% UOCallMC.m

function [C,CI,NCrossed] = UOCallMC(S0,X,r,T,sigma,Sb,NSteps,NRepl)

% Generate asset paths

Payoff = zeros(NRepl,1);

NCrossed = 0;

Path=AssetPaths1(S0,r,sigma,T,NSteps,1);

for i=1:NRepl

Path=AssetPaths1(S0,r,sigma,T,NSteps,1);

crossed = any(Path >= Sb+40);

if crossed == 0

Payoff(i) = max(0, Path(NSteps+1)-X);

else

Payoff(i) = 0;

NCrossed = NCrossed + 1;

end

end

[C,aux,CI] = normfit( exp(-r\*T) \* Payoff);

UOCallMC =

3.6145

CI1 =

3.5936

3.6353

NCrossed1 =

12170

% UOCallMCCond.m

function [Cuo,CI,NCrossed] = UOCallMCCond(S0,X,r,T,sigma,Sb,NSteps,NRepl)

dt = T/NSteps;

[Call,Put] = blsprice(S0,X,r,T,sigma);

% Generate asset paths and payoffs for the down and in option

NCrossed = 0;

Payoff = zeros(NRepl,1);

Times = zeros(NRepl,1);

StockVals = zeros(NRepl,1);

for i=1:NRepl

Path=AssetPaths1(S0,r,sigma,T,NSteps,1);

tcrossed = min(find( Path <= Sb ));

if not(isempty(tcrossed))

NCrossed = NCrossed + 1;

Times(NCrossed) = (tcrossed-1) \* dt;

StockVals(NCrossed) = Path(tcrossed);

end

end

if (NCrossed > 0)

[Caux, Paux] = blsprice(StockVals(1:NCrossed),X,r,T-Times(1:NCrossed),sigma);

Payoff(1:NCrossed) = exp(-r\*Times(1:NCrossed)) .\* Caux;

end

[Cuo, aux, CI] = normfit(Call - Payoff);

>> CompUOCallMCCond

UOCallMCCond =

2.7984

CI2 =

2.7925

2.8043

NCrossed2 =

149680

>>