# MATH 565 Monte Carlo Methods in Finance

**Fred J. Hickernell Take-Home Final Exam Due 8 AM, Tuesday, December 4, 2012**

*Instructions:*

i. *This take-home part of the final exam has THREE questions for a total of* 35 *points possible. You should attempt them all.*

ii. *You may consult any book, web page, software repository, notes, old tests, or other inanimate object. You may use the m-files on Blackboard. You* may not *consult any other person face-to-face, by telephone, by email, Facebook, Twitter, LinkedIn or by any other means.* **Sign here to acknowledge that you followed this instruction:**

*Signature* *Date*

iii. *Show all your work to justify your answers. Submit hard copies of this signed cover page and your derivations, programs, output, and explanations to me before the in-class final exam. Answers without adequate justification will not receive credit.*

iv. *In addition, as a precaution, submit soft copies of your programs to the Blackboard Dropbox. If I have difficulty understanding your computational work, I may look at your programs.*

1. (15 marks)

Consider the situation where the daily high temperature, $Y_t$, during the 31 days of the month of January, satisfies the following difference equation:

$$Y_1 \sim \mathcal{N}(27, 4^2), \quad Y_{t+1} = 12 + 0.6Y_t + 5X_t,\ t = 1, 2, \ldots, 30, \quad X_1, X_2, \ldots, X_{30} \text{ IID} \sim \mathcal{N}(0, 1).$$

A weather derivative pays off \$100 any time the high temperature stays below 20 for 3 consecutive days. For example, a sequence of temperatures

| January, $t$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | $\cdots$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Temperature, $Y_t$ | 32 | 25 | 19 | 18 | 16 | 18 | 22 | 23 | 19 | 14 | 16 | 25 | 31 | $\geq 20$ |

would pay off \$300, i.e., \$100 for the sequence starting January 3 plus \$100 for the sequence starting January 4 plus \$100 for the sequence starting January 9. Use the simple Monte Carlo method to price this weather derivative to the nearest \$0.5 with 99% confidence. How many samples are needed?

```
%Simple Monte Carlo
tic
tol=0.5; %error tolerance
d=31; %number of days in January
n=1e4; %initial sample size
x=randn(n,d-1); %innovations
y=zeros(n,d); %initialize temperature array
y(:,1)=27+4*randn(n,1); %initialize temperature
for j=1:d-1;
    y(:,j+1)=12+0.6*y(:,j)+5*x(:,j); %time step the temperature
end
cold=y<20; %which days below 20 degrees
payoff=100*sum(cold(:,1:d-2)&cold(:,2:d-1)&cold(:,3:d),2); %option payoff
```

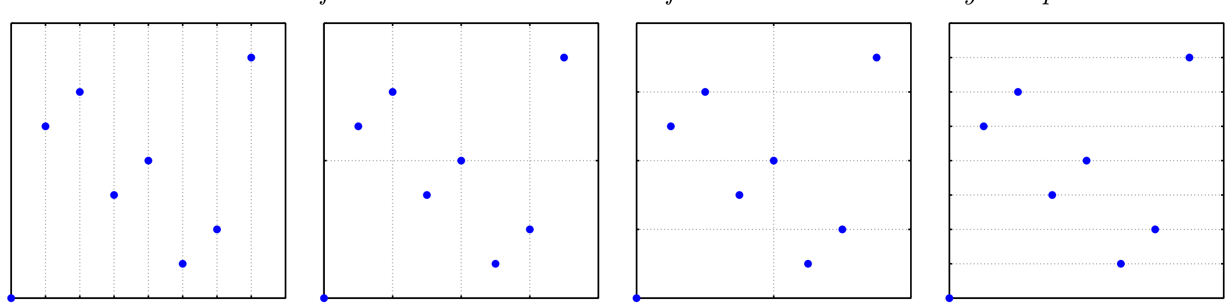
```
n=ceil((2.58*1.1*std(payoff)/tol)^2); %final sample size
x=randn(n,d-1); %innovations
y=zeros(n,d); %initialize temperature array
y(:,1)=27+4*randn(n,1); %initialize temperature
for j=1:d-1;
    y(:,j+1)=12 + 0.6*y(:,j)+5*x(:,j); %time step the temperature
end
cold=y<20; %which days below 20 degrees
payoff=100*sum(cold(:,1:d-2)&cold(:,2:d-1)&cold(:,3:d),2); %option payoff
price=mean(payoff); %option price
err=2.58*std(payoff)/sqrt(n); %estimated error
disp(['The price of this option = $' num2str(price,'%4.2f') ...
    ' +/- $' num2str(err,'%4.2f')])
disp(['    based on ' int2str(n) ' samples'])
toc

The price of this option = $18.10 +/- $0.45
    based on 122795 samples
Elapsed time is 0.254632 seconds.
```

2. (10 marks)
Consider the *unscrambled* Sobol' sequence, $\{\boldsymbol{x}_i\}_{i=1}^{\infty}$, a kind of multi-dimensional quasi-Monte Carlo or low discrepancy sampling set, where $d$ is its dimension. You may generate the $\boldsymbol{x}_i$ using MATLAB. For any $m = 0, 1, \ldots,$ and $t = 0, \ldots, m$, a $(t, m, d)$-net in base 2 is a set of $2^m$ points having the property that every interval of the form

$$\left[a_1 2^{-k_1}, (a_1+1)2^{-k_1}\right) \times \cdots \times \left[a_d 2^{-k_d}, (a_d+1)2^{-k_d}\right), \qquad k_1 + \cdots + k_d = m - t,$$

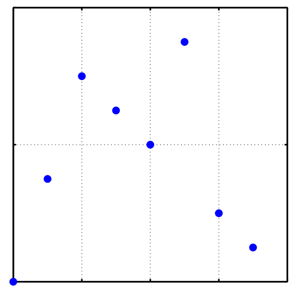
contains exactly $2^t$ points, where the $k_j = 0, \ldots, m$ and the $a_j = 0, \ldots, 2^{k_j} - 1$. Show that the first $2^3 = 8$ points of the Sobol' sequence comprise a $(0, 3, 2)$-net, but not a $(0, 3, 3)$-net, i.e., $t = 0$ if $d = 2$, but $t > 0$ if $d = 3$.

*Answer: The code belows plots the first eight points of a Sobol' net. In the program below we use both a graphical method and a numerical methods. For a two-dimensional Sobol' net, the plots below show that all of the boxes described above for $t = 0$ contain exactly one point.*



*Thus, the Sobol' net is a $(0, 3, 2)$-net.*

*However, when we plot the second and third coordinates of the Sobol' points as below, each* *box with $t = 0$ does not contain exactly one point:*



*Some contain two points and some contain none. For example, the box*

$$[0, 1) \times [1/4, 1/2) \times [0, 1/2) \quad (k_1 = 0, k_2 = 2, k_3 = 1, a_1 = 0, a_2 = 1, a_3 = 0)$$

*contains no points*

```
%% Problem 2
tic
d=3; %largest dimension
p=sobolset(d); %initialize Sobol set
m=3; %log_2 of number of points
n=2^m; %number of points
x=net(p,n); %get points
disp('The first eight points of a three-dimensional Sobol net are:')
disp(num2str(x,'%6.3f'))
%A numerical way
%For dimension 2, t=0
ttry=0; %check this value of t
dtry=2; %use this value of d
isnet=true; %initialize flag
for k1=0:m
    k2=m-ttry-k1; %value of k2 given m and t
    for a1=0:(2^k1-1);
        whin1=(x(:,1)>=a1*2^-k1)&(x(:,1)<(a1+1)*2^-k1);
        for a2=0:(2^k2-1);
           whin2=(x(:,2)>=a2*2^-k2)&(x(:,2)<(a2+1)*2^-k2);
           numin=sum(whin1&whin2);
           if numin ~= 2^ttry; isnet=false; keyboard; break; end
        end
        if ~isnet; break; end
    end
    if ~isnet; break; end
end
if isnet;
    disp(['IS a (' int2str(ttry) ',' int2str(m) ',' int2str(dtry) ') net'])
else
    disp(['IS NOT a (' int2str(ttry) ',' int2str(m) ',' int2str(dtry) ') net'])
end
%For dimension 3, t=0
```

```
ttry=0;
dtry=3;
isnet=true;
for k1=0:m
    for a1=0:(2^k1-1);
        whin1=(x(:,1)>=a1*2^-k1)&(x(:,1)<(a1+1)*2^-k1);
        for k2=0:m-ttry-k1
            k3=m-ttry-k1-k2;
            for a2=0:(2^k2-1);
               whin2=(x(:,2)>=a2*2^-k2)&(x(:,2)<(a2+1)*2^-k2);
                for a3=0:(2^k3-1);
                   whin3=(x(:,3)>=a3*2^-k3)&(x(:,3)<(a3+1)*2^-k3);
                   numin=sum(whin1&whin2&whin3);
                   if numin ~= 2^ttry; isnet=false; break; end
                end
               if ~isnet; break; end
            end
            if ~isnet; break; end
        end
        if ~isnet; break; end
    end
    if ~isnet; break; end
end
if isnet;
    disp(['IS a (' int2str(ttry) ',' int2str(m) ',' int2str(dtry) ') net'])
else
    disp(['IS NOT a (' int2str(ttry) ',' int2str(m) ',' int2str(dtry) ') net'])
end
%For dimension 3, t=1
ttry=1;
dtry=3;
isnet=true;
for k1=0:m
    for a1=0:(2^k1-1);
        whin1=(x(:,1)>=a1*2^-k1)&(x(:,1)<(a1+1)*2^-k1);
        for k2=0:m-ttry-k1
            k3=m-ttry-k1-k2;
            for a2=0:(2^k2-1);
               whin2=(x(:,2)>=a2*2^-k2)&(x(:,2)<(a2+1)*2^-k2);
                for a3=0:(2^k3-1);
                   whin3=(x(:,3)>=a3*2^-k3)&(x(:,3)<(a3+1)*2^-k3);
                   numin=sum(whin1&whin2&whin3);
                   if numin ~= 2^ttry; isnet=false; break; end
                end
               if ~isnet; break; end
            end
            if ~isnet; break; end
        end
        if ~isnet; break; end
```

```
    end
    if ~isnet; break; end
end
if isnet;
    disp(['IS a (' int2str(ttry) ',' int2str(m) ',' int2str(dtry) ') net'])
else
    disp(['IS NOT a (' int2str(ttry) ',' int2str(m) ',' int2str(dtry) ') net'])
end

%A graphical way
labelletters='abcd';
hx=1/16;
hy=2;
for j=1:4
    hx=2*hx;
    hy=hy/2;
    figure;
    plot(x(:,1),x(:,2),'b.','markersize',30)
    set(gca,'xtick',0:hx:1,'ytick',0:hy:1,...
        'xticklabel',' ','yticklabel',' ','linewidth',2)
    grid on
    axis('square')
    eval(['print -depsc Sobol12' labelletters(j) '.eps'])
end
toc
disp(' ')

The first eight points of a three-dimensional Sobol net are:
0.000 0.000 0.000
0.500 0.500 0.500
0.250 0.750 0.250
0.750 0.250 0.750
0.125 0.625 0.875
0.625 0.125 0.375
0.375 0.375 0.625
0.875 0.875 0.125
IS a (0,3,2) net
IS NOT a (0,3,3) net
IS a (1,3,3) net
Elapsed time is 1.928783 seconds.
```

3. (10 marks)
Consider a stock price model with a variable interest rate where the interest rate may be adjusted after half a year. The initial interest rate is $R_1 = 0.03$, and the interest rate six months later is $R_2 = R_1 + X$, where

$$\text{Prob}(X = x) = \begin{cases} 0.25, & x = -0.01 \\ 0.5, & x = 0, \\ 0.25, & x = 0.01. \end{cases}$$

The price of a stock with an initial price of \$100 and a volatility of 30% is

$$S(1/2) = 100\exp((R_1 - 0.045)(1/2) + 0.3\sqrt{1/2}Y_1)$$
$$S(1) = S(1/2)\exp((R_2 - 0.045)(1/2) + 0.3\sqrt{1/2}Y_2)$$

where $Y_1$ and $Y_2$ are IID $\mathcal{N}(0,1)$, and $X$ is independent of the $Y_j$. Use a 3-dimensional Sobol' sequence to generate $2^{16}$ stock price paths and estimate the price of a European call option with strike price \$100. Consider carefully the discount factor that you apply to the payoff.

```
tic
d=3;
p=scramble(sobolset(d),'MatousekAffineOwen');
n=2^16;
xsob=net(p,n);
X=zeros(n,1);
X(xsob(:,1)<=0.25)=-0.01;
X(xsob(:,1)>0.75)=0.01;
%X=0.01*floor(2*xsob(:,1)-0.5); %alternative way
R1=0.03;
R2=R1+X;
Y=norminv(xsob(:,2:3));
S0=100;
sig=0.3;
Shalf=S0*exp((R1-sig^2/2)*(1/2)+sqrt(1/2)*sig*Y(:,1));
S1=Shalf.*exp((R2-sig^2/2)*(1/2)+sqrt(1/2)*sig*Y(:,2));
K=100;
payoff=max(S1-K,0).*exp(-R1*(1/2)-R2*(1/2));
price=mean(payoff);
disp(['The estimated price of this option = $' ...
    num2str(price,'%4.2f')])
toc
disp(' ')
```

```
The estimated price of this option = $13.28
Elapsed time is 0.092957 seconds.
```