $V_{\tilde{\pi}^A,\tilde{\pi}^B}$, and show that the latter approximately dominates the former. To do so, we define an "approximately" increasing sequence of "approximately" intermediate value functions, starting with the expected value of the "look-ahead game" described by the $\boldsymbol{Q}(x,\cdot,\cdot)$ matrix, and ending at the security level of $\tilde{\pi}^A$. Let $V_1(x)$ be $E\,\mathrm{NashVal}(\boldsymbol{Q}(x,\cdot,\cdot))$ and V_{K+1} be $T_{\tilde{\pi}^A,\tilde{\pi}^B}V_K$.

This sequence necessarily converges to the security level of $\tilde{\pi}^A$—it remains to show that the sequence is approximately increasing, and that V_1 approximately dominates the security level of π^A, analyzing how the approximation bounds sum over the sequence. We start with latter.

[1] We believe that, with suitable regularity assumptions, this result extends immediately to continuous state spaces, but we do not explore that claim further here.

Let us denote the event $|\boldsymbol{Q}(x,\cdot,\cdot) - Q_{\pi^A,\pi^B}(x,\cdot,\cdot)|_\infty < \epsilon$ by $\mathfrak{C}$. Then, by our choice of F, we have $P\left[\mathfrak{C}\right] > 1-\delta$. Also, we denote the complement event by $\mathfrak{C}^c$. Thus, $P[\mathfrak{C}^c] < \delta$. Now,

$$
\begin{aligned}
V_1(x) &= E\,\mathrm{NashVal}(\boldsymbol{Q}(x,\cdot,\cdot)) = E\left[\max_{\mu^A\in\Pi(\mathbb{A})}\ \min_{b\in\mathbb{B}} E\left[\boldsymbol{Q}(x,\boldsymbol{\mu}^A,b)\middle|\boldsymbol{Q}\right]\right] \\
&\geq E\left[\min_{b\in\mathbb{B}} E\left[\boldsymbol{Q}(x,\boldsymbol{\mu}^{\pi^A}(x),b)\middle|\boldsymbol{Q}\right]\middle|\mathfrak{C}\right] P\left[\mathfrak{C}\right] \\
&\quad + E\left[\min_{b\in\mathbb{B}} E\left[\boldsymbol{Q}(x,\boldsymbol{\mu}^{\pi^A}(x),b)\middle|\boldsymbol{Q}\right]\middle|\mathfrak{C}^c\right] P\left[\mathfrak{C}^c\right] \\
&\geq E\left[\min_{b\in\mathbb{B}} E\left[\boldsymbol{Q}(x,\boldsymbol{\mu}^{\pi^A}(x),b)\middle|\boldsymbol{Q}\right]\middle|\mathfrak{C}\right] (1-P\left[\mathfrak{C}^c\right]) - P\left[\mathfrak{C}^c\right] V_{\max} \\
&\geq E\left[\min_{b\in\mathbb{B}} E\left[Q_{\pi^A,\pi^B}(x,\boldsymbol{\mu}^{\pi^A}(x),b) - \epsilon\middle|\boldsymbol{Q}\right]\middle|\mathfrak{C}\right] - 2\delta V_{\max} \\
&= E\left[V_{\pi^A,\pi^B}(x)\middle|\mathfrak{C}\right] - \epsilon - 2\delta V_{\max} \quad = \quad V^s_{\pi^A}(x) - \epsilon - 2\delta V_{\max},
\end{aligned}
$$

where the first step follows by our choice of $\tilde{\pi}^A$ and $\tilde{\pi}^B$, and the next to last step follows from the fact that no policy can outperform the best-response policy π^B. We now show that the sequence of value functions is approximately increasing, starting with the preliminary observation that, when $|\boldsymbol{Q}(x,\cdot,\cdot) - Q_{\pi^A,\pi^B}(x,\cdot,\cdot)| < \epsilon$ (i.e., the event $\mathfrak{C}$ occurs), we have, for any $x\in\mathbb{X}$, $a\in\mathbb{A}$, and $b\in\mathbb{B}$,

$$
\begin{aligned}
E\left[R(x,a,b) + \gamma V_1(\boldsymbol{f}(x,a,b))\right] &\geq E\left[R(x,a,b) + \gamma V^s_{\pi^A}(\boldsymbol{f}(x,a,b)) - \gamma(\epsilon + 2\delta V_{\max})\right] \\
&= Q_{\pi^A,\pi^B}(x,a,b) - \gamma(\epsilon + 2\delta V_{\max}) \\
&\geq \boldsymbol{Q}(x,a,b) - \epsilon - \gamma(\epsilon + 2\delta V_{\max}).
\end{aligned}
$$

Now, writing $\nu(\boldsymbol{Q},x)$ for $\mathrm{Nash}_A(\boldsymbol{Q}(x,\cdot,\cdot))$, and noting that $\mu^{\tilde{\pi}^A}(x)$ is $E\,\mathrm{Nash}_A(\boldsymbol{Q}(x,\cdot,\cdot))$,

$$
\begin{aligned}
V_2(x) &= (T_{\tilde{\pi}^A,\tilde{\pi}^B}V_1)(x) = E\left[R(x,\boldsymbol{\mu}^{\tilde{\pi}^A}(x),\boldsymbol{\mu}^{\tilde{\pi}^B}(x)) + \gamma V_1(\boldsymbol{f}(x,\boldsymbol{\mu}^{\tilde{\pi}^A}(x),\boldsymbol{\mu}^{\tilde{\pi}^B}(x)))\right] \\
&= E\left[E\left[R(x,\boldsymbol{\nu}(\boldsymbol{Q},x),\boldsymbol{\mu}^{\tilde{\pi}^B}(x)) + \gamma V_1(\boldsymbol{f}(x,\boldsymbol{\nu}(\boldsymbol{Q},x),\boldsymbol{\mu}^{\tilde{\pi}^B}(x)))\middle|\boldsymbol{Q}\right]\right] \\
&= E\left[E\left[R(x,\boldsymbol{\nu}(\boldsymbol{Q},x),\boldsymbol{\mu}^{\tilde{\pi}^B}(x)) + \gamma V_1(\boldsymbol{f}(x,\boldsymbol{\nu}(\boldsymbol{Q},x),\boldsymbol{\mu}^{\tilde{\pi}^B}(x)))\middle|\boldsymbol{Q}\right]\middle|\mathfrak{C}\right] P\left[\mathfrak{C}\right] \\
&\quad + E\left[E\left[R(x,\boldsymbol{\nu}(\boldsymbol{Q},x),\boldsymbol{\mu}^{\tilde{\pi}^B}(x)) + \gamma V_1(\boldsymbol{f}(x,\boldsymbol{\nu}(\boldsymbol{Q},x),\boldsymbol{\mu}^{\tilde{\pi}^B}(x)))\middle|\boldsymbol{Q}\right]\middle|\mathfrak{C}^c\right] P\left[\mathfrak{C}^c\right] \\
&\geq E\left[E\left[R(x,\boldsymbol{\nu}(\boldsymbol{Q},x),\boldsymbol{\mu}^{\tilde{\pi}^B}(x)) + \gamma V_1(\boldsymbol{f}(x,\boldsymbol{\nu}(\boldsymbol{Q},x),\boldsymbol{\mu}^{\tilde{\pi}^B}(x)))\middle|\boldsymbol{Q}\right]\middle|\mathfrak{C}\right] P\left[\mathfrak{C}\right] \\
&\quad - P\left[\mathfrak{C}^c\right] V_{\max} \\
&\geq E\left[E\left[\boldsymbol{Q}(x,\boldsymbol{\nu}(\boldsymbol{Q},x),\boldsymbol{\mu}^{\tilde{\pi}^B}(x))\middle|\boldsymbol{Q}\right]\middle|\mathfrak{C}\right] P\left[\mathfrak{C}\right] - \epsilon - \gamma(\epsilon + 2\delta V_{\max}) - P\left[\mathfrak{C}^c\right] V_{\max} \\
&\geq E\left[E\left[\boldsymbol{Q}(x,\boldsymbol{\nu}(\boldsymbol{Q},x),\boldsymbol{\mu}^{\tilde{\pi}^B}(x))\middle|\boldsymbol{Q}\right]\right] - \epsilon - \gamma(\epsilon + 2\delta V_{\max}) - 2P\left[\mathfrak{C}^c\right] V_{\max} \\
&\geq E\left[\min_{\mu^B} E\left[\boldsymbol{Q}(x,\boldsymbol{\nu}(\boldsymbol{Q},x),\boldsymbol{\mu}^B(x))\middle|\boldsymbol{Q}\right]\right] - \epsilon - \gamma(\epsilon + 2\delta V_{\max}) - 2\delta V_{\max} \\
&= V_1(x) - (1+\gamma)(\epsilon + 2\delta V_{\max}),\ \text{by the definitions of } \nu() \text{ and } V_1.
\end{aligned}
$$

Using this fact, Propositions 1 and 2 imply that for all $K \geq 1$, $V_{K+1} \geq V_K - \gamma^{k-1}(1+\gamma)(\epsilon + 2\delta V_{\max})e$. Also, as shown above, $V_1 \geq V^s_{\pi^A} - (\epsilon + 2\delta V_{\max})e$. Then, by Proposition 5,

$$\begin{aligned} V^s_{\tilde{\pi}^A} &= \lim_{K\to\infty} V_K \;\geq\; V^s_{\pi^A,\pi^B} - (\epsilon + 2\delta V_{\max})e - \left(\sum_{k=0}^{\infty} \gamma^k (1+\gamma)(\epsilon + 2\delta V_{\max})\right) e \\ &\geq V^s_{\pi^A} - \frac{2(\epsilon + 2\delta V_{\max})}{1-\gamma} e. \end{aligned}$$

We now turn to bounding the loss in the sup norm with these tools. Let $V_1' = TV^s_{\pi^A}$. Then, we have $V_1'(x) = \mathrm{NashVal}(Q_{\pi^A,\pi^B}(x,\cdot,\cdot))$. Now, when $|\boldsymbol{Q}(x,\cdot,\cdot) - Q_{\pi^A,\pi^B}(x,\cdot,\cdot)|_\infty < \epsilon$ (i.e., the event $\mathfrak{E}$ occurs), we have $|\,\mathrm{NashVal}(\boldsymbol{Q}(x,\cdot,\cdot)) - \mathrm{NashVal}(Q_{\pi^A,\pi^B}(x,\cdot,\cdot))| < \epsilon$. Thus,

$$\begin{aligned} |V_1(x) - V_1'(x)| &= |E\,\mathrm{NashVal}(\boldsymbol{Q}(x,\cdot,\cdot)) - \mathrm{NashVal}(Q_{\pi^A,\pi^B}(x,\cdot,\cdot))| \\ &\leq E|\,\mathrm{NashVal}(\boldsymbol{Q}(x,\cdot,\cdot)) - \mathrm{NashVal}(Q_{\pi^A,\pi^B}(x,\cdot,\cdot))| \\ &= E\left[|\,\mathrm{NashVal}(\boldsymbol{Q}(x,\cdot,\cdot)) - \mathrm{NashVal}(Q_{\pi^A,\pi^B}(x,\cdot,\cdot))|\middle|\mathfrak{E}\right] P\left[\mathfrak{E}\right] \\ &\quad + E\left[|\,\mathrm{NashVal}(\boldsymbol{Q}(x,\cdot,\cdot)) - \mathrm{NashVal}(Q_{\pi^A,\pi^B}(x,\cdot,\cdot))|\middle|\mathfrak{E}^c\right] P\left[\mathfrak{E}^c\right] \\ &\leq \epsilon + 2\delta V_{\max}. \end{aligned}$$

But then, $V^*(x) \geq V^s_{\tilde{\pi}^A}(x) \geq V_1(x) - \frac{(1+\gamma)(\epsilon+2\delta V_{\max})}{1-\gamma} \geq V_1'(x) - \frac{2(\epsilon+2\delta V_{\max})}{1-\gamma}$. This holds for all $x \in \mathbb{X}$. Now, when combined with Proposition 8 and the fact that $V_1' = TV^s_{\pi^A}$, this gives

$$|V^* - V^s_{\tilde{\pi}^A}|_\infty \;\leq\; |V^* - V_1'|_\infty + \frac{2(\epsilon + 2\delta V_{\max})}{(1-\gamma)} \;\leq\; \gamma |V^* - V^s_{\pi^A}|_\infty + \frac{2(\epsilon + 2\delta V_{\max})}{(1-\gamma)}. \quad \blacksquare$$

IV. Policy Rollout for Markov Games

A. The Algorithm

Policy rollout is a recently developed technique used for policy improvement in MDPs [4]. The technique starts with a base policy, and uses sampling to determine the Q-function of that policy. It then uses this Q-function for one-step look-ahead to choose optimal actions. The policy resulting from this technique is shown in [4] to be no worse than the base policy for a wide class of MDPs.

In this section, we extend the policy rollout technique to zero-sum, discounted Markov games with bounded rewards. As we have two players, we use two base policies, one for each player. Using these two policies, and a model for the Markov game, we estimate the Q-function for the pair of policies, and then we use this Q-function for one-step look-ahead, solving the matrix game defined by the Q-function in the current state. Figure 1 displays the rollout algorithm for Markov games in detail. In this figure, the function $\mathrm{nextState}(x, a, b)$ returns a random state as specified by the transition law of the Markov game, when action a is used by A and action b is used by B in state x. The stochastic algorithm takes two policies $\langle \pi^A, \pi^B \rangle$, an integer N specifying the number of sample paths to be used, a finite horizon H, and the current state x as inputs, and outputs an action $a \in \mathbb{A}$ for player A. Thus, the algorithm generates a mixed policy $\tilde{\pi}^A$ for A. As we will prove shortly, under certain conditions, the policy $\tilde{\pi}^A$ is better than the policy π^A. All the results in this section are stated and proven, without loss of generality, for player A.

In the main result of this section, we bound the state-wise loss in performance due to rollout, and establish overall improvement in the sup norm due to rollout with appropriate choice of π^B, N, and H.

```
Function: $\mathrm{rollout}(\pi^A, \pi^B, N, H, x)$
input: policy $\pi^A$ for $A$, policy $\pi^B$ for $B$, number of samples $N$, horizon $H$, state $x$
output: action $a \in \mathbb{A}$
1. For each pair $\langle a, b\rangle$, $a \in \mathbb{A}, b \in \mathbb{B}$, and $i = 1 \ldots, N$, let
        $\boldsymbol{q}_i(a, b) = R(x, a, b) + \gamma \, \mathrm{estVal}(\pi^A, \pi^B, H, \mathrm{nextState}(x, a, b))$
2. Let $\boldsymbol{q}(a, b) = \frac{1}{N} \sum_{i=1} \boldsymbol{q}_i(a, b)$
3. Return a random action $\boldsymbol{a} \in \mathbb{A}$ according to distribution $\mathrm{Nash}_A(\boldsymbol{q}(\cdot, \cdot))$

Function: $\mathrm{estVal}(\pi^A, \pi^B, H, x)$
input: policy $\pi^A$ for $A$, policy $\pi^B$ for $B$, horizon $H$, state $x$
output: a sampled estimate of $V_{\pi^A,\pi^B}(x)$
1. If $H = 0$, return 0
2. Choose $a$ according to $\mu^{\pi^A}(x)$, and $b$ according to $\mu^{\pi^B}(x)$
3. Return $R(x, a, b) + \gamma \, \mathrm{estVal}(\pi^A, \pi^B, H{-}1, \mathrm{nextState}(x, a, b))$
```

Fig. 1. The rollout algorithm

B. *A Policy Improvement Result*

In this section we prove that, when called with appropriate parameters, the policy obtained using the algorithm in Figure 1 is an improvement over the base policy. Note that the rollout algorithm uses sampling to generate a stochastic estimate $\boldsymbol{q}(\cdot, \cdot)$ of $Q_{\pi^A,\pi^B}(x, \cdot, \cdot)$—denote this estimate $\boldsymbol{q}_x(\cdot, \cdot)$. By combining independent estimates $\boldsymbol{q}_x(\cdot, \cdot)$, for each state x, we get a random Q-function $\boldsymbol{Q}_{\mathrm{ro}}$. Let F_{ro} be the distribution for $\boldsymbol{Q}_{\mathrm{ro}}$. Then, the policy $\tilde{\pi}^A$ generated by the rollout algorithm is $\mathrm{lookaheadA}(F_{\mathrm{ro}})$. Our analysis in this section relies on examining the properties exhibited by $\boldsymbol{Q}_{\mathrm{ro}}$ and its distribution F_{ro}.

Theorem 1 implies that if F_{ro} is a sufficiently accurate approximation of $Q_{\pi^A,\pi^B}(\cdot, \cdot, \cdot)$, for $\pi^B \in \mathrm{brB}(\pi^A)$, then $\tilde{\pi}^A$ is no worse than π^A in the sup norm. This can be seen by choosing ϵ to be $(1-\gamma)^2 \big|V^* - V^s_{\pi^A}|_\infty/4$ and δ to be $(1-\gamma)^2 \big|V^* - V^s_{\pi^A}|_\infty/(8V_{\max})$ in Theorem 1, to get

$$|V^* - V^s_{\tilde{\pi}^A}|_\infty \leq \gamma |V^* - V^s_{\pi^A}|_\infty + \tfrac{2(\epsilon + 2\delta V_{\max})}{(1-\gamma)} = \left(\tfrac{\gamma+1}{2}\right) |V^* - V^s_{\pi^A}|_\infty \quad \leq \quad |V^* - V^s_{\pi^A}|_\infty.$$

This inequality is strict, giving a strict contraction, whenever π^A is not already a Nash policy, so that $|V^* - V^s_{\pi^A}|_\infty$ is non-zero[2].

We now turn to giving sufficient conditions on the sampling horizon H and the number of samples N to achieve the ϵ and δ values just given, so that policy improvement is guaranteed. Let π^A be a non-Nash policy for A (so that $\left|V^* - V^s_{\pi^A}\right|_\infty > 0$). Let π^B be a corresponding best-response policy. Let $\tilde{\pi}^A$ be the mixed policy resulting from using $\mathrm{rollout}(\pi^A, \pi^B, N, H, x)$, at every state x, for some integers N and H. Let ϵ and δ be chosen as above. Now, for any input state x, and for each $\boldsymbol{q}_i(\cdot, \cdot)$ defined in Step 1 of the rollout algorithm (see Figure 1), we have $\left|Q_{\pi^A,\pi^B}(x, \cdot, \cdot) - E\boldsymbol{q}_i(\cdot, \cdot)\right|_\infty \leq \gamma^{H+1} V_{\max} < \gamma^H V_{\max}$. Also, from Proposition 9, it follows that for the stochastic estimate $\boldsymbol{q}_x(\cdot, \cdot)$ defined above, we have $\left|\boldsymbol{q}_x - E\boldsymbol{q}_i\right|_\infty < \epsilon/2$, for any i, with probability at least $1 - e^{-\epsilon^2 N/32 V_{\max}^2}$.

We now choose H so that $\left|Q_{\pi^A,\pi^B}(x, \cdot, \cdot) - E\boldsymbol{q}_i(\cdot, \cdot)\right|_\infty < \gamma^H V_{\max} \leq \epsilon/2$, and N so that $\left|\boldsymbol{q}_x - E\boldsymbol{q}_i\right|_\infty < \epsilon/2$, with probability at least $1-\delta$, by ensuring that $e^{-\epsilon^2 N/32 V_{\max}^2} < \delta$. With $H > \log(\epsilon/2V_{\max})/(\log \gamma)$ and $N > -32 V_{\max}^2 (\log \delta)/\epsilon^2$, we then have $|Q_{\pi^A,\pi^B}(x, \cdot, \cdot) - \boldsymbol{q}_x(\cdot, \cdot)|_\infty < \epsilon$ with probability at least $1 - \delta$. This holds for every state $x \in \mathbb{X}$. Noting that the random Q-function $\boldsymbol{Q}_{\mathrm{ro}}$ is defined by $\boldsymbol{q}_x$ at each state x, independently, we then have that $\boldsymbol{Q}_{\mathrm{ro}}$ satisfies $|Q_{\pi^A,\pi^B}(x, \cdot, \cdot) - \boldsymbol{Q}_{\mathrm{ro}}(x, \cdot, \cdot)|_\infty < \epsilon$ with probability at least $1 - \delta$.

[2] In addition to this guarantee on the change in the sup-norm, Theorem 1 also provides a bound on the state-wise loss for any choice of ϵ and δ.

Theorem 1 then implies that $\tilde{\pi}^A$ (i.e., $\mathrm{lookaheadA}(F_{\mathrm{ro}})$) is better than π^A in the sup norm. Note that the values of the parameters N and H that guarantee this improvement are independent of the state-space size—it is important that the improvement guarantee is on the *mean* performance of a sampling-based mixed policy, and that any particular execution of this policy could be arbitrarily bad[3]. We have now proven the following theorem.

Theorem 2. Let π^A be any non-Nash policy for A, and π^B be a corresponding best-response policy. Then there exist N and H such that the mixed policy resulting from the rollout algorithm using these parameters is better than π^A in the sup norm. Moreover, the parameters N and H are independent of the size of the state space, $|\mathbb{X}|$.

V. Sparse Sampling for Markov Games

A. The Sparse Sampling Algorithm

Kearns et al. present a sparse-sampling technique for Markov games and prove that the technique computes a near-optimal policy using an amount of sampling independent of the state-space size. We note, though, that the amount of sampling required is exponential in the desired "accuracy", so that the policy rollout technique of the previous section is generally more practically useful.

Here, we show that the state-space independent near-optimality shown by Kearns et al. for this algorithm is also a direct consequence of our main theorem (Theorem 1), providing a distinct proof of their result. We start by presenting the algorithm carefully, for completeness.

The sparse-sampling algorithm for Markov games is straightforward, and is shown in Figure 2. Again the function $\mathrm{nextState}(x, a, b)$ is used to sample a next state when action a is used by A and action b is used by B in state x. Given the sampling width N (the number of samples at each level), sampling depth H, and the current state x, the algorithm builds a sampling tree to estimate $Q^*(x, \cdot, \cdot)$, the optimal Q-function in the current state, and then solves the resulting matrix game to generate a random action to be taken in state x. Let $\boldsymbol{Q}_{\mathrm{ss}}$ be a random Q-function constructed by combining such independent estimates of $Q^*(x, \cdot, \cdot)$ in all states x (such estimates are obtained by calling the function estQ^* in each state), and let F_{ss} denote its distribution. Then the policy generated by $\mathrm{selectAction}$ can be written as $\tilde{\pi}^A = \mathrm{lookaheadA}(F_{\mathrm{ss}})$. We will show that, with the proper choice of N and H, the stochastic Q-function $\boldsymbol{Q}_{\mathrm{ss}}$ approximates Q^* with arbitrary precision. Then near-optimality of $\tilde{\pi}^A$ (the policy generated by the algorithm) follows from Theorem 1.

B. Proof of Near-optimality

Now we will prove that algorithm presented in Figure 2 indeed computes a near-Nash policy. Our development is very similar to that of Kearns et al. for MDPs [8]. Also, we deviate from their line of argument by using Theorem 1, which we proved in Section III—we were unable to use the MDP techniques in [8] to prove our result for Markov games here.

Referring to Figure 2, define $\boldsymbol{Q}^h(x, \cdot, \cdot) = \mathrm{estQ}^*(h, N, x)$. Then, for all $h > 0$, $a \in \mathbb{A}$, and $b \in \mathbb{B}$, $\boldsymbol{Q}^0(x, a, b) = 0$, and $\boldsymbol{Q}^h(x, a, b) = R(x, a, b) + \frac{\gamma}{N} \sum_{x' \in \boldsymbol{S}_{a,b}(x)} \mathrm{NashVal}(\boldsymbol{Q}^{h-1}(x'))$.

Following Kearns et al. [8], given some $\lambda > 0$, define $\alpha_0 = V_{\max}$ and α_h recursively as $\alpha_{h+1} = \gamma(\lambda + \alpha_h)$. Then we can bound α_H with

$$\alpha_H = \left(\sum_{i=1}^{H} \gamma^i \lambda\right) + \gamma^H V_{\max} \le \frac{\lambda}{1-\gamma} + \gamma^H V_{\max}.$$

Analogous to Lemma 4 in Kearns et al. [8], we have the following result. We replicate and adapt their proof here, for completeness.

Lemma 1. With probability at least $1 - 4(|\mathbb{A}||\mathbb{B}|N + 1)^h e^{-\lambda^2 N/(8V_{\max}^2)}$ we have that

$$|Q^*(x, a, b) - \boldsymbol{Q}^h(x, a, b)| \le \alpha^h.$$

[3] A Chernoff-bound analysis can be used to give confidence intervals on the performance of single executions, of course.

```
Function: $\mathrm{selectAction}(N, H, x)$
input: sampling width $N$, sampling depth $H$, current state $x$
output: action $a_0$
1. Return a random action $\boldsymbol{a} \in \mathbb{A}$ according to $\mathrm{Nash}_A(\mathrm{estQ}^*(H, N, x))$

Function: $\mathrm{estQ}^*(H, N, x)$
input: depth $H$, width $N$, state $x$
output: estimated $Q$-function matrix $\hat{\boldsymbol{Q}}(x, \cdot, \cdot)$ for state $x$
1. If $H = 0$, return zero matrix
2. For each pair $\langle a, b\rangle$, $a \in \mathbb{A}, b \in \mathbb{B}$, let $\boldsymbol{S}_{a,b}(x)$ be a multiset of $N$ next-state samples drawn using $\mathrm{nextState}(x, a, b)$
3. For each pair $\langle a, b\rangle$, $a \in \mathbb{A}, b \in \mathbb{B}$, let
       $\hat{\boldsymbol{Q}}(x, a, b) = R(x, a, b) + \frac{\gamma}{N}\sum_{x' \in \boldsymbol{S}_{a,b}(x)} \mathrm{NashVal}(\mathrm{estQ}^*(H-1, N, x'))$
4. return $\hat{\boldsymbol{Q}}(x, \cdot, \cdot)$
```

Fig. 2. The sparse-sampling algorithm for Markov games

Proof. The argument is similar to that of Lemma 4 in [8], and is presented in [18]. ■

Recall that the stochastic Q-function $\boldsymbol{Q}_{\mathrm{ss}}$ was constructed by combining independent estimates of Q^* in each state. The estimate of Q^* generated by the sparse-sampling algorithm in state x is given by $\boldsymbol{Q}^H(x, \cdot, \cdot)$. Hence, from Lemma 1 it is clear that for any state x, we have $|Q^*(x, \cdot, \cdot) - \boldsymbol{Q}_{\mathrm{ss}}(x, \cdot, \cdot)|_\infty \leq \alpha^H$ with probability at least $1 - 4(|\mathbb{A}||\mathbb{B}|N + 1)^H e^{-\lambda^2 N/(8V_{\max}^2)}$. Recall that we denote the distribution of $\boldsymbol{Q}_{\mathrm{ss}}$ by F_{ss}.

But then, Theorem 1 implies that, given $\epsilon_0 > 0$, if F_{ss} is a sufficiently accurate approximation of Q^*, then the policy $\pi^A_{\mathrm{ss}} = \mathrm{lookaheadA}(F_{\mathrm{ss}})$ has a security level within ϵ_0 of the Nash value of the game. To see this, choose ϵ to be $(1-\gamma)\epsilon_0/4$, δ to be $(1-\gamma)\epsilon_0/(8V_{\max})$, and π^A to be some Nash policy for player A. Then, from Theorem 1, we have

$$V^* \quad \leq \quad V^s_{\pi^A_{\mathrm{ss}}}(x) + \frac{2(\epsilon + 2\delta V_{\max})}{(1-\gamma)} \leq V^s_{\pi^A_{\mathrm{ss}}}(x) + \epsilon_0.$$

With this background, we now move onto giving sufficient conditions on the sampling width N and the sampling horizon H to guarantee that F_{ss} as described above approximates Q^* with any given accuracy. Given $\epsilon_0 > 0$, let ϵ and δ be chosen as above. Choose λ and H such that $0 < \lambda < (1-\gamma)\epsilon/2$ and $\log(\epsilon/2V_{\max})/\log\gamma < H$. Then, for any state x, we have

$$\begin{aligned} |Q^*(x, \cdot, \cdot) - \boldsymbol{Q}_{\mathrm{ss}}(x, \cdot, \cdot)|_\infty \quad &\leq \quad \alpha^H \leq \frac{\lambda}{1-\gamma} + \gamma^H V_{\max} \\ &\leq \quad (\epsilon/2) + (\epsilon/2) \quad = \quad \epsilon \end{aligned}$$

with probability at least $1 - 4(|\mathbb{A}||\mathbb{B}|N+1)^H e^{-\lambda^2 N/(8V_{\max}^2)}$. Now, as the expression $4(|\mathbb{A}||\mathbb{B}|N + 1)^H e^{-\lambda^2 N/(8V_{\max}^2)}$ goes to zero as N goes to infinity, there exists finite N such that, for δ computed as above, $1 - 4(|\mathbb{A}||\mathbb{B}|N + 1)^H e^{-\lambda^2 N/(8V_{\max}^2)} \geq 1 - \delta$. Such a value of the sampling width N along with the horizon length of H described above ensures that the security level of the resulting policy is within ϵ_0 of the Nash value of the game. Note that the values of the parameters N and H that guarantee ϵ_0-optimality are independent of the size of the state space. Thus, we have proven the following Theorem.

Theorem 3. Given $\epsilon > 0$, there exist N and H such that the policy $\tilde{\pi}^A$ generated by $\mathrm{selectAction}(N, H, \gamma, G, \cdot)$ satisfies $|V^* - V^s_{\tilde{\pi}^A}|_\infty < \epsilon$. Moreover, the values of N and H do not depend on the size of the state space, $|\mathbb{X}|$.

VI. Conclusions

We presented a key approximation result for discounted Markov games with bounded rewards. This result establishes a bound on the state-wise loss that could be incurred from using approximate Q-functions for look-ahead. Our development of this result is more general than similar pre-existing results, and is applicable to state spaces with countable cardinality. Using this key approximation result, we discussed two sampling techniques for Markov games. The first technique—policy rollout—is our extension of the policy rollout technique for MDPs. We proved that under appropriate conditions, the policy generated by the extended policy rollout technique is closer to the Nash equilibrium than the base policy in the sup norm. We also put bound on the state-wise loss that could be incurred because of using approximate Q-function. The second technique is the sparse sampling technique presented by Kearns et al. [16]. We provided an alternate proof, using our new theorem, of Kearns' result that, when used with appropriate parameters, this technique produces a policy that is close to the Nash equilibrium with desired accuracy. For both of the techniques, the amount of sampling required to guarantee the results presented in this paper is independent of the state-space size.

References

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

[2] Dimitri P. Bertsekas and John N. Tsitsiklis, *Neuro-dynamic Programming*, Athena Scientific, Belmont, MA 02178, 1996.

[3] Peter Marbach, Oliver Mihatsch, and John N. Tsitsiklis, "Call admission control and routing in integrated services networks using neuro-dynamic programming," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 2, pp. 197–208, February 2000.

[4] Dimitri P. Bertsekas and David A. Castanon, "Rollout algorithms for stochastic scheduling problems," *Journal of Heuristics*, vol. 5, pp. 89–108, 1999.

[5] Uday Savagaonkar, Robert L. Givan, and Edwin K. P. Chong, "Dynamic pricing for bandwidth provisioning," in *Proceedings of the 36th Annual Conference on Information Sciences and Systems*, Princeton, New Jersey, March 2002, pp. 177–182.

[6] Edwin K. P. Chong, Robert L. Givan, and Hyeong-Soo Chang, "A framework for simulation-based network control via hindsight optimization," *Proceedings of the 39th IEEE Conference on Decision and Control*, vol. 2, pp. 1433–1438, 2000.

[7] Hyeong-Soo Chang, *On-line Sampling-based Control for Network Queuing*, Ph.D. thesis, Department of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907, July 2001.

[8] Michael J. Kearns, Yishay Mansour, and Andrew Y. Ng, "A sparse sampling algorithm for near-optimal planning in large markov decision processes," in *IJCAI*, 1999, pp. 1324–1231.

[9] Tamer Başar and Geert Jan Olsder, *Dynamic Noncooperative Game Theory*, Society for Industrial and Applied Mathematics, Philadelphia, 2nd edition, 1995.

[10] Stephen D. Patek and Dimitri P. Bertsekas, "Stochastic shortest path games," *SIAM Journal on Control and Optimization*, vol. 37, no. 3, pp. 804–824, 1999.

[11] Junling Hu and Michael P. Wellman, "Multiagent reinforcement learning: Theoretical framework and an algorithm," in *Proceedings of the 15th International Conference on Machine Learning*, 1998, pp. 242–250.

[12] Michael L. Littman, "Friend-or-foe Q-learning in general-sum games," in *Proceedings of the 18th International Conference on Machine Learning*, 2001, pp. 322–328.

[13] Michael L. Littman and Csaba Szepesvári, "A generalized reinforcement-learning model: Convergence and applications," in *Proceedings of the 13th International Conference on Machine Learning*, 1996, pp. 310–318.

[14] Michael Bowling and Manuela Veloso, "Multiagent learning using a variable learning rate," *Artificial Intelligence*, vol. 136, no. 2, pp. 215–250, 2002.

[15] Satinder Singh and Richard Yee, "An upper bound on the loss from approximate optimal-value functions," *Machine Learning*, vol. 16, pp. 227–233, 1994.

[16] Michael Kearns, Yishay Mansour, and Satinder Singh, "Fast planning in stochastic games," in *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence*, Morgan Kaufmann, 2000, pp. 309–316.

[17] Noga Alon, Joel H. Spencer, and Paul Erdős, *The Probabilistic Method*, Wiley-Interscience Series in Discrete Mathematics and Optimization, John Wieley and Sons, New York, 1992.

[18] Uday Savagaonkar, *Network Pricing using Sampling Techniques for Markov Games*, Ph.D. thesis, Department of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907, September 2002.