Geometric Variance Reduction in Markov Chains. Application to Value Function and Gradient Estimation

Rémi Munos
Centre de Mathématiques Appliquées,
Ecole Polytechnique, 91128 Palaiseau Cedex, France.
remi.munos@polytechnique.fr

Abstract

We study a sequential variance reduction technique for Monte Carlo estimation of functionals in Markov Chains. The method is based on designing *sequential control variates* using successive approximations of the function of interest V. Regular Monte Carlo estimates have a variance of $O(1/N)$, where N is the number of samples. Here, we obtain a geometric variance reduction $O(\rho^N)$ (with $\rho < 1$) up to a threshold that depends on the approximation error $V - \mathcal{A}V$, where $\mathcal{A}$ is an *approximation operator* linear in the values. Thus, if V belongs to the right approximation space (i.e. $\mathcal{A}V = V$), the variance decreases geometrically to zero.

An immediate application is value function estimation in Markov chains, which may be used for policy evaluation in policy iteration for Markov Decision Processes.

Another important domain, for which variance reduction is highly needed, is gradient estimation, that is computing the sensitivity $\partial_\alpha V$ of the performance measure V with respect to some parameter α of the transition probabilities. For example, in parametric optimization of the policy, an estimate of the policy gradient is required to perform a gradient optimization method.

We show that, using two approximations, the *value function* and the *gradient*, a geometric variance reduction is also achieved, up to a threshold that depends on the approximation errors of both of those representations.

Introduction

We consider a Markov chain over a finite state space $\mathcal{X}$ defined by the transition matrix P. We write $X(x)$ a trajectory $(x_t)_{t\geq 0}$ starting at a state $x_0 = x$. Let $\Psi(r, X(x))$ be a functional that depends on some function $r : \mathcal{X} \to I\!R$ and the trajectory $X(x)$, and write $V(x)$ the expectation of the functional that we wish to evaluate:

$$V(x) = \mathbb{E}[\Psi(r, X(x))]. \quad (1)$$

Here, the quantity of interest V is expressed in terms of a **Markov representation**, as an expectation of a functional that depends on trajectories. We will consider a functional $\Psi(r, \cdot)$ that is linear in r, and such that its expectation V may be equivalently be expressed in terms of a **solution to a linear system**

$$\mathcal{L}V = r, \quad (2)$$

with $\mathcal{L}$ an invertible linear operator (matrix).

An example of Ψ is the sum of discounted rewards r received along the trajectory:

$$\Psi(r, X(x)) = \sum_{t\geq 0} \gamma^t r(x_t). \quad (3)$$

with $\gamma < 1$ being a discount factor. In that case, V is solution to the Bellman equation (2) with $\mathcal{L} = I - \gamma P$. Indeed, using matrix notations, V equals $\sum_{t\geq 0} \gamma^t P^t r = (I - \gamma P)^{-1} r$.

Other functionals include finite-horizon sum of rewards $\Psi(r, X(x)) = \sum_{t=0}^{T} r(x_t)$ or infinite-time average reward $\Psi(r, X(x)) = \lim_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} r(x_t)$.

A regular Monte-Carlo (MC) method would estimate $V(x)$ by sampling N independent trajectories $\{X^n(x)\}_{1\leq n\leq N}$ starting from x and calculate the average $\frac{1}{N}\sum_{n=1}^{N} \Psi(r, X^n(x))$. The variance of such an estimator is of order $1/N$. Variance reduction is crucial since the numerical approximation error of the quantity of interest is directly related to the variance of its estimate.

Variance reduction techniques include importance sampling, correlated sampling, control variates, antithetic variates and stratified sampling, see e.g. (Hammersley and Handscomb, 1964; Halton, 1970). Geometric variance reduction rates have been obtained by processing these variance reduction methods iteratively, the so-called *sequential* (or *recursive*) *Monte-Carlo*. Examples include adaptive importance sampling (Kollman et al., 1999) and what Halton called the "Third Sequential Method" (Halton, 1994) based on sequential correlated sampling and control variates. This approach has been recently developed in (Maire, 2003) for numerical integration and, more related to our work, applied to (continuous time) Markov processes in (Gobet and Maire, 2005).

The idea is to replace the expectation of $\Psi(r, \cdot)$ by an expectation of $\Psi(r - \mathcal{L}W, \cdot)$ for some function W close to V. From the linearity of Ψ and the equivalence between the representations (1) and (2), for any W, one has

$$V(x) = W(x) + \mathbb{E}[\Psi(r - \mathcal{L}W, X(x))].$$

Thus, if W is a good approximation of V, the residual $r - \mathcal{L}W$ is small, and the variance is low.

In the sequential method described in this paper, we use successive approximations V_n of V to estimate by Monte

Carlo a correction E_n using the residual $r-\mathcal{L}V_n$ in Ψ, which is used to process a new approximation V_{n+1}. We consider an approximation operator $\mathcal{A}$ that is *linear in the values*. We show that (for enough sampled trajectories at each iteration), the variance of the estimator has a geometric rate ρ^N (with $\rho < 1$, and N the total number of sampled trajectories) until some threshold is reached, whose value is related to the approximation error $\mathcal{A}V - V$.

An interesting extension of this method concerns the estimation of the gradient $\partial_\alpha V$ of V with respect to (w.r.t.) some parameter α of the transition matrix P. A useful application of such sensitivity analysis appears in policy gradient estimation. An optimal control problem may be approximated by a parametric optimization problem in a given space of parameterized policies. Thus, the transition matrix P depends on some (possible multidimensional) policy parameter α. In order to apply gradient methods to search for a local maximum of the performance in the parameter space, one wishes to estimate the policy gradient, i.e. the sensitivity $Z = \partial_\alpha V$ of the performance measure with respect to α. The gradient may be expressed as an expectation $Z(x) = \mathbb{E}[\Phi(r, X(x))]$, using the so-called *likelihood ratio* or *score method* (Reiman and Weiss, 1986; Glynn, 1987; Williams, 1992; Baxter and Bartlett, 2001; Marbach and Tsitsiklis, 2003). The gradient Z is also the solution to a linear system

$$\mathcal{L}Z = -\partial_\alpha \mathcal{L}\, \mathcal{L}^{-1} r = -\partial_\alpha \mathcal{L} V. \tag{4}$$

Indeed, since V solves $V = \mathcal{L}^{-1} r$, we have $Z = \partial_\alpha V = -\mathcal{L}^{-1}\partial_\alpha \mathcal{L}\, \mathcal{L}^{-1} r$. For example, in the discounted case (3), the functional Φ is defined by

$$\Phi(r, X(x)) = \sum_{t\geq 0} \gamma^t r(x_t) \sum_{s=0}^{t-1} \frac{\partial_\alpha P(x_s, x_{s+1})}{P(x_s, x_{s+1})}, \tag{5}$$

and Z is solution to (4) with $\mathcal{L} = I - \gamma P$ and $\partial_\alpha \mathcal{L} = -\gamma \partial_\alpha P$.

We show that, using two approximations V_n and Z_n of the *value function* and the *gradient*, a geometric variance reduction is also achieved, up to a threshold that depends on the approximation errors of both of those representations.

Numerical experiments on a simple Gambler's ruin problem illustrate the approach.

Value function estimation

We first describe the approximation operator *linear in the values* considered here, then describe the algorithm, and state the main result on geometric variance reduction.

Approximation operator $\mathcal{A}$

We consider a fixed set of J representative states $\mathcal{X}_J := \{x_j \in \mathcal{X}\}_{1\leq j\leq J}$ and functions $\{\phi_j : \mathcal{X} \to I\!R\}_{1\leq j\leq J}$. The linear approximation operator $\mathcal{A}$ maps any function $W : \mathcal{X}_J \to I\!R$ to the function $\mathcal{A}W : \mathcal{X} \to I\!R$, according to

$$\mathcal{A}W(x) = \sum_{j=1}^{J} W(x_j)\phi_j(x). \tag{6}$$

This kind of function approximation includes:

- Linear regression, for example with *Spline*, *Polynomial*, *Radial Basis*, *Fourier* or *Wavelet* decomposition. This is the projection of a function W onto the space spanned by a set of functions $\{\psi_k : \mathcal{X} \to I\!R\}_{1\leq k\leq K}$, that is which minimizes some norm (induced by a discrete inner product $\langle f, g\rangle := \sum_{j=1}^{J} \mu_j f(x_j) g(x_j)$, for some distribution μ over $\mathcal{X}_J$):
$$\min_{\alpha\in I\!R^K} \Big|\Big| \sum_{k=1}^{K} \alpha_k \psi_k - W \Big|\Big|^2.$$
The solution α solves the linear system $A\alpha = b$ with A an $K \times K$-matrix of elements $A_{kl} = \langle \psi_k, \psi_l\rangle$ and b a K-vector of components $b_k = \langle W, \psi_k\rangle$. Thus $\alpha_k = \sum_{l=1}^{K} A^{-1}_{kl} \sum_{j=1}^{J} \mu_j \psi_l(x_j) W(x_j)$ and the best fit $\sum_{k=1}^{K} \alpha_k \psi_k$ is thus of type (6) with
$$\phi_j(x) = \mu_j \sum_{k=1}^{K}\sum_{l=1}^{K} A^{-1}_{kl} \psi_l(x_j)\psi_k(x). \tag{7}$$
- k-nearest neighbors (Hastie et al., 2001): here $\phi_j(x) = \frac{1}{k}$ if x has x_j as one of its $k-$nearest neighbors, and $\phi_j(x) = 0$ otherwise.
- Locally weighted learning and Kernel regression (Atkeson et al., 1997). Functions similar to (7) may be derived, with the matrix A being dependent on x (through the kernel).

The algorithm

We assume the equivalence between the Markov representation (1) and its interpretation in terms of the solution to the linear system (2), i.e. for any function $f : \mathcal{X} \to I\!R$, we have

$$f(x) = \mathbb{E}[\Psi(\mathcal{L}f, X(x))]. \tag{8}$$

We consider successive approximations $V_n \in I\!R^J$ of V defined at the states $\mathcal{X}_J = (x_j)_{1\leq j\leq J}$ recursively:

- We initialize $V_0(x_j) = 0$.
- At stage n, we use the values $V_n(x_j)$ to provide a new estimation of $V(x_j)$. Let $E_n(x_j) := V(x_j) - \mathcal{A}V_n(x_j)$ be the approximation error at the states $\mathcal{X}_J$. From the equivalence property (8), we have: $\mathcal{A}V_n(x) = \mathbb{E}[\Psi(\mathcal{L}\mathcal{A}V_n, X(x))]$. Thus, by linearity of Ψ w.r.t. its first variable,
$$E_n(x_j) = \mathbb{E}[\Psi(r - \mathcal{L}\mathcal{A}V_n, X(x_j))].$$
Now, we use a Monte Carlo technique to estimate $E_n(x_j)$ with M trajectories $(X^{n,m}(x_j))_{1\leq m\leq M}$: we calculate the average
$$\widehat{E_n}(x_j) := \frac{1}{M}\sum_{m=1}^{M} \Psi(r - \mathcal{L}\mathcal{A}V_n, X^{n,m}(x_j))$$
and define the new approximation at the states $\mathcal{X}_J$:
$$V_{n+1}(x_j) := \mathcal{A}V_n(x_j) + \widehat{E_n}(x_j). \tag{9}$$

Remark 1. *Notice that there is a slight difference between this algorithm and that of (Gobet and Maire, 2005), which may be written $V_{n+1}(x_j) = V_n(x_j) + \mathcal{A}[\frac{1}{M}\sum_{m=1}^{M} \Psi(r - \mathcal{L}V_n, X^{n,m}(x_j))]$. Our formulation enables us to avoid the assumption of the idempotent property for $\mathcal{A}$ (i.e. that $\mathcal{A}^2 = \mathcal{A}$), and guarantees that V_n is an unbiased estimate of V, for all n, as showed in the next paragraph.*

Properties of the estimates V_n

We write the conditional expectations and variances:

$$\mathbb{E}^n[Y] = \mathbb{E}[Y|X^{p,m}(x_j), 0 \leq p < n, 1 \leq m \leq M, 1 \leq j \leq J]$$

and $\text{Var}^n[Y] = \mathbb{E}^n[Y^2] - (\mathbb{E}^n[Y])^2$. We have the following properties on the estimates:

Expectation of V_n. From the definition (9),

$$\mathbb{E}^n[V_{n+1}(x_j)] = \mathcal{A}V_n(x_j) + E_n(x_j) = V(x_j).$$

Thus $\mathbb{E}[V_n(x_j)] = V(x_j)$ for all $n \geq 1$: the approximation $V_n(x_j)$ is an unbiased estimate of $V(x_j)$.

Variance of V_n. Write $v_n = \sup_{1 \leq j \leq J} \text{Var}\, V_n(x_j)$. The following result expresses that for a large enough value of M, the variance decreases geometrically.

Theorem 1. *We have*

$$v_{n+1} \leq \rho_M v_n + \frac{2}{M}\mathcal{V}_\Psi(V - \mathcal{A}V) \tag{10}$$

with $\rho_M = \frac{2}{M}\big(\sum_{j=1}^J \sqrt{\mathcal{V}_\Psi(\phi_j)}\big)^2$, *using the notation*

$$\mathcal{V}_\Psi(f) := \sup_{1 \leq j \leq J} \textit{Var}\, \Psi(\mathcal{L}f, X(x_j)).$$

Thus, for a large enough value of M, (i.e. whenever $\rho_M < 1$), $(v_n)_n$ decreases geometrically at rate ρ_M, up to the threshold

$$\limsup_{n \to \infty} v_n \leq \frac{1}{1 - \rho_M}\frac{2}{M}\mathcal{V}_\Psi(V - \mathcal{A}V).$$

If V belongs to the space of functions that are representable by $\mathcal{A}$, i.e. $\mathcal{A}V = V$, then the variance geometrically decreases to 0 at rate ρ^N with $\rho := \rho_M^{1/M}$ and N the total number of sampled trajectories per state x_j (i.e. N is the number n of iterations times the number M of trajectories per iteration and state x_j). Further research should consider the best tradeoff between n and M, for a given budget of a total of $N = nM$ trajectories per state.

Notice that the threshold depends on the variance of Ψ for the function $\mathcal{L}(V - \mathcal{A}V) = r - \mathcal{L}\mathcal{A}V$, the residual of the representation (by $\mathcal{A}$) of V. Notice also that this threshold depends on $V - \mathcal{A}V$ only at states reached by the trajectories $\{X(x_j)\}_{x_j \in \mathcal{X}_J}$: a uniform (over the whole domain) representation of V is not required.

Of course, once the threshold is reached, a further convergence of $O(1/N)$ can be obtained thereafter, using regular Monte Carlo.

Example: the discounted case

Let us illustrate the sequential control variates algorithm to value function estimation in Markov chains. As an example, we consider the infinite horizon, discounted case (3). The value function $V(x) = \mathbb{E}\big[\sum_{t \geq 0}\gamma^t r(x_t)\big]$ solves the Bellman equation: $V = r + \gamma PV$, which may be written as the linear system (2) with $\mathcal{L} = I - \gamma P$.

In the previous algorithm, at stage n, the approximation error $E_n(x_j) = V(x_j) - \mathcal{A}V_n(x_j)$ is therefore the expectation

$$E_n(x_j) = \mathbb{E}\big[\sum_{t \geq 0}\gamma^t [r(x_t) - (I - \gamma P)\mathcal{A}V_n(x_t)]|x_0 = x_j\big]. \tag{11}$$

We notice that the term $r - (I - \gamma P)\mathcal{A}V_n$ is the *Bellman residual* of the approximation $\mathcal{A}V_n$. The estimate has thus zero variance if this approximation happens to be the value function.

In model-free learning, it may be interesting (in order to avoid the computation of the expectation in P) to replace the term $P\mathcal{A}V_n(x_t)$ by $\mathcal{A}V_n(x_{t+1})$ in (11), leaving the expectation unchanged (because of the linearity of the approximation operator $\mathcal{A}$):

$$E_n(x_j) = \mathbb{E}\Big[\sum_{t \geq 0}\gamma^t [r(x_t) - \mathcal{A}V_n(x_t) + \gamma\mathcal{A}V_n(x_{t+1})]|x_0 = x_j\Big], \tag{12}$$

at the cost of a slight increase of the variance (here, even if the approximation $\mathcal{A}V_n$ is the value function, the variance is not zero). The approximation error is thus the expected discounted sum of Temporal Differences (Sutton, 1988) $r(x_t) - \mathcal{A}V_n(x_t) + \gamma\mathcal{A}V_n(x_{t+1})$.

Following the algorithm, the next approximation V_{n+1} is defined by (9) with $\widehat{E_n(x_j)}$ being a Monte Carlo estimate of (11) or (12).

Numerical experiment

We consider the *Gambler's ruin problem* described in (Kollman et al., 1999): a gambler with i dollars bets repeatedly against the house, whose initial capital is $L - i$. Each bet is one dollar and the gambler has probability p of winning. The state space is $\mathcal{X} = \{0, \ldots, L\}$ and the transition matrix P is defined, for $i, j \in \mathcal{X}$, by

$$P_{ij} = \begin{cases} p, & \text{if } j - i = 1 \text{ and } 0 < i < L, \\ 1 - p, & \text{if } i - j = 1 \text{ and } 0 < i < L, \\ 0, & \text{otherwise.} \end{cases}$$

Betting continues until either the gambler is ruined ($i = 0$) or he has "broken the bank" ($i = L$) (thus 0 and L are terminal states). We are interested in computing the probability of the gambler's eventual ruin $V(i)$ when starting from initial fortune i. We thus define the function $r(0) = 1$ and $r(i \neq 0) = 0$. The value function V solves the Bellman equation $(I - P)V = r$, and its value is

$$V(i) = \frac{\lambda^i - \lambda^L}{1 - \lambda^L}, \text{ for } i \in \mathcal{X}, \tag{13}$$

with $\lambda := \frac{1-p}{p}$ when $p \neq 0.5$, and $V(i) = 1 - i/L$ for $p = 0.5$. The representative states are $X_J = \{1, 7, 13, 19\}$. We consider two linear function approximation $\mathcal{A}_1$ and $\mathcal{A}_2$ that are projection operators (minimizing the L_2 norm at the states X_J) onto the space spanned by a set of functions $\{\psi_k : \mathcal{X} \to I\!R\}_{1 \leq k \leq K}$. $\mathcal{A}_1$ uses $K = 2$ functions $\psi_1(i) = 1, \psi_2(i) = \lambda^i, i \in \mathcal{X}$, whereas $\mathcal{A}_2$ uses $K = 4$ functions $\psi_1(i) = 1, \psi_2(i) = i, \psi_3(i) = i^2, \psi_4(i) = i^3$, $i \in \mathcal{X}$. Notice that V is representable by $\mathcal{A}_1$ (i.e. $\mathcal{A}_1 V = V$) but not by $\mathcal{A}_2$. We chose $p = 0.51$.

We ran the algorithm with $\mathcal{L} = I - P$ (which is an invertible matrix). At each iteration, we used $M = 100$ simulations per state. Figure 1 shows the L_∞ approximation error ($\max_{j \in X_J} |V(j) - V_n(j)|$) in logarithmic scale, as a

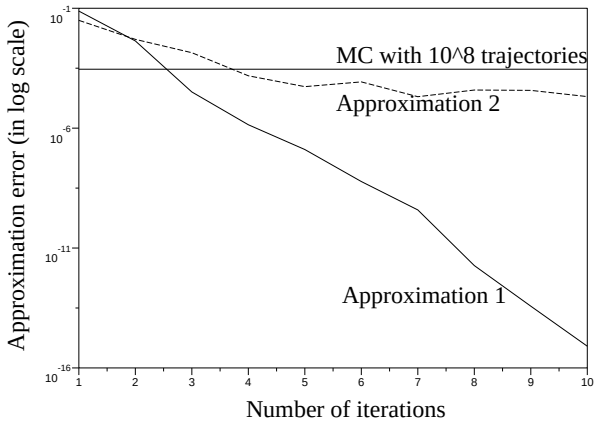

Figure 1: Approximation error for regular MC and sequential control variate algorithm using two approximations $\mathcal{A}_1$ and $\mathcal{A}_2$, as a function of the number of iterations.

function of the iteration number $1 \leq n \leq 10$. This approximation error (which is the true quantity of interest) is directly related to the variance of the estimates V_n.

For the approximation $\mathcal{A}_1$, we observe the geometric convergence to 0, as predicted in Theorem 1. It takes less than 10×100 simulations per state to reach an error of 10^{-15}. Using $\mathcal{A}_2$, the error does not decrease below some threshold $\simeq 2.10^{-5}$ due to the approximation error $V - \mathcal{A}_2 V$. This threshold is reached using about 5×100 simulations per state. For comparison, usual MC reaches an error of 10^{-4} with 10^8 simulations per state.

The variance reduction obtained when using such sequential control variates is thus considerable.

Gradient estimation

Here, we assume that the transition matrix P depends on some parameter α, and that we wish to estimate the sensitivity of $V(x) = \mathbb{E}[\Psi(r, X(x))]$ with respect to α, which we write $Z(x) = \partial_\alpha V(x)$.

An example of interest consists in solving approximately a Markov Decision Problem by searching for a feedback control law in a given class of parameterized stochastic policies. The optimal control problem is replaced by a parametric optimization problem, which may be solved (at least in order to find a local optimum) using gradient methods. Thus we are interested in estimating the gradient of the performance measure w.r.t. the parameter of the policy. In this example, the transition matrix P would be the transition matrix of the MDP combined with the parameterized policy.

As mentioned in the introduction, the gradient may be expressed by an expectation $Z(x) = \mathbb{E}[\Phi(r, X(x))]$ (using the so-called *likelihood ratio* or *score method* (Reiman and Weiss, 1986; Glynn, 1987; Williams, 1992; Baxter and Bartlett, 2001; Marbach and Tsitsiklis, 2003)) where $\Phi(r, X(x))$ is also a functional that depends on the trajectory $X(x)$, and that is linear in its first variable. For example, in the discounted case (3), the functional Φ is given by (5). The variance is usually high, thus variance reduction techniques are highly needed (Greensmith et al., 2005).

The gradient Z is also the solution to the linear system (4). Unfortunately, this linear expression is not of the form (2) since $\partial_\alpha \mathcal{L}$ is not invertible, which prevents us from using directly the method of the previous section.

However, the linear equation (4) provides us with another representation for Z in terms of Markov chains:

$$Z(x) = -\mathbb{E}\big[\textstyle\sum_{t\geq 0} \gamma^t \partial_\alpha \mathcal{L} V(x_t)\big] = -\mathbb{E}[\Psi(\partial_\alpha \mathcal{L} V, X(x))]. \tag{14}$$

We may extend the previous algorithm to the estimation of Z by using two representations: V_n and Z_n. The approximation V_n of V is updated from Monte-Carlo estimation of the residual $r - \mathcal{L}V_n$, and Z_n, which approximates Z, is updated from the gradient residual $\partial_\alpha \mathcal{L} V_n - \mathcal{L} Z_n$ built from the current V_n. This approach may be related to the so-called *Actor-Critic algorithms* (Konda and Borkar, 1999; Sutton et al., 2000), which use the representation (14) with an approximation of the value function.

A geometric variance reduction is also achieved, up to a threshold that depends on the approximation errors of both of those representations.

The algorithm

From (14) and the equivalence property (8), we obtain the following representation for Z:

$$\begin{aligned} Z(x) &= \mathcal{A} Z_n(x) + \mathbb{E}\big[\Psi(-\partial_\alpha \mathcal{L} V - \mathcal{L}\mathcal{A} Z_n, X(x))\big] \\ &= \mathcal{A} Z_n(x) + \mathbb{E}\big[\Psi(-\partial_\alpha \mathcal{L}(V - \mathcal{A} V_n), X(x)) \\ &\quad -\Psi(\partial_\alpha \mathcal{L}\mathcal{A} V_n + \mathcal{L}\mathcal{A} Z_n, X(x))\big] \\ &= \mathcal{A} Z_n(x) + \mathbb{E}\big[\Phi(r - \mathcal{L}\mathcal{A} V_n, X(x)) \\ &\quad -\Psi(\partial_\alpha \mathcal{L}\mathcal{A} V_n + \mathcal{L}\mathcal{A} Z_n, X(x))\big]. \end{aligned} \tag{15}$$

from which the algorithm is deduced. We consider successive approximations $V_n \in \mathbb{R}^J$ of V and $Z_n \in \mathbb{R}^J$ of Z defined at the states $\mathcal{X}_J = (x_j)_{1\leq j\leq J}$.

- We initialize $V_0(x_j) = 0$, $Z_0(x_j) = 0$.
- At stage n, we simulate by Monte Carlo M trajectories $(X^{n,m}(x_j))_{1\leq m\leq M}$ and define the new approximations V_{n+1} and Z_{n+1} at the states $\mathcal{X}_J$:

$$V_{n+1}(x_j) = \mathcal{A} V_n(x_j) + \frac{1}{M}\sum_{m=1}^{M} \Psi(r - \mathcal{L}\mathcal{A} V_n, X^{n,m}(x_j))$$

$$\begin{aligned} Z_{n+1}(x_j) = \mathcal{A} Z_n(x_j) + \frac{1}{M}\sum_{m=1}^{M} \big[&\Phi(r - \mathcal{L}\mathcal{A} V_n, X^{n,m}(x_j)) \\ &-\Psi(\partial_\alpha \mathcal{L}\mathcal{A} V_n + \mathcal{L}\mathcal{A} Z_n, X^{n,m}(x_j))\big]. \end{aligned}$$

Expectation of V_n and Z_n. We have already seen that $\mathbb{E}[V_n] = V$ for all $n > 0$. Now, (15) implies that $\mathbb{E}^n[Z_{n+1}] = Z$, thus $\mathbb{E}[Z_n] = Z$ for all $n > 0$.

Variance of V_n and Z_n. We write $v_n = \sup_{1\leq j\leq J} \mathrm{Var}\, V_n(x_j)$ and $z_n = \sup_{1\leq j\leq J} \mathrm{Var}\, Z_n(x_j)$. The next theorem state the geometric variance reduction for large enough values of M.

Theorem 2. *We have*

$$\begin{aligned} v_{n+1} &\leq \rho_M v_n + \frac{2}{M}\mathcal{V}_\Psi(V - \mathcal{A}V) \\ z_{n+1} &\leq \rho_M z_n + \frac{2}{M}[c_1(V - \mathcal{A}V, Z - \mathcal{A}Z) + c_2 v_n] \end{aligned}$$

with $\rho_M = \frac{2}{M}\big(\sum_{j=1}^J \sqrt{\mathcal{V}_\Psi(\phi_j)}\big)^2$*, and the coefficients*

$$\begin{aligned} c_1(f,g) &= \Big(\sqrt{\mathcal{V}_\Phi(f)} + \sqrt{\mathcal{V}_\Psi(\mathcal{L}^{-1}\partial_\alpha \mathcal{L} f)} + \sqrt{\mathcal{V}_\Psi(g)}\Big)^2 \\ c_2 &= \Big[\sum_{j=1}^J \sqrt{\mathcal{V}_\Phi(\phi_j)} + \sqrt{\mathcal{V}_\Psi(\mathcal{L}^{-1}\partial_\alpha \mathcal{L}\phi_j)}\Big]^2, \end{aligned}$$

using the notations $\mathcal{V}_\Psi(f) := \sup_{1\leq j\leq J}$ *Var* $\Psi(\mathcal{L}f, X(x_j))$ *and* $\mathcal{V}_\Phi(f) := \sup_{1\leq j\leq J}$ *Var* $\Phi(\mathcal{L}f, X(x_j))$*. Thus, for a large enough value of* M*, (i.e. whenever* $\rho_M < 1$*), the convergence of* $(v_n)_n$ *and* $(z_n)_n$ *is geometric at rate* ρ_M*, up to the thresholds*

$$\begin{aligned} \limsup_{n\to\infty} v_n &\leq \frac{1}{1-\rho_M}\frac{2}{M}\mathcal{V}_\Psi(V - \mathcal{A}V) \\ \limsup_{n\to\infty} z_n &\leq \frac{1}{1-\rho_M}\frac{2}{M}\Big[c_1(V - \mathcal{A}V, Z - \mathcal{A}Z) \\ &\quad + c_2\frac{1}{1-\rho_V}\frac{2}{M}\mathcal{V}_\Psi(V - \mathcal{A}V)\Big]. \end{aligned}$$

Here also, if V and Z are representable by $\mathcal{A}$, then the variance converges geometrically to 0.

Numerical experiment

Again we consider the *Gambler's ruin problem* described previously. The transition matrix is parameterized by $\alpha = p$, the probability of winning. The gradient $Z(i) = \partial_\alpha V(i)$ may be derived from (13):

$$Z(i) = \frac{L(1-\lambda^i)\lambda^{L-1} - i(1-\lambda^L)\lambda^{i-1}}{(1-\lambda^L)^2\alpha^2} \text{ for } i \in \mathcal{X},$$

(for $\alpha \neq 0.5$), and $Z(i) = 0$ for $\alpha = 0.5$. Again we use the representative states $X_J = \{1, 7, 13, 19\}$. Here, we consider two approximators $\mathcal{A}_1$ and $\mathcal{A}_2$ for the value function representations V_n, and two approximators $\mathcal{A}_2$ and $\mathcal{A}_3$ for the gradient representations Z_n, where $\mathcal{A}_3$ is a projection that uses $K = 3$ functions $\psi_1(i) = 1, \psi_2(i) = \lambda^i, \psi_3(i) = i\lambda^i, i \in \mathcal{X}$. Notice that Z is representable by $\mathcal{A}_3$ but not by $\mathcal{A}_2$. We chose $p = 0.51$ and $M = 1000$.

Figure 2 shows the L_∞ approximation error of Z ($\max_{j\in X_J} |Z(j) - Z_n(j)|$) in logarithmic scale, for the different possible approximations of V and Z.

When both V and Z may be perfectly approximated (i.e. $\mathcal{A}_1$ for V and $\mathcal{A}_3$ for Z) we observe the geometric convergence to 0, as predicted in Theorem 2. The error is about 10^{-14} using a total of 10^4 simulations. When either the value function or the gradient is not representable in the approximation spaces, the error does not decrease below some threshold ($\simeq 3.10^{-3}$ when Z is not representable) reached in 2.10^3 simulations. The threshold is lower ($\simeq 2.10^{-4}$) when Z is representable. For comparison, usual MC reaches an error (for Z) of 3.10^{-3} with 10^8 simulations per state.

The variance reduction of this sequential method compared to regular MC is thus also considerable.

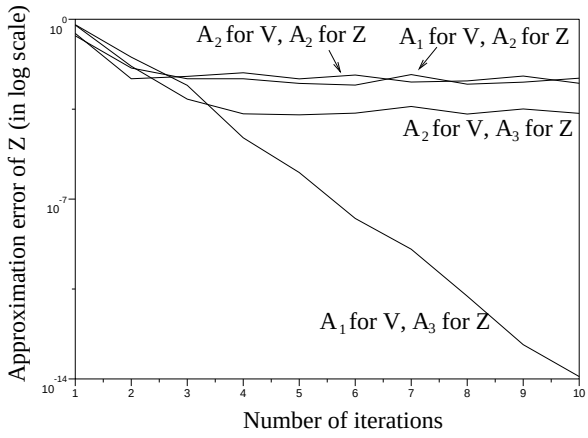

Figure 2: Approximation error of the gradient $Z = \partial_\alpha V$ using approximators $\mathcal{A}_1$ and $\mathcal{A}_2$ for the value function, and $\mathcal{A}_2$ and $\mathcal{A}_3$ for the gradient.

Conclusion

We described a sequential control variates method for estimating an expectation of functionals in Markov chains, using linear approximation (in the values). We illustrate the method on value function and policy estimates. We proved geometric variance reduction up to a threshold that depends on the approximation error of the functions of interest.

Future work would consider sampling initial states from some distribution over $\mathcal{X}$ instead of using representative states $\mathcal{X}_J$.

Proof of Theorem 1

From the decomposition

$$V - \mathcal{A}V_n = V - \mathcal{A}V + \textstyle\sum_{i=1}^J (V - V_n)(x_i)\phi_i, \qquad (16)$$

we have

$$\begin{aligned} V_{n+1}(x_j) = \mathcal{A}V_n(x_j) + \frac{1}{M}\sum_{m=1}^M \Big[&\Psi(\mathcal{L}(V - \mathcal{A}V), X^{n,m}(x_j)) \\ &+ \textstyle\sum_{i=1}^J (V - V_n)(x_i)\Psi(\mathcal{L}\phi_i, X^{n,m}(x_j))\Big]. \end{aligned}$$

Thus

$$\begin{aligned} \mathrm{Var}^n V_{n+1}(x_j) \;=\; \frac{1}{M}\mathrm{Var}^n\big[&\Psi(\mathcal{L}(V - \mathcal{A}V), X(x_j)) \\ &+ \textstyle\sum_{i=1}^J (V - V_n)(x_i)\Psi(\mathcal{L}\phi_i, X(x_j))\big]. \end{aligned}$$

We use the general bound

$$\mathrm{Var}\left[\textstyle\sum_i \alpha_i Y_i\right] \leq \big[\sum_i |\alpha_i|\sqrt{\mathrm{Var}\,[Y_i]}\big]^2, \qquad (17)$$

for any real numbers $(\alpha_i)_i$ and square integrable real random variables $(Y_i)_i$, to deduce that

$$\begin{aligned} \mathrm{Var}^n V_{n+1}(x_j) \;\leq\; \frac{1}{M}\Big[&\sqrt{\mathcal{V}_\Psi(V - \mathcal{A}V)} \\ &+ \textstyle\sum_{i=1}^J |V - V_n|(x_i)\sqrt{\mathcal{V}_\Psi(\phi_i)}\Big]^2, \end{aligned} \qquad (18)$$

with $\mathcal{V}_\Psi(f) := \sup_{1\le j\le J} \text{Var}\, \Psi(\mathcal{L}f, X(x_j))$. Now, we use the variance decomposition

$$\begin{aligned}\text{Var}\, V_{n+1}(x_j) &= \text{Var}\,[\mathbb{E}^n[V_n(x_j)]] + \mathbb{E}[\text{Var}^n[V_n(x_j)]] \\ &= \mathbb{E}[\text{Var}^n[V_n(x_j)]],\end{aligned}$$

and the general bound

$$\mathbb{E}\big[(\alpha_0+\sum_{i=1}^J \alpha_i Y_i)^2\big] \le 2\alpha_0^2 + 2\Big(\sum_{i=1}^J |\alpha_i|\sqrt{\mathbb{E}[Y_i^2]}\Big)^2, \quad (19)$$

to deduce from (18) that

$$v_{n+1} \le \frac{2}{M}\big[\mathcal{V}_\Psi(V-\mathcal{A}V) + \big(\sum_{i=1}^J \sqrt{\mathcal{V}_\Psi(\phi_i)}\big)^2 v_n\big],$$

which gives (10). Now, if M is such that $\rho_M := \frac{2}{M}\big(\sum_{i=1}^J \sqrt{\mathcal{V}_\Psi(\phi_i)}\big)^2 < 1$, then taking the upper limit finishes the proof of Theorem 1.

Proof of Theorem 2

Using (4) and (6), we have the decomposition

$$\begin{aligned}&-\partial_\alpha \mathcal{L}\mathcal{A}V_n - \mathcal{L}\mathcal{A}Z_n \\ &= -\partial_\alpha \mathcal{L}\mathcal{A}(V_n - V) - \partial_\alpha \mathcal{L}(\mathcal{A}V - V) \\ &\quad +\mathcal{L}(Z-\mathcal{A}Z) + \mathcal{L}\mathcal{A}(Z-Z_n) \\ &= \textstyle\sum_{i=1}^J (V-V_n)(x_i)\partial_\alpha \mathcal{L}\phi_i - \partial_\alpha \mathcal{L}(\mathcal{A}V - V) \\ &\quad +\mathcal{L}(Z-\mathcal{A}Z) + \textstyle\sum_{i=1}^J (Z-Z_n)(x_i)\mathcal{L}\phi_i.\end{aligned}$$

Now, using (16), $\text{Var}^n Z_{n+1}(x_j)$ equals

$$\begin{aligned}&\frac{1}{M}\text{Var}^n\Big[\Phi(\mathcal{L}(V-\mathcal{A}V), X(x_j)) \\ &+\sum_{i=1}^J (V-V_n)(x_i)\Phi(\mathcal{L}\phi_i, X(x_j)) - \Psi(\partial_\alpha\mathcal{L}(\mathcal{A}V-V), X(x_j)) \\ &+\sum_{i=1}^J (V-V_n)(x_i)\Psi(\partial_\alpha\mathcal{L}\phi_i, X(x_j)) + \Psi(\mathcal{L}(Z-\mathcal{A}Z), X(x_j)) \\ &+\sum_{i=1}^J (Z-Z_n)(x_i)\Psi(\mathcal{L}\phi_i, X(x_j))\Big].\end{aligned}$$

We use (17) to deduce that $\text{Var}^n Z_{n+1}(x_j)$ is bounded by

$$\begin{aligned}&\frac{1}{M}\Big[\sqrt{\mathcal{V}_\Phi(V-\mathcal{A}V)} + \sqrt{\mathcal{V}_\Psi(\mathcal{L}^{-1}\partial_\alpha\mathcal{L}(\mathcal{A}V-V))} \\ &+\textstyle\sum_{i=1}^J |V-V_n|(x_i)\big(\sqrt{\mathcal{V}_\Phi(\phi_i)} + \sqrt{\mathcal{V}_\Psi(\mathcal{L}^{-1}\partial_\alpha\mathcal{L}\phi_i)}\big) \\ &+\sqrt{\mathcal{V}_\Psi(Z-\mathcal{A}Z)} + \textstyle\sum_{i=1}^J |Z-Z_n|(x_i)\sqrt{\mathcal{V}_\Psi(\phi_i)}\Big]^2,\end{aligned}$$

Now, we use (19) to deduce that

$$\begin{aligned}z_{n+1} \le \frac{2}{M}\Big\{&\big(\sqrt{\mathcal{V}_\Phi(V-\mathcal{A}V)} + \sqrt{\mathcal{V}_\Psi(\mathcal{L}^{-1}\partial_\alpha\mathcal{L}(\mathcal{A}V-V))} \\ &+\Big[\textstyle\sum_{i=1}^J \sqrt{\mathcal{V}_\Phi(\phi_i)} + \sqrt{\mathcal{V}_\Psi(\mathcal{L}^{-1}\partial_\alpha\mathcal{L}\phi_i)}\Big]^2 v_n \\ &+\sqrt{\mathcal{V}_\Psi(Z-\mathcal{A}Z)} + \Big[\textstyle\sum_{i=1}^J \sqrt{\mathcal{V}_\Psi(\phi_i)}\Big]^2 z_n\Big\},\end{aligned}$$

and Theorem 2 follows.

References

Atkeson, C. G., Schaal, S. A., and Moore, A. W. (1997). Locally weighted learning. *AI Review*, 11.

Baxter, J. and Bartlett, P. (2001). Infinite-horizon gradient-based policy search. *Journal of Artificial Intelligence Research*, 15:319–350.

Glynn, P. (1987). Likelihood ratio gradient estimation: an overview. In Thesen, A., Grant, H., and Kelton, W., editors, *Proceedings of the 1987 Winter Simulation Conference*, pages 366–375.

Gobet, E. and Maire, S. (2005). Sequential control variates for functionals of markov processes. *To appear in SIAM Journal on Numerical Analysis*.

Greensmith, E., Bartlett, P., and Baxter, J. (2005). Variance reduction techniques for gradient estimates in reinforcement learning. *Journal of Machine Learning Research*, 5:1471–1530.

Halton, J. H. (1970). A retrospective and prospective survey of the monte carlo method. *SIAM Review*, 12(1):1–63.

Halton, J. H. (1994). Sequential monte carlo techniques for the solution of linear systems. *Journal of Scientific Computing*, 9:213–257.

Hammersley, J. and Handscomb, D. (1964). *Monte Carlo Methods*. Chapman and Hall.

Hastie, T., Tibshirani, R., and Friedman, J. (2001). *The Elements of Statistical Learning*. Springer Series in Statistics.

Kollman, C., Baggerly, K., Cox, D., and Picard, R. (1999). Adaptive importance sampling on discrete markov chains. *The Annals of Applied Probability*, 9(2):391–412.

Konda, V. R. and Borkar, V. S. (1999). Actor-critic-type learning algorithms for markov decision processes. *SIAM Journal of Control and Optimization*, 38:1:94–123.

Maire, S. (2003). An iterative computation of approximations on korobov-like spaces. *J. Comput. Appl. Math.*, 54(6):261–281.

Marbach, P. and Tsitsiklis, J. N. (2003). Approximate gradient methods in policy-space optimization of markov reward processes. *Journal of Discrete Event Dynamical Systems*, 13:111–148.

Reiman, M. and Weiss, A. (1986). Sensitivity analysis via likelihood ratios. In Wilson, J., Henriksen, J., and Roberts, S., editors, *Proceedings of the 1986 Winter Simulation Conference*, pages 285–289.

Sutton, R. (1988). Learning to predict by the method of temporal differences. *Machine Learning*, pages 9–44.

Sutton, R., McAllester, D., Singh, S., and Mansour, Y. (2000). Policy gradient methods for reinforcement learning with function approximation. *Neural Information Processing Systems. MIT Press*, pages 1057–1063.

Williams, R. J. (1992). Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8:229–256.