Analysis about Gap Option

Fuhuan Zhu

Xi Jia

Xianxin Wang

Yidi Wang

1. Introduction

Gap Option is an option that its trigger price K_2 only determines whether the option can be exercised and the strike determines the value of the option. For a Gap Call Option, the option exercises when stock price S is greater than the trigger price K_2, and it pays stock price minus strike K_1.

The price of a Gap Option is linearly related to its strike price. This means if we know the prices of two Gap Options that are identical besides their strike prices, we can use linear combination of these two Gap Options to derive the price of a new Gap Option with a different strike price.

The advantage of Gap Option is that the payout can depend not only on the strike price, and it is not tied to the exercise price thus allow buyer of the Gap Option to hedge risk against great downfall without bother the investor with minor movement of the underlying asset. Since Gap Option can offer protection against rapid downside or upside moves, it will be very useful for companies and individuals who wants to hedge their risk in extreme cases with a cheaper price. From an insurance point of view, one can see this as the assurance that one will be 100% covered if the loss exceeds a certain amount, and only expose the risk of normal market fluctuations. The potential buyers can also be speculators who believes the market will experience a great down fall or up rise. In real life, the protection buyers are usually the hedge funds who wants to make extra money out of the market downfall. Like all the other exotic options, Gap Options are traded in over the counter market. Theoretically, every term of Gap Option contracts is negotiable, but this will greatly reduce liquidity of the option. In order to increase liquidity, standardization of the terms of Gap Option is common like the strike price, trigger price, maturity and the number of shares that each contract represents are very common.

The following part of the paper will proceed as follows: part two is an analysis about European Gap Option. In this part we will provides explicit formulas of Gap Option pricing and its price sensitivity. We will also discuss the hedging strategy with real market data; In part three, we will provide analysis for American Gap Option including the pricing formula and a discussion about early exercise of American Gap Option.

2.1 Option Price

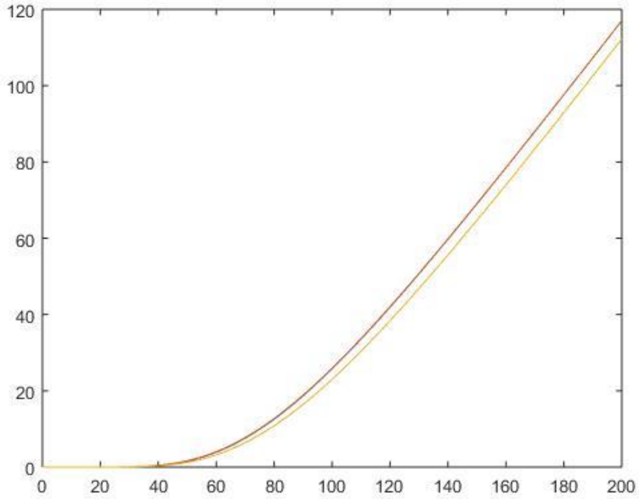
According to Black-Scholes Model, the price of a European gap call option would be:

$$c = S_t e^{-\delta\tau} N(d_1) - K_1 e^{-r\tau} N(d_2) \qquad (2.1)$$

$$d_1 = \frac{ln\left({}^{S_t}/_{K_2}\right) + \left(r - \delta + {}^{\sigma^2}/_2\right)\tau}{\sigma\sqrt{\tau}} \qquad d_2 = d_1 - \sigma\sqrt{\tau} \qquad \tau = T - t$$

Where N (·) is the standard normal cumulative distribution, and we will denote ∅ (·) as the standard normal probability density.

Use $K_1 = 80$, $K_2 = 85$, r=0.5%, dividend rate=3.5%, volatility=0.6, maturity =0.5, we plot the gap call option price with respect to its underlying stock.



The blue line denotes the gap option curve, the orange line denotes the value of a plain vanilla option on strike 80, and the yellow line denotes the value of a plain vanilla option on strike 85. As can see from the plot, the value of the gap option is very close to that of a plain vanilla option with strike 80, whereas more expensive than a plain vanilla option with strike 85.

2.2 Hedging

The hedging strategy for the gap option under Black-Scholes model would be a delta-hedge:

$$\Delta = \frac{\partial C}{\partial S_t} = \frac{\partial}{\partial S_t}\left[S_t e^{-\delta\tau} N(d_1) + K_1 e^{-r\tau} N(d_2)\right] \qquad (2.2)$$

Applying the Chain Rule we find that:

$$\frac{\partial d_1}{\partial S_t} = \frac{\partial d_2}{\partial S_t} = \frac{1}{S_t \sigma\sqrt{\tau}}$$

We could separate the option price in two parts, a plain Vanilla option and a digital option with K=$K_2 - K_1$:

$$c = \left[S_t e^{-\delta\tau} N(d_1) + K_2 e^{-r\tau} N(d_2)\right] + \left[(K_2 - K_1) e^{-r\tau} N(d_2)\right]$$

Since we already know the delta for a plain vanilla call is N(d$_1$), the differentiation comes down to solve for the delta of a digital option:

$$\Delta = \frac{\partial}{\partial S_t}\left[(K_2 - K_1) e^{-r\tau} N(d_2)\right]$$

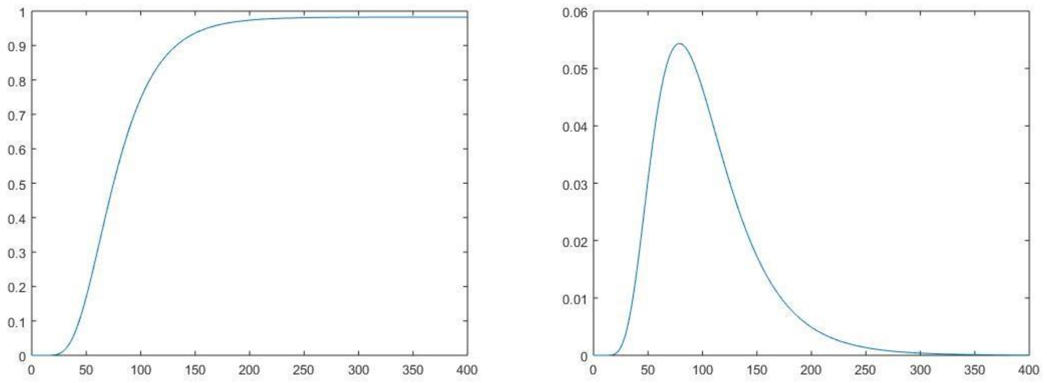
$$= \frac{(K_2 - K_1) e^{-r\tau} N'(d_2)}{S_t \sigma \sqrt{\tau}}$$

Therefore, the solution to (2.2) is:

$$\Delta = e^{-\delta\tau} N(d_1) + \frac{(K_2 - K_1) e^{-r\tau} N'(d_2)}{S_t \sigma \sqrt{\tau}} \qquad (2.3)$$

The hedging strategy would be purchasing Δ shares of underlying stock for each one share of gap call option purchased.

The following plot shows the delta of the gap option (left) and the corresponding digital option (right) with respect to the underlying.



As we discussed above, the hedging strategy for a gap option comes down to hedge a combination of a plain vanilla option and a digital option. Apparently, we could use delta hedge for the plain vanilla option. However, from the plots above, the delta of the corresponding digital option (right plot) increases to highest value around strike price [K1, K2], then reduces to 0. This rapid changing of delta value implies that the delta hedge of a digital option will result in high transaction fee, therefore too expensive in reality.

In reality, we would recommend to use a spread combination as the hedging strategy for the digital option. In particular, we need to match the payoff of the spread to that of a digital option:

$$(K_2 - K_1) * \boldsymbol{I}_{\{S_t \geq K_2\}}$$

Since the payoff of this digital option has a jump at K$_2$, the hedging strategy for the digital option would be a long position of call option with exercise price K3 and a short position of call option with exercise price K4 (K4>K2>K3). The smaller (K4-K3) gets, the more precise our hedging strategy is. That is:

$$\frac{K_2 - K_1}{K_4 - K_3}[C(S_t, K_3) - C(S_t, K_4)] \approx \boldsymbol{E}^*\left[e^{-r\tau}(K_2 - K_1) \times \boldsymbol{I}_{\{S_t \geq K_2\}}\right]$$

In conclusion, by consider the gap option as a combination of a plain vanilla option and a digital option, out hedging strategy in reality constitutes of:

(1) Long position of $e^{-\delta\tau}N(d_1)$ shares of stock

(2) Long position of $\frac{K_2 - K_1}{K_4 - K_3}$ shares of call option with strike K3

(3) Short position of $\frac{K_2 - K_1}{K_4 - K_3}$ shares of call option with strike K4

2.3 Sensitivity

The sensitivity of the European gap call option with respect to different parameters are as follows:

$$\Delta = \frac{\partial C}{\partial S_t} = e^{-\delta\tau}N(d_1) + \frac{(K_2 - K_1)e^{-\delta\tau}N'(d_1)}{\sigma\sqrt{\tau}K_2}$$

$$\Gamma = \frac{\partial \Delta}{\partial S_t} = \frac{e^{-\delta\tau}N'(d_1)}{S_t\sigma\sqrt{\tau}}\left[1 - \frac{(K_2 - K_1)d_1}{\sigma\sqrt{\tau}K_2}\right]$$

$$\upsilon = \frac{\partial C}{\partial \sigma} = \sqrt{\tau}S_t e^{-\delta\tau}N'(d_1)\left[1 - \frac{(K_2 - K_1)d_1}{\sigma\sqrt{\tau}K_2}\right]$$

$$\rho = \frac{\partial C}{\partial r} = \tau K_1 e^{-r\tau}N(d_2) + S_t e^{-\delta\tau}N'(d_1)\frac{\sqrt{\tau}(K_2 - K_1)}{\sigma K_2}$$

$$\theta = \frac{\partial C}{\partial t} = \delta S_t e^{-\delta\tau}N(d_1) - rK_1 e^{-r\tau}N(d_2) + \frac{\sigma S_t e^{-\delta\tau}N'(d_1)}{2\sqrt{\tau}}$$

$$\times\left[\frac{ln\left(\frac{S_t}{K_2}\right) - \left(r - \delta - \frac{\sigma^2}{2}\right)\tau}{\sigma^2\tau} \times \frac{K_2 - K_1}{K_2} - 1\right]$$

2.4 Monte Carlo simulations of European Gap Call Option price of XOM.

For the Monte Carlo simulation, we conduct maximum of 100000 iterations (N) for each run and we observe improving convergence as N increased.

As for parameters, we assume initial stock price of XOM = \$85.75; Strike price K_1 = \$80; Strike price K_2 = \$85; Maturity = 0.5; Dividend rate = 3.5%; Interest rate = 0.5%; Volatility =60%.

Analytically The theoretical price of European call option using Black-Scholes model is \$15.9851.

The European call option price based on Monte Carlo Simulation is as follows:

	Number of iterations	Time	Price
Run 1	N=1000	1.948 s	16.0238
Run 2	N=1000	1.829 s	15.4794
Run 3	N=1000	1.813 s	16.0356
Run 4	N=1000	1.895 s	16.1515

	Number of iterations	Time	Price
Run 1	N=10000	185.160 s	16.1814
Run 2	N=10000	182.464 s	15.9247
Run 3	N=10000	200.406 s	15.8696
Run 4	N=10000	191.320 s	15.9058

	Number of iterations	Time	Price
Run 1	N=100000	17064.812 s	16.0142
Run 2	N=100000	19421.458 s	15.9982
Run 3	N=100000	19729.808 s	15.9243
Run 4	N=100000	19468.089 s	16.0488

(The CPU running time is based on Intel G3900. The variation of the running time might be subject to other background tasks.)

2.5 Hedging gap option in Reality

The payoff of the European gap option can be replicated by a portfolio constructed with a vanilla European call option and a cash-or-nothing option as discussed in the part 2.1. Therefore, we can take opposite position in such portfolio to hedge the risk for taking position in gap option. One main concern is the transaction fee for buying the portfolio. Another concern is that the cash-or-nothing option is mainly traded in OTC market and not as liquate as the plain vanilla option market, thus sometimes not available to form a suitable portfolio.

2.6 Implied volatility of XOM based on plain vanilla option prices.

Firstly, for each call or put price, we take the average of the best bid and ask prices in the market. And we use this price to obtain forward price of the XOM using put-call-parity (Therefore we get rid of the data that misses bid or ask price in the market).

$$\mathrm{F} = \mathrm{Strike} + \mathrm{e}^{\mathrm{rT}}(C - P)$$

Then we use Black’s Formula and plugin forward price to get the implied volatility.

$$\mathrm{C} = \mathrm{e}^{-\mathrm{rT}}[F \cdot N(d_1) - K \cdot N(d_2)]$$

$$\mathrm{P} = \mathrm{e}^{-\mathrm{rT}}[K \cdot N(-d_2) - F \cdot N(-d_1)$$

$$d_1 = \frac{\ln\left(\frac{F}{K}\right) + \left(\frac{\sigma^2}{2}\right)T}{\sigma\sqrt{T}}$$

$$d_2 = d_1 - \sigma\sqrt{T}$$

Using this method, we try to acquire Implied Volatility for $T_1 = 2$ days, $T_2 = 152$ days, T_3= 429 days.

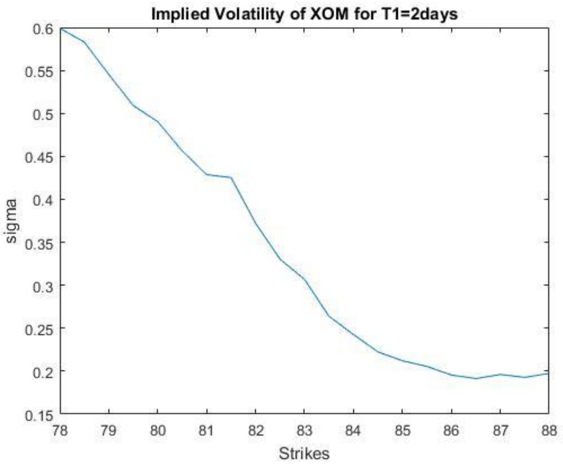
For the interest rates, we use the data from Federal Reserve as of Nov. 1st, 2016 and rates are:

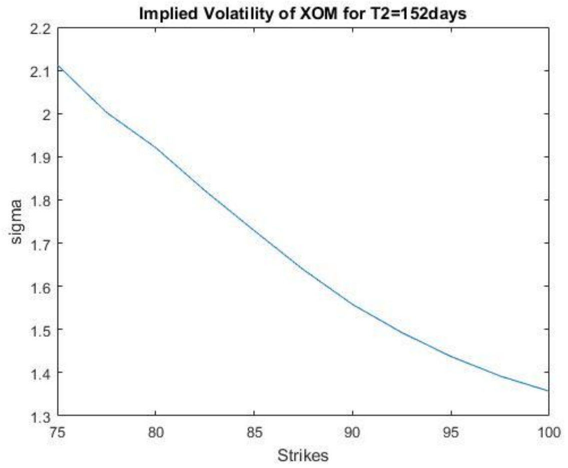
		3-Months		6-Months	1-Year		2-Year
Interest rate		0.35%		0.5%	0.65%		0.83%

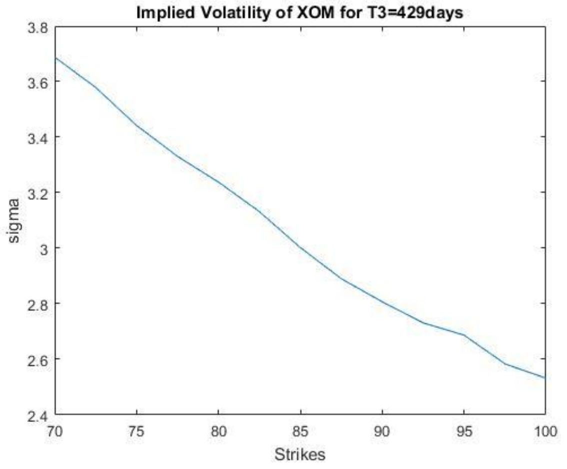
sing linear interpolation, we get the interest rates for each maturity:

	2 days	3-Months	***152 days***	6-Months	1-Year	***429 days***	2-Year
Interest rate	***0.35%***	0.35%	***0.453%***	0.5%	0.65%	***0.6845%***	0.83%

Finally, we obtain Implied Volatility for the 3 difference maturities:







2.7 Recover the implied risk-neutral distribution for some T from market vanilla option prices.

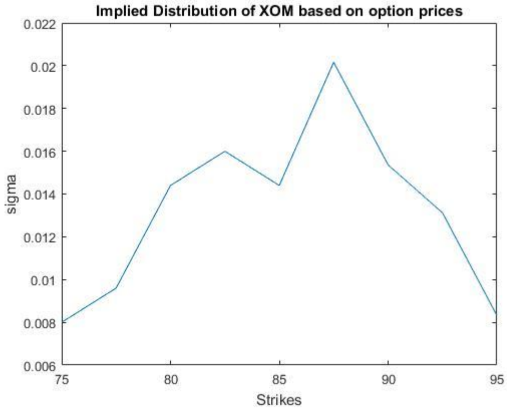
Under Risk-Neutral measure **Q**, we can derive that:

$$\mathrm{C(K,T)} = \mathrm{E}^{\mathrm{Q}}[e^{-rT}(S_T - K)^+] = e^{-rT}\int_K^{+\infty}(x-K)f(x)dx$$

Where f(x) is the distribution of $\mathrm{S_T}$ under measure **Q.** Then we obtain:

$$\mathrm{f(K)} = \mathrm{e}^{\mathrm{rT}}\frac{\left(\partial^2 C(K,T)\right)}{\partial K^2}, \qquad K > 0$$

Numerically, if we choose T = 152 days, we obtain the distribution as follows:



(We observe that the distribution curve is not very smooth around $\mathrm{K} = \mathrm{S_0} = \$85.75$ due to the limitation of data but it still remains representative.)

3.1 Formulating the Problem

The pricing problem for American gap option can be described as to find the price of American styled gap option which has the payoff function as $C_t^a(S_t,t) = (S_t - K_1)I((S_t \geq K_2)$ subject to the free boundary conditions that

$$\begin{cases} \partial C_t + (r-\delta) * S * \partial C_s + \frac{\sigma^2}{2} * \partial C_{ss} = r * C & for\ \ S < B(t) \\ C(S,t) = S - K\ _1 \qquad for\ \ S \geq B(t) \\ C(B(t)-,t) = B(t) - K_1\ \ \forall t \in [0,T) \\ \partial C_s(B(t)-,t) = 1\ \ \forall t \in [0,T) \\ \mathrm{C(0+,t)} = 0 \qquad \forall t \in [0,T) \end{cases}$$

3.2 Optimal Exercise Region

The optimal exercise region can be described as

$$\begin{cases} \varepsilon = \{(S,t)\epsilon[0,+\infty) \times [0,T]: C_t^a(S_t,t) = (S_t - K_1)\} \\ B(t) = inf\{S > 0: (S,t) \in \varepsilon\} \end{cases}$$

3.3 The early Exercise premium and integral equation

The Early Exercise premium representation can be described as follows:

$$\pi = E_t\left[\int_t^T \xi_{t,u} I(\tau_u = u)\,(r_u y_u du - df_u - dA_u^y)\right]$$

For gap option, $y_t = S_t - K_1, df_u = 0, dA_u^y = d(S_u - K_1) = (r - \ \delta)S_u du$

Therefore, we have that early exercise premium for the gap option

$$\begin{aligned}\pi(St, t, B(*)) &= E_t\left[\int_t^T I(\tau_u = u)\left(r_u y_u du - df_u - dA_u^y\right)\right]\\ &= E_t\left[\int_t^T e^{-r(u-t)} I(S_u \geq B(u))\,(\delta \mathrm{S}_u \ \ - rK_1)du\right]\\ &= \int_t^T e^{-r(u-t)}(\delta E_t[\mathrm{S}_u \ \ I(S_u \geq B(u))] - rK_1 E_t[I(S_u \geq B)])du\end{aligned}$$

$$= \int_t^T [e^{-\delta(u-t)}\delta S_t N\big(d(S_t, B(u), u-t)\big) - e^{-r(u-t)} r K_1 N(d(S_t, B(u), u-t) - \sigma\sqrt{u-t})]du$$

where $d(S_t, B(u), u-t) = \frac{log \ \ St/B(u) \ \ +(r-\delta+\sigma\text{^}2/2)(u-t)}{\sigma\sqrt{u-t}}$ and immediate exercise boundary B solves the recursive non-linear integral equation $B(t) - K_1 = C^e(B_t, t) + \pi(B(t), t, B(*))$ for$t\epsilon[0, T)$

Subject to the boundary condition $B_{T-} = \max\{K_2, (rK_1/\delta)\}$

Therefore, the integral equation for the immediate exercise boundary for the gap option can be expressed as:

$$\begin{cases} B(t) - K_1 = C^e(B(t), t) + \pi\big(B(t), t, B(*)\big) \ \text{ for } \ t\epsilon[0, \mathrm{T}] \\ \qquad\qquad B_{T-} = \max\{K_2, (rK_1/\delta)\} \end{cases}$$

for t=u

$$\begin{aligned}B(u) - K_1 = B(u)e^{-\delta(u-t)}N(d1) - K_1 e^{-r(u-t)} N\big(d1 - \sigma\sqrt{u-t}\big)\\ + \int_t^T [e^{-\delta(u-t)}\delta S_t N\big(d(S_t, B(u), u-t)\big) - e^{-r(u-t)} r K_1 N(d(S_t, B(u), u-t)\\ - \sigma\sqrt{u-t})]du\end{aligned}$$

where $\mathrm{d1} = \frac{\log\left(\frac{B(u)}{K_2}\right) + (r-\delta+\sigma\text{^}2/2)(u-t)}{\sigma\sqrt{u-t}}$

3.4 Numerical Solution

From the lecture notes and textbook, "American Style Derivatives：Valuation and Computation", we know that the integration can be solved numerically by the following algorithm:

Step 1: Discretize the time horizon [0, T]: $0 = \mathrm{t}_0 < t_1 < t_2 < \cdots < t_n = T, \qquad h = T/n$

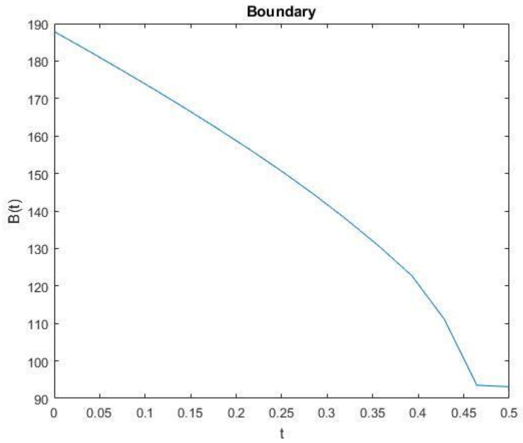
Step 2：Approximate π by the scheme

$\int_t^T \phi\ (B(tn-1), B(tn), tn - tn - 1) = \sum_{i=j+1}^{n} h * \phi(B(tj), B(ti), ti - tj)$

Step 3: Given $B(t_n) = B(T)$, we can find the $B(t_{n-1})$ as the root of the algebraic equation by newton Raphson method.

$$\begin{aligned}B(t_{n-1}) - K_1 &= C^e(B(t_{n-1}), t_{n-1}) + \pi\big(B(t), t, B(*)\big)\\ &= C^e(B(t_{n-1}), t_{n-1}) + h \times \phi(B(t_{n-1}), B(t_n), t_n - t_{n-1})\end{aligned}$$

Step 4: Continue by induction, we can get the corresponding exercise boundary B with respect to each time interval. Here for simplicity, we use 15 time intervals and solve the nonlinear equations.
We can then plot the trajectory of the $\{B(t_i)\}\ for\ i\epsilon 0,1,2,\ldots,200$ to obtain the immediate exercise boundary for $t\epsilon[0,T)$



After getting the immediate exercise boundary, we can find the corresponding early exercise premium for the initial time t=0 and get the American style gap option price by the equation:

$$C_t^a(S_t,t) = C_t^e(S_t,t) + \sum_{i=0}^{n} h * \phi(B(t_i), B(t_{i+1}), h)$$

Therefore, the American style gap call option price is *\$17.5073* compare to the European gap price *\$15.9851.*

References

1. Detemple, Jérôme. *American-style Derivatives: Valuation and Computation*. Boca Raton: Taylor & Francis, 2006. Print.
2. Tankov, Peter. "*Pricing and Hedging Gap Risk*." SSRN Electronic Journal (n.d.): n. pag. Web.

Appendix

- **All files can be fetched from *https://github.com/wxianxin/Gap-Option***
- **Please modify the directory path of file "*xom_option_chain_20161116.xlsx*" on your own computer before import the data.**

A. Matlab Code for Monti Carlo Simulation for European Gap Option Price

```
% Author: Steven Wang        Date:20161116
% XOM option chain data as of 20161116

%%
% Initialization
clear all

S0=85.75;
St=85.75;
K1=80;
K2=85;
T=0.5;
delta=0.035;
r=0.005;
sigma=0.6;
N=10000;
Delta=T/N;

%%
%BS pricing formula

d_1 = (log((St*exp(1)^(-delta*T))/(K2*exp(1)^(-r*T))) + 0.5*sigma^2*T)/(sigma*sqrt(T));
d_2 = d_1 - sigma*sqrt(T);
GapCall = St*exp(1)^(-delta*T)*normcdf(d_1) - K1*exp(1)^(-r*T)*normcdf(d_2);
GapPut = K1*exp(1)^(-r*T)*normcdf(-d_2) - St*exp(1)^(-delta*T)*normcdf(-d_1);

%%
%Monte Carlo of BS
GapCall_MC_prices = zeros(N,1);
GapPut_MC_prices = zeros(N,1);

for c = 1:10*N
    St = S0;
    for n = 1:N
        dS_t = (r-delta)*St*Delta + sigma*St*randn*sqrt(Delta);
        St = St + dS_t;
    end
    if (St > K2) && (St > K1)
        GapCall_MC_prices(c,1) = St - K1;
    end
end

GapCall_MC = exp(1)^(-r*T)*mean(GapCall_MC_prices)
```

B. Matlab Code for Implied Volatility

- **Please run different sections separately regarding to the different maturities.**
- **Please comment or uncomment the code line of parameter definition of "r" and the code line of "title" under the plot function.**

```
% Author: Steven Wang        Date:20161116
% XOM option chain data as of 20161116


%% Import data from spreadsheet
% T1
clear all
% Script for importing data from the following spreadsheet:
%
% Workbook: L:\Backup\Onedrive\Dropbox\AcademyIII\MF770\Project\xom_option_chain_20161116.xlsx
%    Worksheet: Sheet1

[~, ~, raw0_0] = xlsread('D:\Onedrive\Dropbox\AcademyIII\MF770\Project\xom_option_chain_20161116.xlsx','Sheet1','A23:A43');
[~, ~, raw0_1] = xlsread('D:\Onedrive\Dropbox\AcademyIII\MF770\Project\xom_option_chain_20161116.xlsx','Sheet1','C23:D43');
[~, ~, raw0_2] = xlsread('D:\Onedrive\Dropbox\AcademyIII\MF770\Project\xom_option_chain_20161116.xlsx','Sheet1','H23:H43');
[~, ~, raw0_3] = xlsread('D:\Onedrive\Dropbox\AcademyIII\MF770\Project\xom_option_chain_20161116.xlsx','Sheet1','J23:K43');
raw = [raw0_0,raw0_1,raw0_2,raw0_3];

data = reshape([raw{:}],size(raw));

Strike1 = data(:,1);
Bid1 = data(:,2);
Ask1 = data(:,3);
```

```
Strike2 = data(:,4);
Bid2 = data(:,5);
Ask2 = data(:,6);


clearvars data raw raw0_0 raw0_1 raw0_2 raw0_3;

%% Import data from spreadsheet
% T2
clear all
% Script for importing data from the following spreadsheet:
%
% Workbook:
D:\Onedrive\Dropbox\AcademyIII\MF770\Project\xom_option_chain_20161116.xlsx
%    Worksheet: Sheet1

[~, ~, raw0_0] =
xlsread('D:\Onedrive\Dropbox\AcademyIII\MF770\Project\xom_option_chain_20161116.xlsx','She
et1','A134:A144');
[~, ~, raw0_1] =
xlsread('D:\Onedrive\Dropbox\AcademyIII\MF770\Project\xom_option_chain_20161116.xlsx','She
et1','C134:D144');
[~, ~, raw0_2] =
xlsread('D:\Onedrive\Dropbox\AcademyIII\MF770\Project\xom_option_chain_20161116.xlsx','She
et1','H134:H144');
[~, ~, raw0_3] =
xlsread('D:\Onedrive\Dropbox\AcademyIII\MF770\Project\xom_option_chain_20161116.xlsx','She
et1','J134:K144');
raw = [raw0_0,raw0_1,raw0_2,raw0_3];

data = reshape([raw{:}],size(raw));

Strike1 = data(:,1);
Bid1 = data(:,2);
Ask1 = data(:,3);
Strike2 = data(:,4);
Bid2 = data(:,5);
Ask2 = data(:,6);

clearvars data raw raw0_0 raw0_1 raw0_2 raw0_3;

%% Import data from spreadsheet
% T3
clear all
```

```
% Script for importing data from the following spreadsheet:
%
% Workbook:
D:\Onedrive\Dropbox\AcademyIII\MF770\Project\xom_option_chain_20161116.xlsx
%    Worksheet: Sheet1

[~, ~, raw0_0] =
xlsread('D:\Onedrive\Dropbox\AcademyIII\MF770\Project\xom_option_chain_20161116.xlsx','She
et1','A188:A200');
[~, ~, raw0_1] =
xlsread('D:\Onedrive\Dropbox\AcademyIII\MF770\Project\xom_option_chain_20161116.xlsx','She
et1','C188:D200');
[~, ~, raw0_2] =
xlsread('D:\Onedrive\Dropbox\AcademyIII\MF770\Project\xom_option_chain_20161116.xlsx','She
et1','H188:H200');
[~, ~, raw0_3] =
xlsread('D:\Onedrive\Dropbox\AcademyIII\MF770\Project\xom_option_chain_20161116.xlsx','She
et1','J188:K200');
raw = [raw0_0,raw0_1,raw0_2,raw0_3];

data = reshape([raw{:}],size(raw));

Strike1 = data(:,1);
Bid1 = data(:,2);
Ask1 = data(:,3);
Strike2 = data(:,4);
Bid2 = data(:,5);
Ask2 = data(:,6);

clearvars data raw raw0_0 raw0_1 raw0_2 raw0_3;

%% Initialization
call_mid_price = 0.5*(Ask1 + Bid1);
put_mid_price = 0.5*(Ask2 + Bid2);
K=Strike1;
T=2/365;
delta=0.035;
% r=0.0035;
% r=0.00453;
  r=0.006845;
S_t = 85.75;
sigmas=zeros(length(Strike1),1);

%% Implied volatility
```

```
%Using put call parity we can calculate the implied forward price from the European call and put
prices closest to at-the-money
F = K + exp(1)^(r*T)*(call_mid_price - put_mid_price);

for i=1:length(Strike1)
    syms sigma
    d_1 = (log(F(i)/K(i))+0.5*sigma^2*T)/(sigma*sqrt(T));
    d_2 = d_1 - sigma*sqrt(T);
    eqn = call_mid_price(i) == exp(1)^(-r*T)*(F(i)*normcdf(d_1) - K(i)*normcdf(d_2));
    sigmas(i) = solve(eqn, sigma);


end

% for i=1:length(Strike)
%     syms sigma
%     d_1 = (log(S_t/K(i))+(r-delta+0.5*sigma^2)*T)/(sigma*sqrt(T));
%     d_2 = d_1 - sigma*sqrt(T);
%     eqn = call_mid_price(i) == exp(1)^(-delta*T)*S_t*normcdf(d_1) - exp(1)^(-
r*T)*K(i)*normcdf(d_2);
%     sigmas(i) = solve(eqn, sigma)
% end

%% Plot
plot(K,sigmas)
xlabel('Strikes')
ylabel('sigma')
% title('Implied Volatility of XOM for T1=2days')
% title('Implied Volatility of XOM for T2=152days')
  title('Implied Volatility of XOM for T3=429days')
```

C. Matlab Code for Implied Distribution

```
or: Steven Wang      Date:20161120
% XOM option chain data as of 20161116

%% Import data from spreadsheet
% T2
clear all
% Script for importing data from the following spreadsheet:
%
%                                                                    Workbook:
L:\Backup\Onedrive\Dropbox\AcademyIII\MF770\Project\xom_option_chain_20161116.xlsx
%    Worksheet: Sheet1
```

```
[~, ~, raw0_0] =
xlsread('D:\Onedrive\Dropbox\AcademyIII\MF770\Project\xom_option_chain_20161116.xlsx','She
et1','A23:A43');
[~, ~, raw0_1] =
xlsread('D:\Onedrive\Dropbox\AcademyIII\MF770\Project\xom_option_chain_20161116.xlsx','She
et1','C23:D43');
[~, ~, raw0_2] =
xlsread('D:\Onedrive\Dropbox\AcademyIII\MF770\Project\xom_option_chain_20161116.xlsx','She
et1','H23:H43');
[~, ~, raw0_3] =
xlsread('D:\Onedrive\Dropbox\AcademyIII\MF770\Project\xom_option_chain_20161116.xlsx','She
et1','J23:K43');
raw = [raw0_0,raw0_1,raw0_2,raw0_3];

data = reshape([raw{:}],size(raw));

Strike1 = data(:,1);
Bid1 = data(:,2);
Ask1 = data(:,3);
Strike2 = data(:,4);
Bid2 = data(:,5);
Ask2 = data(:,6);


clearvars data raw raw0_0 raw0_1 raw0_2 raw0_3;

%% Import data from spreadsheet
% Script for importing data from the following spreadsheet:
%
% Workbook:
D:\Onedrive\Dropbox\AcademyIII\MF770\Project\xom_option_chain_20161116.xlsx
%    Worksheet: Sheet1

[~, ~, raw0_0] =
xlsread('D:\Onedrive\Dropbox\AcademyIII\MF770\Project\xom_option_chain_20161116.xlsx','She
et1','A134:A144');
[~, ~, raw0_1] =
xlsread('D:\Onedrive\Dropbox\AcademyIII\MF770\Project\xom_option_chain_20161116.xlsx','She
et1','C134:D144');
[~, ~, raw0_2] =
xlsread('D:\Onedrive\Dropbox\AcademyIII\MF770\Project\xom_option_chain_20161116.xlsx','She
et1','H134:H144');
[~, ~, raw0_3] =
xlsread('D:\Onedrive\Dropbox\AcademyIII\MF770\Project\xom_option_chain_20161116.xlsx','Sheet1','J134:K144');
raw = [raw0_0,raw0_1,raw0_2,raw0_3];

data = reshape([raw{:}],size(raw));

Strike1 = data(:,1);
Bid1 = data(:,2);
Ask1 = data(:,3);
Strike2 = data(:,4);
Bid2 = data(:,5);
Ask2 = data(:,6);

clearvars data raw raw0_0 raw0_1 raw0_2 raw0_3;

%% Initialization
call_mid_price = 0.5*(Ask1 + Bid1);
put_mid_price = 0.5*(Ask2 + Bid2);
K=Strike1;
T=2/365;
delta=0.035;
r=0.005;
S_t = 85.75;
sigmas=zeros(21,1);

%% Implied Distribution
derivative = @(f,x,i) (f(i+1) - f(i))/(x(i+1) - x(i));
first_derivatives = zeros(length(Strike1)-1,1);
second_derivatives = zeros(length(Strike1)-2,1);

for i = 1:length(Strike1)-1
    first_derivatives(i,1) = derivative(call_mid_price,K,i);
end

for i = 1:length(Strike1)-2
    second_derivatives(i) = (first_derivatives(i+1) - first_derivatives(i))/((K(i+1)-K(i))^2);
end


%% Plot
plot(K(1:length(Strike1)-2),second_derivatives)
xlabel('Strikes')
ylabel('sigma')
title('Implied Distribution of XOM based on option prices')
```

D. Matlab Code for immediate exercise boundary

```
% Author: Steven Wang        Date: 20161208
% XOM option chain data as of 20161116

%% Initialization

clear all

sigma=0.6;
r=0.005;
delta=0.035;
T=0.5;
n=15;
h=T/n;
K1=80;
K2=85;
X0=85;
S_t = 85.75;

%% Boundary

X = zeros(n,1);

% Initialize the first values

syms X
d1 = (log(X/K2) + (r - delta + 0.5*sigma^2)*h)/(sigma*sqrt(h));
d2 = (log(X/X0) + (r - delta + 0.5*sigma^2)*h)/(sigma*sqrt(h));
eqn = X - K1 == X*exp(-delta*h)*normcdf(d1) - K1*exp(-r*h)*normcdf(d1 - sigma*sqrt(h)) + h*(delta*X*exp(-delta*h)*normcdf(d2) - exp(-r*h)*r*K1*normcdf(d2-sigma*sqrt(h)));
Y(n) = double(solve(eqn, X));

% loop
t = linspace(0,T,n);
summation = 0;
summ = zeros(n-1,1);
for i=1:(n-1)
    j = n-i;
    syms X
    d1 = (log(X/K2) + (r - delta + 0.5*sigma^2)*(T-t(j)))/(sigma*sqrt(T-t(j)));
    d2 = (log(X/Y(j+1)) + (r - delta + 0.5*sigma^2)*h)/(sigma*sqrt(h));
```

```
        pi = h*(delta*exp(-delta*h)*X*normcdf(d2) - exp(-r*h)*r*K1*normcdf(d2-sigma*sqrt(h)));
        summation = summation + pi;


        eqn = X - K1 == X*exp(-delta*(T-t(j)))*normcdf(d1) - K1*exp(-r*(T-t(j)))*normcdf(d1 - sigma*sqrt(T-t(j))) + summation;
        Y(j) = double(solve(eqn, X));

        pi = h*(delta*exp(-delta*h)*Y(j)*normcdf((log(Y(j)/Y(j+1)) + (r - delta + 0.5*sigma^2)*h)/(sigma*sqrt(h))) - exp(-r*h)*r*K1*normcdf((log(Y(j)/Y(j+1)) + (r - delta + 0.5*sigma^2)*h)/(sigma*sqrt(h))-sigma*sqrt(h)));
        summ(i) = pi;

  end

  plot(t,Y)
  xlabel('t')
  ylabel('B(t)')
  title('Boundary')


  %% Option Prices
    d1 = (log(S_t/K2) + (r - delta + 0.5*sigma^2)*(T-t(1)))/(sigma*sqrt(T-t(1)));
        d2 = d1-sigma*sqrt(T-t(1));
        C = S_t*exp(-delta*(T-t(1)))*normcdf(d1)-K1*exp(-r*(T-t(1)))*normcdf(d1-sigma*sqrt(T-t(1))) + sum(summ);
```