Ayush Sharma
(150123046)

11 January 2018

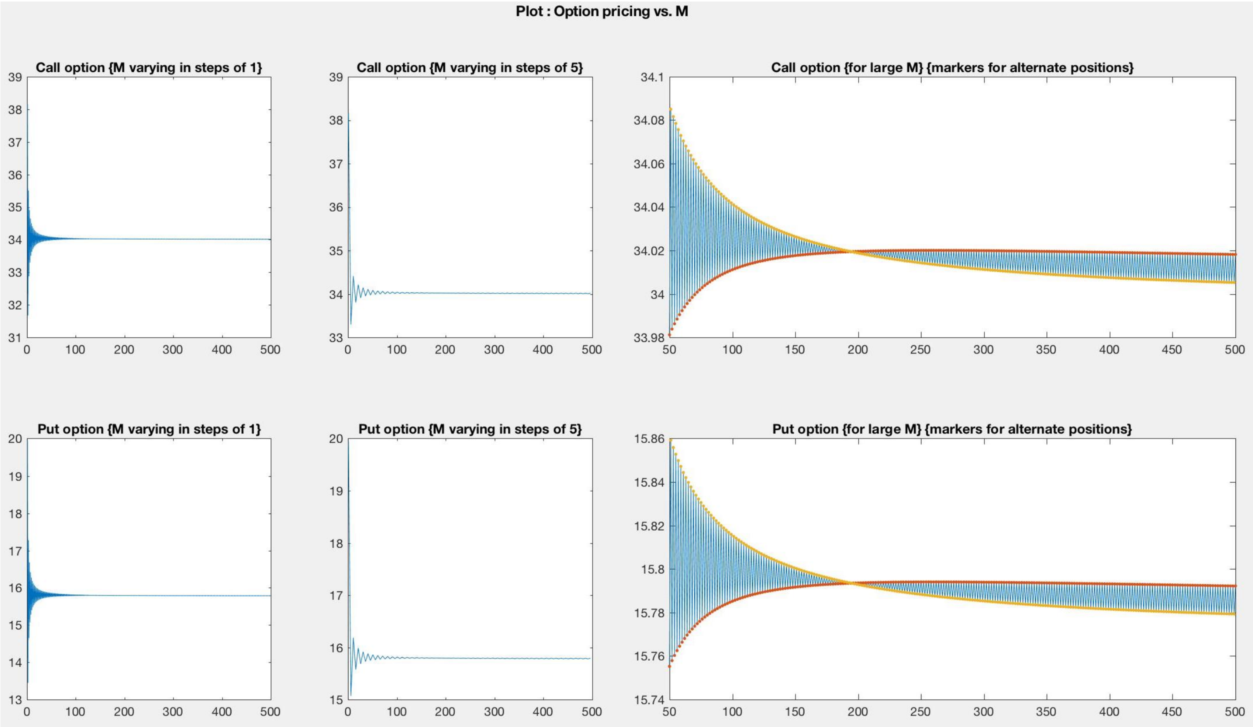
QUESTION 1.

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

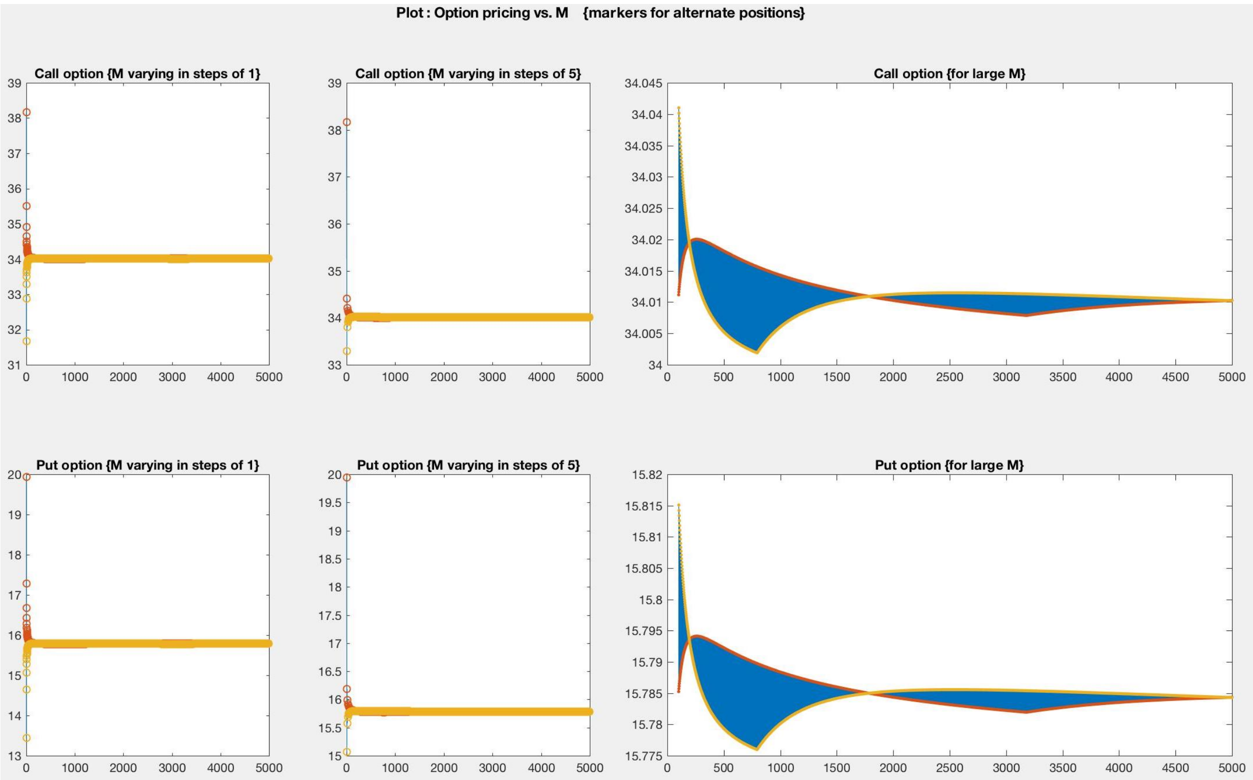
M	1	5	10	20	50	100	200	400
Call Option	38.167635	34.906532	33.625021	33.859449	33.981184	34.011160	34.019578	34.019131
Put Option	19.941717	16.680614	15.399103	15.633531	15.755266	15.785243	15.793660	15.793213

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
	33.859449488	59.958768900	100.66266571	160.61138775	519.09968885	1419.4245121
		31.893253222	57.699994687	98.438869248	359.93418379	1024.9933728
		15.095872513	29.803955121	55.295355678	242.03018282	732.79159802
			13.469716242	27.573204236	154.84169905	516.32319915
			5.1548311299	11.767496962	91.193433296	355.95946506
				4.1214046210	46.976187784	237.15908891
				1.1250032145	19.725206220	149.14960563
					6.1485204634	83.950576831
					1.2359711338	36.251494491
					0.118330144 851688	8.1491738726
					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
	15.63353171C	8.4792042285	3.5041738979	0.942426524 411339	0	0
		15.487143431	8.0042234597	2.9982497452	0	0
		24.672817161	15.269432108	7.4362620091	0.008705281 62829204	0
			24.983286569	14.963371872	0.172102756 885190	0
			35.965303616	25.270959639	1.2357022342	0
				36.970072066	4.9581855829	0
				48.304950835	13.221828652	0
					25.955023925	0
					40.533313846	0.601546168 262695
					53.85484171C	8.2812112191
					64.433310943	26.639984302
					72.357694826	46.277554400
					78.228222793	60.825424139
						71.602751113
						79.586791306
						85.501513755
						89.883247916
						93.129316421
						95.534063115

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
```