Method of Reducing Variance of Monte Carlo

Pan Yiming, Tai Lo Yeung, Qiwei Liu

2021/06/07

1 Introduction

In this project, we investigate three methods to reduce the variance of Monte Carlo, which can improve the simulation efficiency. We verify the effect of this three methods by simulating Heston model. The three methods are antithetic variate, control variate, and conditional expectation. In addition, we combine the conditional expectation method and antithetic variate, which gives the best results.

2 Basic Theory

2.1 Control Variate Method

Suppose again that we wish to estimate $\theta := \mathbb{E}[Y]$ where $Y = h(\mathbf{X})$ is the output of a simulation experiment. Suppose that Z is also an output of the simulation or that we can easily output it if we wish. Finally, we assume that we know $\mathbb{E}[Z]$. Then we can construct many unbiased estimators of θ :

1. $\widehat{\theta} = Y$, our usual estimator
2. $\widehat{\theta}_c := Y + c(Z - \mathbb{E}[Z])$

for any $c \in \mathbb{R}$. The variance of $\widehat{\theta}_c$ satisfies:

$$\operatorname{Var}\left(\widehat{\theta}_c\right) = \operatorname{Var}(Y) + c^2 \operatorname{Var}(Z) + 2c \operatorname{Cov}(Y, Z)$$

and we can choose c to minimize this quantity. Simple calculus then implies the optimal value of c is given by:

$$c^* = -\frac{\operatorname{Cov}(Y, Z)}{\operatorname{Var}(Z)}$$

and that the minimized variance satisfies:

$$\begin{aligned} \operatorname{Var}\left(\widehat{\theta}_{c^*}\right) &= \operatorname{Var}(Y) - \frac{\operatorname{Cov}(Y, Z)^2}{\operatorname{Var}(Z)} \\ &= \operatorname{Var}(\widehat{\theta}) - \frac{\operatorname{Cov}(Y, Z)^2}{\operatorname{Var}(Z)} \end{aligned}$$

In order to achieve a variance reduction it is therefore only necessary that $\operatorname{Cov}(Y, Z) \neq 0$. The new resultin Monte Carlo algorithm proceeds by generating n samples of Y and Z and then setting

$$\widehat{\theta}_{c^*} = \frac{\sum_{i=1}^{n} \left(Y_i + c^* \left(Z_i - \mathbb{E}[Z]\right)\right)}{n}$$

There is a problem with this, however, as we usually do not know $\operatorname{Cov}(Y, Z)$. We overcome this problem by doing p pilot simulations and setting

$$\widehat{\operatorname{Cov}}(Y, Z) = \frac{\sum_{j=1}^{p} \left(Y_j - \bar{Y}_p\right)\left(Z_j - \mathbb{E}[Z]\right)}{p - 1}$$

If it is also the case that $\mathrm{Var}(Z)$ is unknown, then we also estimate it with

$$\widehat{\mathrm{Var}}(Z)=\frac{\sum_{j=1}^{p}\left(Z_j-\mathbb{E}[Z]\right)^2}{p-1}$$

and finally set

$$\widehat{c}^*=-\frac{\widehat{\mathrm{Cov}}(Y,Z)}{\widehat{\mathrm{Var}}(Z)}$$

Algorithm 1 Control Variate Simulation Algorithm for Estimating $\mathbb{E}[Y]$

for i = 1 to n **do**
  generate (Y_i, Z_i)
  set $V_i=Y_i+\widehat{c}^*\left(Z_i-\mathbb{E}[Z]\right)$
end for
set $\hat{\theta}_{\widehat{c}^*}=\bar{V}_n=\sum_{i=1}^{n} V_i/n$

2.2 Antithetic Variates

As usual we would like to estimate $\theta=\mathbb{E}[h(\mathbf{X})]=\mathbb{E}[Y]$, and suppose we have generated two samples, Y_1 and Y_2. Then an unbiased estimate of θ is given by $\hat{\theta}=\left(Y_1+Y_2\right)/2$ with:

$$\mathrm{Var}(\widehat{\theta})=\frac{\mathrm{Var}\left(Y_1\right)+\mathrm{Var}\left(Y_2\right)+2\,\mathrm{Cov}\left(Y_1,Y_2\right)}{4}$$

If Y_1 and Y_2 are *iid*, then $\mathrm{Var}(\widehat{\theta})=\mathrm{Var}(Y)/2$. However, we could reduce $\mathrm{Var}(\widehat{\theta})$ if we could arrange it so that $\mathrm{Cov}\left(Y_1,Y_2\right)<0$. We now describe the method of antithetic variates for doing this. We will begin with the case where Y is a function of IID $U(0,1)$ random variables so that $\theta=\mathbb{E}[h(\mathbf{U})]$ where $\mathbf{U}=\left(U_1,\ldots,U_m\right)$ and the U_i 's are IID $\sim U(0,1)$. The usual Monte Carlo algorithm, assuming we use $2n$ samples, is shown below.

Algorithm 2 Usual Simulation Algorithm for Estimating θ

for i = 1 to 2n **do**
  generate $\mathrm{U_i}$
  set $Y_i=h\left(\mathbf{U_i}\right)$
end for
set $\hat{\theta}_{2n}=\bar{Y}_{2n}=\sum_{i=1}^{2n} Y_i/2n$

In the above algorithm, however, we could also have used the $-U_i$'s to generate sample Y values, since if $U_i\sim U(0,1)$, then so too is $1-U_i$. We can use this fact to construct another estimator of θ as follows. As before, we set $Y_i=h\left(\mathbf{U_i}\right)$, where $\mathbf{U_i}=\left(U_1^{(i)},\ldots,U_m^{(i)}\right)$. We now also set $\tilde{Y}_i=h\left(-\mathbf{U_i}\right)$, where we use $-\mathbf{U_i}$ to denote $\left(-U_1^{(i)},\ldots,-U_m^{(i)}\right)$. Note that $\mathbb{E}\left[Y_i\right]=\mathbb{E}\left[\tilde{Y}_i\right]=\theta$ so that in particular, if

$$Z_i:=\frac{Y_i+\tilde{Y}_i}{2}$$

then $\mathbb{E}\left[Z_i\right]=\theta$ so that Z_i is an also unbiased estimator of θ. If the U_i 's are IID, then so too are the Z_i 's and we can use them as usual to compute approximate confidence intervals for θ. We say that U_i and $-U_i$ are antithetic variates and we then have the following antithetic variate simulation algorithm.

Algorithm 3 Antithetic Variate Simulation Algorithm for Estimating θ

for i = 1 to n **do**
  generate U_i
  set $Y_i = h(\mathbf{U_i})$ and $\tilde{Y}_i = h(-\mathbf{U_i})$
  set $Z_i = \left(Y_i + \tilde{Y}_i\right)/2$
end for
set $\hat{\theta}_{n,a} = \bar{Z}_n = \sum_{i=1}^{n} Z_i/n$

2.3 The Conditional Monte Carlo Simulation Algorithm

Summarizing the previous discussion, we want to estimate $\theta := \mathbb{E}[h(\mathbf{X})] = \mathbb{E}[Y]$ using conditional Monte Carlo. To do so, we must have another variable or vector, $\mathbf{Z}$, that satisfies the following requirements:

1. $\mathbf{Z}$ can be easily simulated

2. $V := g(\mathbf{Z}) := \mathbb{E}[Y \mid \mathbf{Z}]$ can be computed exactly.

This means that we can simulate a value of V by first simulating a value of $\mathbf{Z}$ and then setting $V = g(\mathbf{Z}) = \mathbb{E}[Y \mid \mathbf{Z}]$. We therefore have the following algorithm for estimating θ:

Algorithm 4 Conditional Monte Carlo Algorithm for Estimating θ

for i = 1 to n **do**
  generate Z_i
  compute $g(\mathbf{Z}_i) = \mathbb{E}[Y \mid \mathbf{Z}_i]$
  set $V_i = g(\mathbf{Z}_i)$
end for
set $\widehat{\theta}_{n,cm} = \bar{V}_n = \sum_{i=1}^{n} V_i/n$

3 Heston model

3.1 Heston model formulas

Assume a Heston model of the type :

$$dS_t = rS_t dt + \sigma_t S_t \left(\rho dW_t + \sqrt{1-\rho^2} dB_t\right)$$

where

$$d\sigma_t^2 = \kappa\left(\theta - \sigma_t^2\right) dt + \nu\sqrt{\sigma_t^2} dW_t$$

Here ρ denotes the correlation parameter. We are going to simulate the corresponding asset prices using Milstein scheme as the baseline.

3.2 Milstein scheme

This method uses a discretization for the square volatility process, that takes into account the corresponding second order term in the Taylor expansion:

$$\sigma_{i+1}^2 = \sigma_i^2 + \kappa\left(\theta - \sigma_i^2\right)\Delta t + \nu\sqrt{\sigma_i^2}\Delta W_i + \frac{\nu^2}{4}\left(\Delta W_i^2 - \Delta t\right)$$

Notice that, even when the Heston volatility is always greater of equal than zero, the simulated σ_i^2 can be negative. Then $\sqrt{\sigma_i^2}$ is not well defined.

To overcome this problem, practicioners consider the absorbing asumption (if $\sigma_i^2 < 0$, then $\sigma_i^2 = 0$, of the reflecting assumption (if $\sigma_i^2 < 0$, then $\sigma_i^2 = -\sigma_i^2$) . In practice, this method needs a huge number of time intervals to achieve good results.

Once we simulate the volatility process, we can simulate asset prices using that $S_i = S_0 \exp(X_i)$, where

$$X_{i+1} = X_i + \left(r - 0.5\sigma_i^2\right)\Delta t + \sqrt{\sigma_i^2}\left(\rho\Delta W_i + \sqrt{1-\rho^2}\Delta B_i\right)$$

and where B is a Brownian motion independent on W.

3.3 Combining Antithetic Variate with Milstein scheme

.

When we apply antithetic variate to Milstein scheme of Heston model, we take the two Brownian motions W_i and B_i as the antithetic variates. And then for one simulation, we compute the asset price by W_i , B_i and $-W_i$ and $-B_i$ to get S and S'. Finally, we compute the call price as

$$\frac{(max(S-K,0) + max(S'-K,0))}{2}$$

3.4 Combining Control Variate with Milstein scheme

.

When we apply antithetic variate to Milstein scheme of Heston model, we take the finanl asset ptice S_T as the control variate. The logic is described in Algorithm 1 of section 2.1.

3.5 Conditional Monte Carlo

If we assume r=0, then we have

$$\begin{aligned}
S_T =& S_0 \exp\left(-\frac{1}{2}\int_0^T \sigma_s^2 ds + \int_0^T \sigma_s\left(\rho dW_s + \sqrt{1-\rho^2}dB_s\right)\right). \\
=& S_0 \exp\left(-\frac{\rho^2}{2}\int_0^T \sigma_s^2 ds + \int_0^T \rho\sigma_s\left(dW_s\right)\right) \\
& \times \exp\left(-\frac{(1-\rho^2)}{2}\int_0^T \sigma_s^2 ds + \sqrt{1-\rho^2}\int_0^T \sigma_s dB_s\right)
\end{aligned}$$

S can be seen (conditioned to W) as a process with deterministic volatility given by

$$\sqrt{(1-\rho^2)}\sigma$$

and the initial asset price is

$$S_0' = S_0 \exp\left(-\frac{\rho^2}{2}\int_0^T \sigma_s^2 ds + \rho\int_0^T \sigma_s\left(dW_s\right)\right)$$

Then

$$E\left(BS\left(T, S_T, K, v_T\right) \mid W\right) = BS\left(0, S_0', K, \sqrt{(1-\rho^2)}v_0\right)\Big]$$

Notice that the above formula allows us to compute the Monte Carlo price with no need to simulate the Brownian motion B. That is, the steps will be

- Simulate the processes W and σ
- Compute S_0' and v_0
- Compute $BS\left(0, S_0', K, \sqrt{(1-\rho^2)}v_0\right)$

Then, we repeat the previous steps n times to get n simulations of $BS\left(0, S_0', K, \sqrt{(1-\rho^2)}v_0\right)$ and we take the mean.

4 Experiment and results

4.1 Experiments

In the experiments, we use different ν , ρ , κ , σ_0^2 , θ to simulate the call price. At the same time, we change the simulation times for every group parameters.

The common parameters is $T = 1$, $K = 100$, $S_0 = 100$, $r = 0$.

method	mean	std	average time	times	ν	ρ	κ	σ_0^2	θ
Milstein	5.9487	0.3843	4.55	500	0.3	-0.4	0.5	0.02	0.05
Milstein + antithetic	5.8511	0.3467	4.49	500	0.3	-0.4	0.5	0.02	0.05
Milstein + control	5.8806	0.2526	4.61	500	0.3	-0.4	0.5	0.02	0.05
Milstein + conditional	5.9002	0.0602	2.79	500	0.3	-0.4	0.5	0.02	0.05
Milstein + conditional + antithetic	5.8955	0.0320	2.85	500	0.3	-0.4	0.5	0.02	0.05
Milstein	5.8660	0.2507	8.75	1000	0.3	-0.4	0.5	0.02	0.05
Milstein + antithetic	5.9022	0.2201	8.73	1000	0.3	-0.4	0.5	0.02	0.05
Milstein + control	5.8744	0.1733	9.26	1000	0.3	-0.4	0.5	0.02	0.05
Milstein + conditional	5.8860	0.0464	5.81	1000	0.3	-0.4	0.5	0.02	0.05
Milstein + conditional + antithetic	5.9005	0.0186	5.88	1000	0.3	-0.4	0.5	0.02	0.05
Milstein	5.8821	0.2330	17.47	2000	0.3	-0.4	0.5	0.02	0.05
Milstein + antithetic	5.8568	0.1502	17.67	2000	0.3	-0.4	0.5	0.02	0.05
Milstein + control	5.8954	0.1095	18.49	2000	0.3	-0.4	0.5	0.02	0.05
Milstein + conditional	5.8909	0.0268	11.07	2000	0.3	-0.4	0.5	0.02	0.05
Milstein + conditional + antithetic	5.8904	0.0166	11.01	2000	0.3	-0.4	0.5	0.02	0.05
Milstein	8.8014	0.5634	4.59	500	0.4	-0.5	1.0	0.03	0.10
Milstein + antithetic	8.8228	0.4757	4.98	500	0.4	-0.5	1.0	0.03	0.10
Milstein + control	8.8470	0.2209	4.49	500	0.4	-0.5	1.0	0.03	0.10
Milstein + conditional	8.8198	0.1805	3.15	500	0.4	-0.5	1.0	0.03	0.10
Milstein + conditional + antithetic	8.8105	0.0333	4.31	500	0.4	-0.5	1.0	0.03	0.10
Milstein	8.8282	0.4355	9.86	1000	0.4	-0.5	1.0	0.03	0.10
Milstein + antithetic	8.9059	0.3390	10.12	1000	0.4	-0.5	1.0	0.03	0.10
Milstein + control	8.8226	0.2560	9.31	1000	0.4	-0.5	1.0	0.03	0.10
Milstein + conditional	8.8269	0.1260	6.09	1000	0.4	-0.5	1.0	0.03	0.10
Milstein + conditional + antithetic	8.8022	0.0251	6.21	1000	0.4	-0.5	1.0	0.03	0.10
Milstein	8.8819	0.2481	18.70	2000	0.4	-0.5	1.0	0.03	0.10
Milstein + antithetic	8.7755	0.2077	18.21	2000	0.4	-0.5	1.0	0.03	0.10
Milstein + control	8.8315	0.1680	18.35	2000	0.4	-0.5	1.0	0.03	0.10
Milstein + conditional	8.8013	0.0816	12.90	2000	0.4	-0.5	1.0	0.03	0.10
Milstein + conditional + antithetic	8.8081	0.0185	12.19	2000	0.4	-0.5	1.0	0.03	0.10

method	mean	std	average time	times	ν	ρ	κ	σ_0^2	θ
Milstein	11.3646	0.8003	5.39	500	0.5	-0.6	1.5	0.04	0.15
Milstein + antithetic	11.4239	0.6148	5.18	500	0.5	-0.6	1.5	0.04	0.15
Milstein + control	11.4842	0.5031	4.52	500	0.5	-0.6	1.5	0.04	0.15
Milstein + conditional	11.5027	0.3082	3.03	500	0.5	-0.6	1.5	0.04	0.15
Milstein + conditional + antithetic	11.4522	0.0933	3.04	500	0.5	-0.6	1.5	0.04	0.15
Milstein	11.6072	0.5441	9.33	1000	0.5	-0.6	1.5	0.04	0.15
Milstein + antithetic	11.4869	0.3413	9.45	1000	0.5	-0.6	1.5	0.04	0.15
Milstein + control	11.4277	0.2460	10.64	1000	0.5	-0.6	1.5	0.04	0.15
Milstein + conditional	11.4767	0.2353	5.97	1000	0.5	-0.6	1.5	0.04	0.15
Milstein + conditional + antithetic	11.4591	0.0551	5.98	1000	0.5	-0.6	1.5	0.04	0.15
Milstein	11.4933	0.3869	18.53	2000	0.5	-0.6	1.5	0.04	0.15
Milstein + antithetic	11.3995	0.2855	19.25	2000	0.5	-0.6	1.5	0.04	0.15
Milstein + control	11.4526	0.1642	18.72	2000	0.5	-0.6	1.5	0.04	0.15
Milstein + conditional	11.4673	0.1661	13.15	2000	0.5	-0.6	1.5	0.04	0.15
Milstein + conditional + antithetic	11.4426	0.0374	12.18	2000	0.5	-0.6	1.5	0.04	0.15

4.2 Conclusions

- When we combine all the three methods with normal Milstein, they all consistently outperformance the crude Milstein scheme.
- The conditional expectation method can reduce the simulation time and variance significantly.
- The best result is given by combination of crude Milstein, conditional expectation, and antithetic variate, which has smallest standard deviation and shortest time-consuming.