Ayush Sharma 11 January 2018

(150123046)

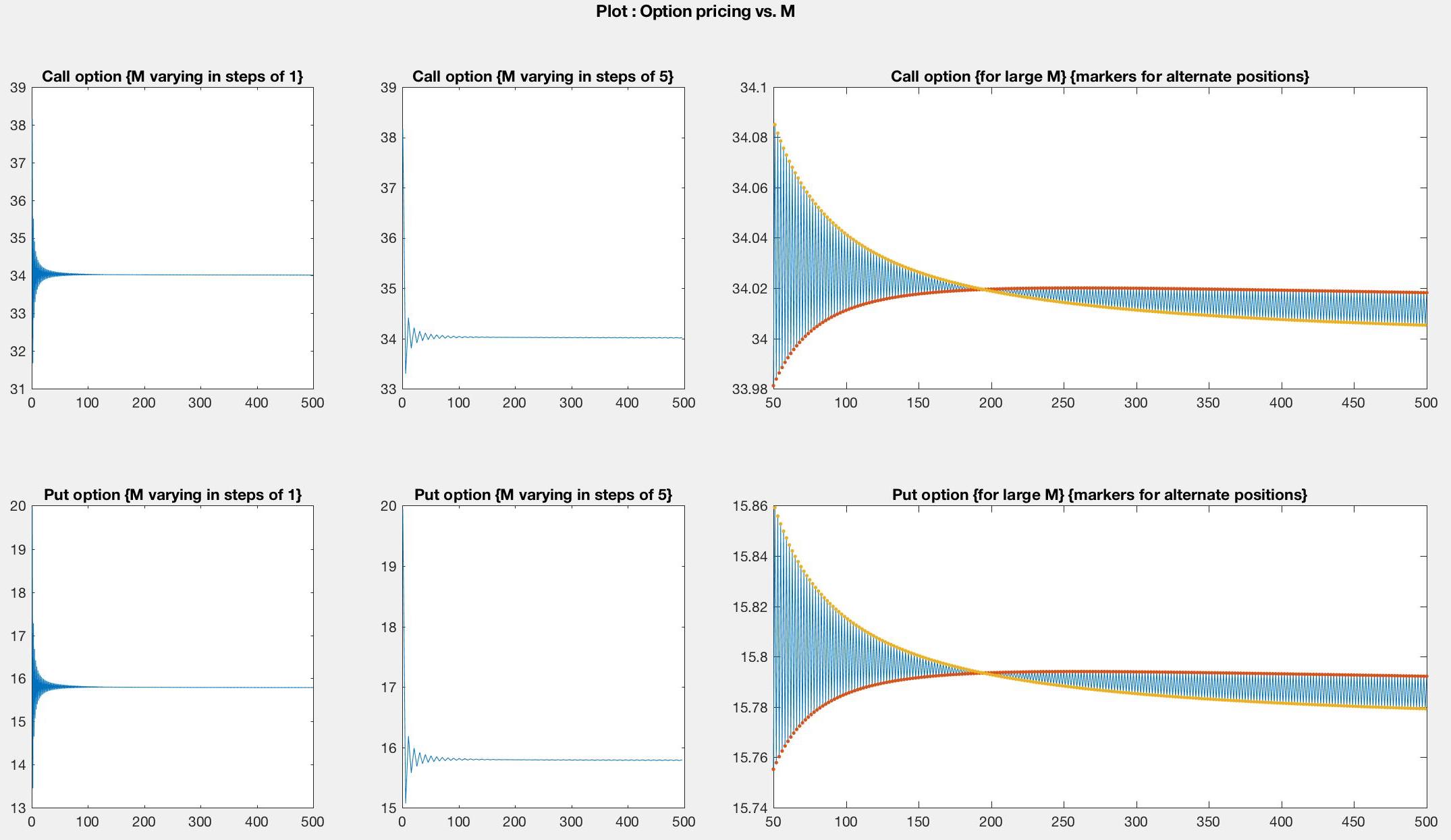
Question 1.

Initial Option Prices : {vs. M being the number of subintervals in the time interval [0, T]}.

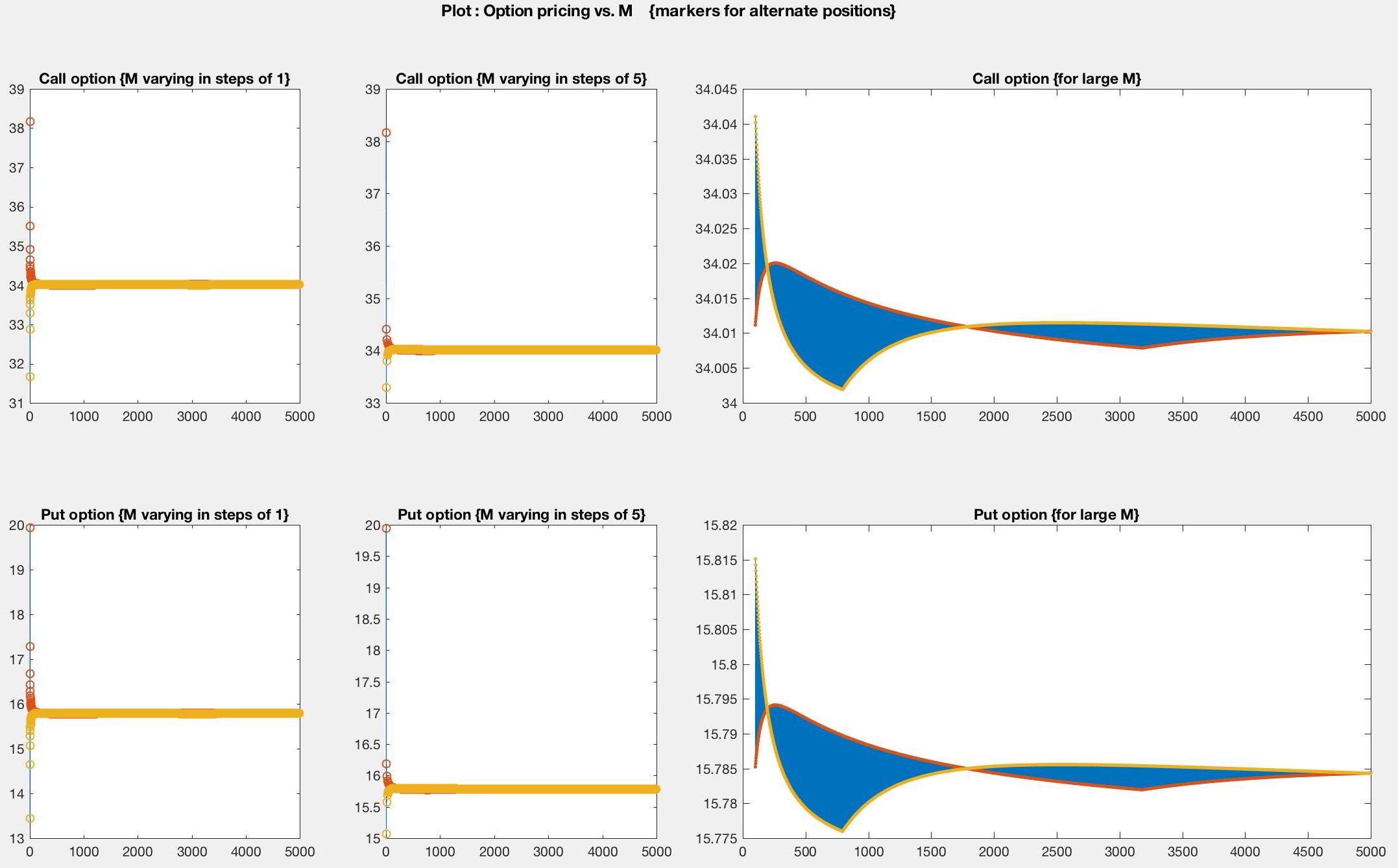
| M | **1** | **5** | **10** | **20** | **50** | **100** | **200** | **400** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Call Option | 38.1676350252277 | 34.9065325113806 | 33.6250217531477 | 33.8594494884939 | 33.9811843657195 | 34.0111609847907 | 34.0195787044072 | 34.0191317690076 |
| Put Option | 19.9417172477252 | 16.6806147338781 | 15.3991039756452 | 15.6335317109913 | 15.7552665882175 | 15.7852432072873 | 15.7936609269040 | 15.7932139915073 |

Question 2.

Graphs of the initial option prices vs. varying M in steps of 1 and in steps of 5.



- when max(M) = 500.



- when max(M) = 5000.

The options’ prices converge theoretically, but, practically the approximations while computing accumulate and generate very small errors for large M (the number of subintervals in the time interval [0, T]).

Moreover, the prices appear/seem to follow a particular sequence at each of the alternate positions (i.e. at odd and even positions/values of M).

Hence, it can be concluded that the sequence of the options’ prices appears to converge, or at least their consecutive differences seem to diminish with increasing M. Although the pattern of the consecutive differences makes one ponder whether forces other than the errors of computational approximations are at play.

Question 3.

Table of values of the **call options** at t = 0, 0.50, 1, 1.50, 3, 4.5 for the case M = 20.

| T | **0** | **0.5** | **1** | **1.5** | **3** | **4.5** |
| --- | --- | --- | --- | --- | --- | --- |
|  | 33.8594494884939 | 59.9587689009226 | 100.662665713361 | 160.611387753017 | 519.099688850719 | 1419.42451210004 |
|  |  | 31.8932532222464 | 57.6999946871752 | 98.4388692488003 | 359.934183790789 | 1024.99337281541 |
|  |  | 15.0958725138798 | 29.8039551213269 | 55.2953556785672 | 242.030182820014 | 732.791598029106 |
|  |  |  | 13.4697162427970 | 27.5732042363838 | 154.841699053598 | 516.323199151877 |
|  |  |  | 5.15483112999247 | 11.7674969625988 | 91.1934332962958 | 355.959465061829 |
|  |  |  |  | 4.12140462102741 | 46.9761877848510 | 237.159088911363 |
|  |  |  |  | 1.12500321452092 | 19.7252062201026 | 149.149605635256 |
|  |  |  |  |  | 6.14852046342652 | 83.9505768315326 |
|  |  |  |  |  | 1.23597113385790 | 36.2514944912452 |
|  |  |  |  |  | 0.118330144851688 | 8.14917387261671 |
|  |  |  |  |  | 0 | 0 |
|  |  |  |  |  | 0 | 0 |
|  |  |  |  |  | 0 | 0 |
|  |  |  |  |  |  | 0 |
|  |  |  |  |  |  | 0 |
|  |  |  |  |  |  | 0 |
|  |  |  |  |  |  | 0 |
|  |  |  |  |  |  | 0 |
|  |  |  |  |  |  | 0 |

Table of values of the **put options** at t = 0, 0.50, 1, 1.50, 3, 4.5 for the case M = 20.

| T | **0** | **0.5** | **1** | **1.5** | **3** | **4.5** |
| --- | --- | --- | --- | --- | --- | --- |
|  | 15.6335317109913 | 8.47920422853985 | 3.50417389797198 | 0.942426524411339 | 0 | 0 |
|  |  | 15.4871434314014 | 8.00422345974075 | 2.99824974526608 | 0 | 0 |
|  |  | 24.6728171615361 | 15.2694321085748 | 7.43626200913783 | 0.00870528162829204 | 0 |
|  |  |  | 24.9832865693941 | 14.9633718726971 | 0.172102756885190 | 0 |
|  |  |  | 35.9653036163975 | 25.2709596397774 | 1.23570223423873 | 0 |
|  |  |  |  | 36.9700720665165 | 4.95818558292699 | 0 |
|  |  |  |  | 48.3049508351933 | 13.2218286523064 | 0 |
|  |  |  |  |  | 25.9550239252640 | 0 |
|  |  |  |  |  | 40.5333138464162 | 0.601546168262695 |
|  |  |  |  |  | 53.8548417107225 | 8.28121121914701 |
|  |  |  |  |  | 64.4333109439046 | 26.6399843026775 |
|  |  |  |  |  | 72.3576948261289 | 46.2775544006558 |
|  |  |  |  |  | 78.2282227937572 | 60.8254241391526 |
|  |  |  |  |  |  | 71.6027511135352 |
|  |  |  |  |  |  | 79.5867913064024 |
|  |  |  |  |  |  | 85.5015137559336 |
|  |  |  |  |  |  | 89.8832479168215 |
|  |  |  |  |  |  | 93.1293164213908 |
|  |  |  |  |  |  | 95.5340631151567 |

Code (MATLAb)

*###Function for “binomial pricing algorithm”*

function [ AssetPrice, OptionValue, Time ] = binopt( S0, K, r, T, M, vol, Flag ) %Flag = 1 for Call; 0 for Put

%BINOPT Summary of this function goes here

% Detailed explanation goes here

dt = T/M;

Time = (0:dt:T);

u = exp(vol\*sqrt(dt) + (r-((vol^2)/2))\*dt);

d = exp(-vol\*sqrt(dt) + (r-((vol^2)/2))\*dt);

%Continuous Compounding so "exp(r\*dt)".

if (~(d < exp(r\*dt)) || ~(exp(r\*dt) < u))

msgID = 'MYFUN:ArbitargePossible';

msg = '"d < exp(r\*dt) < u" not true.';

baseException = MException(msgID,msg);

throw(baseException)

end

AssetPrice = zeros(M+1, M+1);

OptionValue = zeros(M+1, M+1);

AssetPrice(1,1) = S0;

for i=2:(M+1)

AssetPrice(1, i) = AssetPrice(1, (i-1))\*u;

AssetPrice(2:i, i) = AssetPrice(1:(i-1), (i-1))\*d;

end

%Flag = 1 for a call option, or Flag = 0 for a put option.

if (Flag == 1)

OptionValue(:, M+1) = max((AssetPrice(:, M+1) - K), 0);

else

OptionValue(:, M+1) = max((K - AssetPrice(:, M+1)), 0);

end

%Continuous Compounding so "exp(r\*dt)".

p\_ = (exp(r\*dt) - d)/(u-d);

q\_ = (u - exp(r\*dt))/(u-d);

for i = M:-1:1

OptionValue(1:i, i) = (p\_\*OptionValue(1:i, i+1) + q\_\*OptionValue(2:(i+1), i+1))/exp(r\*dt);

end

end

*### Script for question 1.*

clear;clc;

S0 = 100;

K = 105;

T = 5;

r = 0.05;

vol = 0.3;

M=[1, 5, 10, 20, 50, 100, 200, 400];

Callopt = 1:length(M); Putopt = 1:length(M);

for i=1:length(M)

[ ~, OptionValue, ~ ] = binopt( S0, K, r, T, M(i), vol, 1 );

Callopt(i) = OptionValue(1,1);

[ ~, OptionValue, ~ ] = binopt( S0, K, r, T, M(i), vol, 0 );

Putopt(i) = OptionValue(1,1);

end

*### Script for Question 2.*

clear;clc;

S0 = 100;

K = 105;

T = 5;

r = 0.05;

vol = 0.3;

M = 5000;

Callopt = 1:M; Putopt = 1:M;

for i=1:M

[ ~, OptionValue, ~ ] = binopt( S0, K, r, T, i, vol, 1 );

Callopt(i) = OptionValue(1,1);

[ ~, OptionValue, ~ ] = binopt( S0, K, r, T, i, vol, 0 );

Putopt(i) = OptionValue(1,1);

end

ques2plot( M, Callopt, Putopt );

% save(“ques2workspace");

*###Function for Plotting Initial option prices {question 2}.*

function [ ] = ques2plot( M, Callopt, Putopt )

%QUES2PLOT Summary of this function goes here

% Detailed explanation goes here

%F = figure('Color','white', 'pos',[10 10 900 600]);

F = figure('Color','white');

set(gcf, 'Units', 'Normalized', 'OuterPosition', [0, 0.04, 1, 0.96]);

p = uipanel('Parent',F,'BorderType','none');

p.Title = ['Plot : Option pricing vs. M {markers for alternate positions}'];

p.TitlePosition = 'centertop';

p.FontSize = 12;

p.FontWeight = 'bold';

subplot(2,4,1, 'Parent',p);

plot(1:M, Callopt);

hold on;

scatter(1:2:M, Callopt(1:2:M), 'o');

scatter(2:2:M, Callopt(2:2:M), 'o');

hold off;

title("Call option \{M varying in steps of 1\}");

subplot(2,4,2, 'Parent',p);

plot(1:5:M, Callopt(1:5:M));

hold on;

scatter(1:10:M, Callopt(1:10:M), 'o');

scatter(6:10:M, Callopt(6:10:M), 'o');

hold off;

title("Call option \{M varying in steps of 5\}");

subplot(2,4,3:4, 'Parent',p);

plot(floor(M\*.02):M, Callopt(floor(M\*.02):M));

hold on;

scatter(floor(M\*.02):2:M, Callopt(floor(M\*.02):2:M), '.');

scatter(floor(M\*.02)+1:2:M, Callopt(floor(M\*.02)+1:2:M), '.');

hold off;

title("Call option \{for large M\}");

subplot(2,4,5, 'Parent',p);

plot(1:M, Putopt);

hold on;

scatter(1:2:M, Putopt(1:2:M), 'o');

scatter(2:2:M, Putopt(2:2:M), 'o');

hold off;

title("Put option \{M varying in steps of 1\}");

subplot(2,4,6, 'Parent',p);

plot(1:5:M, Putopt(1:5:M));

hold on;

scatter(1:10:M, Putopt(1:10:M), 'o');

scatter(6:10:M, Putopt(6:10:M), 'o');

hold off;

title("Put option \{M varying in steps of 5\}");

subplot(2,4,7:8, 'Parent',p);

plot(floor(M\*.02):M, Putopt(floor(M\*.02):M));

hold on;

scatter(floor(M\*.02):2:M, Putopt(floor(M\*.02):2:M), '.');

scatter(floor(M\*.02)+1:2:M, Putopt(floor(M\*.02)+1:2:M), '.');

hold off;

title("Put option \{for large M\}");

saveas(F,'2.jpg', 'jpg');

end

*### Script for question 3.*

clear;clc;

S0 = 100;

K = 105;

T = 5;

r = 0.05;

vol = 0.3;

M = 20;

t = [0, 0.50, 1, 1.50, 3, 4.5];

idx = (t/(T/M)) + 1;

[ ~, CallOptionValue, Time ] = binopt( S0, K, r, T, M, vol, 1 );

[ ~, PutOptionValue, Time ] = binopt( S0, K, r, T, M, vol, 0 );

for i=1:length(t)

disp(['T =', num2str(t(i))]);

disp("Call option value:");

disp(CallOptionValue(1:idx(i), idx(i)));

disp("Put option value:");

disp(PutOptionValue(1:idx(i), idx(i)));

end