MATH 565 Monte Carlo Methods in Finance

Fred J. Hickernell **Test 2** **Wednesday, October 31, 2012**

Instructions:

i. *This test consists of FOUR questions. Answer all of them.*

ii. *The time allowed is 75 minutes.*

iii. *This test is closed book, but you may use 4 double-sided letter-size sheets of notes.*

iv. *(Programmable) calculators are allowed, but they must not have stored text. Computers are also allowed, but only using MATLAB, Mathematica, or JMP. No internet access.*

v. *Show all your work to justify your answers. Answers without adequate justification will not receive credit. Write out pseudo-code for the programs that you run to get your answers.*

1. (25 marks)
Consider the Brownian motion, B, defined on the interval $[0, \infty)$. The Brownian motion satisfies

$$B(t) \sim \mathcal{N}(0,t), \quad \mathrm{cov}(B(s), B(t)) = \min(s,t) \qquad \text{for all } 0 \le s \le t < \infty.$$

Consider also two IID $\mathcal{N}(0,1)$ random numbers from a (pseudo)-random number generator: $X_1 = -1.7967$, $X_2 = 2.4191$.

a) For a single Brownian path, first find the value of $B(1/4)$, and then use that to find the value of $B(1/2)$ using a time-discretization method and the random numbers above.

Answer:

$$B(1/4) = B(0) + \sqrt{1/4}X_1 = 0 + (1/2)(-1.7967) = -0.8984,$$
$$B(1/2) = B(1/4) + \sqrt{1/4}X_2 = -0.8984 + (1/2)(2.4191) = 0.3112$$

b) For another single Brownian path, first find the value of $B(1/2)$, and then use that to find the value of $B(1/4)$ using the random numbers above.

Answer: Now we use a Brownian bridge:

$$B(1/2) = B(0) + \sqrt{1/2}X_1 = 0 + \sqrt{1/2}(-1.7967) = -1.2705,$$
$$B(1/4) = [B(0) + B(1/2)]/2 + \sqrt{1/8}X_2 = -1.2705/2 + \sqrt{1/8}(2.4191) = 0.2200$$

2. (20 marks)
Consider a stock price path, $S(t)$, $t \ge 0$, modeled by a geometric Brownian motion with initial price $S(0) = \$50$, interest rate $r = 3\%$, and volatility $\sigma = 50\%$.

a) Use the two Brownian paths that you generated in the previous problem to find *two pairs* of values $S(1/4)$ and $S(1/2)$.

Answer: First note that geometric Brownian motion model says that

$$S(t) = S(0)e^{(r-\sigma^2/2)t+\sigma B(t)} = 50e^{(0.03-0.5^2/2)t+0.5B(t)} = 50e^{-0.095t+0.5B(t)}$$

For the time discretization, this becomes

$$S(1/4) = 50e^{-0.095(1/4)+0.5(-0.8984)} = 31.16,$$
$$S(1/2) = 50e^{-0.095(1/2)+0.5(0.3112)} = 55.71,$$

and for the Brownian Bridge construction this becomes

$$S(1/4) = 50e^{-0.095(1/4)+0.5(0.2200)} = 54.51,$$
$$S(1/2) = 50e^{-0.095(1/2)+0.5(-1.2705)} = 25.26.$$

b) For each of these stock price paths, what is the discounted payoff of a *lookback call* option expiring one half year from purchase and being monitored every quarter?

Answer: For the time discretization, the discounted lookback call payoff is

$$[S(1/2) - \min(S(0), S(1/4), S(1/2))]e^{-r(1/2)} = [55.71 - 31.16]e^{-0.03(1/2)} = 24.18.$$

For the Brownian bridge construction, the discounted lookback call payoff is

$$[S(1/2) - \min(S(0), S(1/4), S(1/2))]e^{-r(1/2)} = [25.26 - 25.26]e^{-0.03(1/2)} = 0.$$

3. (20 marks)
Consider the same stock price model as in the previous problem, but now consider an *American put* option with a strike price of \$50 that expires at $t = 1/2$. Actually, this is a Bermudan put option because one may exercise only at times 0, 1/4, and 1/2. Here are 10 stock paths:

i	$S(0)$	$S(1/4)$	$S(1/2)$
1	50.00	62.96	108.64
2	50.00	29.88	44.13
3	50.00	44.27	34.77
4	50.00	33.17	29.19
5	50.00	67.70	53.52
6	50.00	27.15	23.66
7	50.00	72.95	84.22
8	50.00	53.91	37.51
9	50.00	46.38	62.11
10	50.00	54.49	43.20

The exercise boundary at $t = 1/4$ is \$34.47.

a) Which of these paths should be exercised at $t = 0$? $t = 1/4$? $t = 1/2$? never?

Answer: Since the strike price is the initial price, the option never should be exercised at $t = 0$. The paths $i = 2, 4, 6$ are below the exercise boundary at $t = 1/4$ and should be exercised then. The paths $i = 3, 8, 10$ exercise at $t = 1/2$, and the rest of the paths are never exercised.

b) What is the average discounted put payoff for these ten paths assuming that they are exercised as prescribed by the exercise boundary?

Answer:

$$\frac{1}{10}\Big[0 + 20.12\mathrm{e}^{-0.03/4} + 15.23\mathrm{e}^{-0.03/2} + 16.83\mathrm{e}^{-0.03/4} + 0 + 22.85\mathrm{e}^{-0.03/4} + 0 + 12.49\mathrm{e}^{-0.03/2} + 0 + 6.80\mathrm{e}^{-0.03/2}\Big]$$
$$= \frac{1}{10}[0 + 19.96 + 15.00 + 16.70 + 022.68 + 012.30 + 0 + 6.69] = 9.33$$

4. (35 marks)
Consider the same stock price model as in the previous two problems. Consider the *Asian arithmetic mean call option* expiring in $T = 1/2$ of a year, where the strike price is $K = \$50$, and the discounted payoff is

$$\max\left(\frac{1}{2}[S(T/2) + S(T)] - K, 0\right)\mathrm{e}^{-rT}.$$

a) Compute the fair price of this option to the nearest \$0.1 using simple Monte Carlo. How many stock paths do you need?

b) The European call option has a price of \$7.3420. Use *control variates* to price this option. Now how many stock paths do you need? What is the savings in terms of reduced sample size?

Answer: The program is long, but many parts are repetitive. The sample size for control variates is about $1/8$ the size of ordinary Monte Carlo.

```
T=1/2; %time to expiry
d=2; %numbe of time steps
delt=T/d; %time step size
K=50; %strike price
tol=0.1; %error tolerance
Europrice=S0*normcdf((log(S0/K)+(r+sigma^2/2)*T)/(sigma*sqrt(T))) ...
    - K*exp(-r*T)*normcdf((log(S0/K)+a*T)/(sigma*sqrt(T)));
%First time with simple Monte Carlo
n=1e3; %initial sample size
xnorm=randn(n,d); %initial normal random variables
```

```
%initial stock paths
spath=S0*exp(cumsum([zeros(n,1) a*repmat(delt,n,d)+sigma*sqrt(delt)*xnorm],2));
Asiancallpay=max(mean(spath(:,2:d+1),2)-K,0).*exp(-r*T); %Asian payoff
stdAsianpay=std(Asiancallpay); %standard deviation of Asian call payoff
n=ceil((2.58*1.2*stdAsianpay/tol)^2) %new sample size to get tolerance
xnorm=randn(n,2); %new sample of normal random variables
%new stock paths
spath=S0*exp(cumsum([zeros(n,1) a*repmat(delt,n,d)+sigma*sqrt(delt)*xnorm],2));
Asiancallpay=max(mean(spath(:,2:d+1),2)-K,0).*exp(-r*T); %Asian payoff
AsianMCprice=mean(Asiancallpay) %mean of payoffs

%Second time with control variates Monte Carlo
n=1e3; %initial sample size
xnorm=randn(n,d); %initial normal random variables
%initial stock paths
spath=S0*exp(cumsum([zeros(n,1) a*repmat(delt,n,d)+sigma*sqrt(delt)*xnorm],2));
Asiancallpay=max(mean(spath(:,2:d+1),2)-K,0).*exp(-r*T); %Asian payoff
meanAsian=mean(Asiancallpay); %sample mean of Asian payoff
Eurocallpay=max(spath(:,d+1)-K,0).*exp(-r*T); %European payoff
beta=(Eurocallpay-Europrice)\(Asiancallpay-meanAsian); %control variate coefficient
AsianCV=Asiancallpay+beta*(Europrice-Eurocallpay); %control variate estimator
stdAsianCV=std(AsianCV); %standard deviation of Asian call payoff control variate
n=ceil((2.58*1.2*stdAsianCV/tol)^2) %new sample size
xnorm=randn(n,2); %new sample of normal random variables
%new stock paths
spath=S0*exp(cumsum([zeros(n,1) a*repmat(delt,n,d)+sigma*sqrt(delt)*xnorm],2));
Asiancallpay=max(mean(spath(:,2:d+1),2)-K,0).*exp(-r*T); %Asian payoff
meanAsian=mean(Asiancallpay); %sample mean of Asian payoff
Eurocallpay=max(spath(:,d+1)-K,0).*exp(-r*T); %European payoff
beta=(Eurocallpay-Europrice)\(Asiancallpay-meanAsian); %control variate coefficient
AsianCV=Asiancallpay+beta*(Europrice-Eurocallpay); %control variate estimator
stdAsianCV=std(AsianCV);
AsianCVprice=mean(AsianCV) %mean of control variate payoffs

n =
      108554
AsianMCprice =
    5.7889
n =
       12337
AsianCVprice =
    5.7015
```