Error analysis of biased stochastic algorithms for a class of integral equations.

Ivan Dimov[1], Rayna Georgieva[1], Venelin Todorov[1]

IICT, Bulgarian Academy of Sciences
Acad. G. Bonchev 25 A, 1113 Sofia, Bulgaria
idimov@parallel.bas.bg, rayna@parallel.bas.bg, dvespas@mail.bg

1 Introduction

In this paper we discuss error analysis of biased stochastic algorithms for a class of integral equations. There are unbiased and biased stochastic algorithms, but the latter algorithms are more interesting , because there are two errors in them- stochastic and systematic errors. The systematic error depends both on the number of iterations performed and the characteristic values of the iteration operator, while the stochastic errors depend on the probabilistic nature of the algorithm. In order to obtain good results the stochastic error r_N must be approximately equal to the systematic error r_k that is

$$r_N = O(r_k).$$

The problem of balancing of the errors is closely connected with the problem of obtaining an optimal ratio between the number of realizations N of the random variable and the mean value k of the number of steps in each random trajectory [1]. The problem of balancing of both systematic and stochastic error is very important when Monte Carlo algorithms are used. The balancing of errors (both, systematic and stochastic) allows to get an approximation of the quantity of interest in the most efficient way by fixing the number of samples N and the number of iterations k if the error is fixed.

2 Monte Carlo algorithm and error estimates

We study the Fredholm integral equation of the second kind:

$$u\left(x\right) = \int_{\Omega} k\left(x,x'\right) u\left(x'\right) dx' + f\left(x\right) \ \ or\ u = \mathcal{K}u + f, \tag{1}$$

We want to construct a Monte Carlo algorithm to evaluate the linear functional from the solution, denoted by the following expression:

$$J(u) = \int \varphi(x) u(x) dx = \left(\varphi, u\right). \tag{2}$$

We construct a Monte Carlo method for integral equations based on discrete Markov chains. We define a set of permissible densities. In correspondance with the initial and transition probabilities we define a discrete Markov chain. If $u^{(0)} \equiv f$ then Monte Carlo algorithm for integral equations is given by the following expressions [2]:

$$E\theta_k[\varphi] = \left(\varphi, u^{(k)}\right),\ \theta_k[\varphi] = \frac{\varphi\left(x_0\right)}{\pi\left(x_0\right)} \sum_{j=0}^{k} W_j f\left(x_j\right),$$

$$W_0 = 1,\ W_j = W_{j-1} \frac{k(x_{j-1}, x_j)}{p(x_{j-1}, x_j)},\ j = 1, \ldots, k,$$

$$\left(\varphi, u^{(k)}\right) \approx \frac{1}{N} \sum_{n=1}^{N} \theta_k[\varphi]_n.$$

We obtain the following estimate for the probable and the systematic error:

$$r_N \leq \frac{0.6745 \|f\|_{L_2} \|\varphi\|_{L_2}}{\sqrt{N}\left(1 - \|\mathcal{K}\|_{L_2}\right)}.$$

$$r_k \leq \frac{\|\varphi\|_{L_2} \|f\|_{L_2} \|\mathcal{K}\|_{L_2}^{k+1}}{1 - \|\mathcal{K}\|_{L_2}}.$$

3 Balancing of the errors

Theorem 1 (Main result). *For a Fredholm integral equation with a preliminary given error δ, the lower bounds for N and k for the Monte Carlo algorithm with a balancing of the errors are:*

$$N \geq \left(\frac{1.349 \|\varphi\|_{L_2} \|f\|_{L_2}}{\delta\left(1 - \|\mathcal{K}\|_{L_2}\right)}\right)^2,\ k \geq \frac{\ln \frac{\delta\left(1 - \|\mathcal{K}\|_{L_2}\right)}{2 \|\varphi\|_{L_2} \|f\|_{L_2} \|\mathcal{K}\|_{L_2}}}{\ln \|\mathcal{K}\|_{L_2}}.$$

Theorem 2 (The optimal ratio).

In a Monte Carlo algorithm based on a balancing of the errors, if N is close to its lower bound, then for k:

$$k \geq \frac{\ln \dfrac{0.6745}{\|\mathcal{K}\|_{L_2} \sqrt{N}}}{\ln \|\mathcal{K}\|_{L_2}}.$$

Theorem 3 (Equivalence). *If N is close to the smallest possible natural number for which*

$$N \geq \left(\frac{1.349 \|\varphi\|_{L_2} \|f\|_{L_2}}{\delta\left(1 - \|\mathcal{K}\|_{L_2}\right)}\right)^2$$

then the two obtained lower bounds for k are equivalent.

Corollary 1. *In the next tests with a preliminary given error in Monte Carlo algorithm with a balancing of the errors we choose*

$$N = \left\lceil \left(\frac{1.349 \|\varphi\|_{L_2} \|f\|_{L_2}}{\delta \left(1 - \|\mathcal{K}\|_{L_2}\right)} \right)^2 \right\rceil, \; k = \left\lceil \frac{\ln \frac{0.6745}{\|\mathcal{K}\|_{L_2} \sqrt{N}}}{\ln \|\mathcal{K}\|_{L_2}} \right\rceil. \tag{3}$$

Corollary 2. *One can first choose k to be close to its lower bound in the theorem and to receive the following inequality for N:*

$$N \geq \frac{0.455}{\|\mathcal{K}\|_{L_2}^{2k+2}}.$$

Corollary 3. *One can also choose the following values for N and k:*

$$k = \left\lceil \frac{\ln \frac{\delta\left(1 - \|\mathcal{K}\|_{L_2}\right)}{2 \|\varphi\|_{L_2} \|f\|_{L_2} \|\mathcal{K}\|_{L_2}}}{\ln \|\mathcal{K}\|_{L_2}} \right\rceil, \; N = \left\lceil \frac{0.455}{\|\mathcal{K}\|_{L_2}^{2k+2}} \right\rceil.$$

4 Numerical examples and results

4.1 Example 1

The first example is:

$$u\left(x\right) = \int_{\Omega} k\left(x, x'\right) u\left(x'\right) dx' + f\left(x\right),$$

$\Omega \equiv [0, 1]$, $k\left(x, x'\right) = \dfrac{1}{6} e^{x + x'}$, $f\left(x\right) = 6x - e^x$, $\varphi(x) = \delta(x)$, $u_{exact}(x) = 6x$. We want to find the solution in the middle of the interval. We make 20 algorithm runs on Intel Core i5-2410M @ 2.3 GHz.

4.2 Example 2

The next example is analytically tractable model taken from the biology from population growth model:

$$u\left(x\right) = \int_{\Omega} k\left(x, x'\right) u\left(x'\right) dx' + f\left(x\right),$$

$\Omega \equiv [0, 1]$, $k\left(x, x'\right) = \frac{1}{3} e^x$, $f\left(x\right) = \frac{2}{3} e^x$, $\varphi(x) = \delta(x)$, $u_{exact}(x) = e^x$.

4.3 Example 3

We study the following example taken from neuron networking:

$$u\left(x\right)=\int_{\Omega}k\left(x,x'\right)u\left(x'\right)dx'+f\left(x\right),$$

$\Omega\equiv[-2,2]$, $k\left(x,x'\right)=\dfrac{0.055}{1+e^{-3x}}+0.07$, $f\left(x\right)=0.02\left(3x^2+e^{-0.35x}\right)$, $\varphi(x)=0.7((x{+}1)^2\cos(5x){+}20)$. We want to find (φ,u), where $\varphi(x)=0.7((x{+}1)^2\cos(5x){+}20)$, $u_{exact}(x)=8.98$.

In this example MAO gives the best results and the experimental relative error is very close to the expected theoretical error.

4.4 Example 4

We consider the MAO algorithm to estimate the functional (2). The function $u\left(x\right)$ is a solution of the following integral equation with polynomial nonlinearity: [1]:

$$u\left(x\right)=\int_{\Omega}\int_{\Omega}k\left(x,y,z\right)u\left(y\right)u\left(z\right)dydz+f\left(x\right), \tag{4}$$

where $\Omega=\mathbf{E}\equiv[0,1]$ and $x\in\mathbb{R}^1$

In our test $k\left(x,y,z\right)=\frac{x(a_2y-z)^2}{a_1}$ and $f(x)=c-\frac{x}{a_3}$. We consider the results for calculating the linear functional (2) for $a_1=11, a_2=4, a_3=12, c=0.5$.

5 Conclusion

An original approach to the problem of controlling the error in non deterministic methods is presented. Monte Carlo method based on balancing of the systematic error and probability error is presented. Lower bounds for N and k are obtained. Meaningful numerical examples and results are discussed. Monte Carlo algorithms with various initial and transition probabilities are compared. Experimental relative errors confirm expected theoretical errors. Monte Carlo algorithms with probabilities chosen to be proportional to the function from the linear functional under consideration and the kernel, respectively, give more reliable results. The balancing of errors (both systematic and stochastic) allows to increase the accuracy of the solution if the computational effort is fixed or to reduce the computational complexity if the error is fixed.

References

1. I. Dimov, Monte Carlo Methods for Applied Scientists, New Jersey, London, Singapore, World Scientific, 2008, 291p.
2. I. Sobol. Numerical methods Monte Carlo. Nauka, Moscow, 1973.