Finite Difference Method in Option Pricing

Xiang Meng
13 Mar 2017

Goals for Today

1. Introduction of the model
2. Finite difference method
    a. Explicit scheme
    b. Implicit scheme
3. Empirical results
4. Conclusion

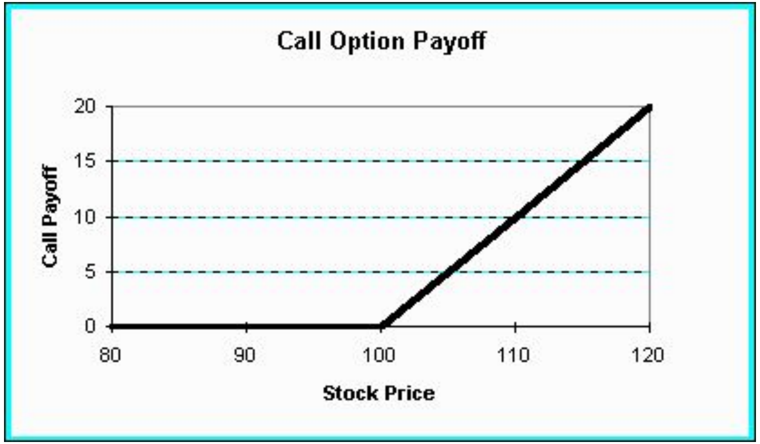
Introduction of the Model

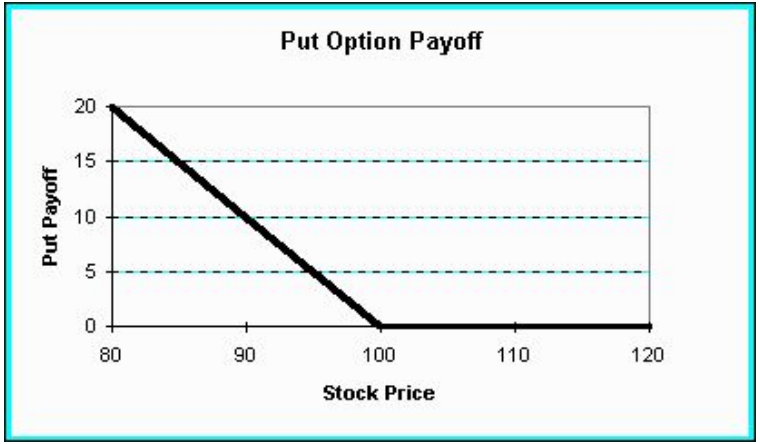
What is an option?

Call (put) option is a contract that gives its holder the right to buy (sell) a specific quantity of a specified financial asset (underlying asset) at a specified price (strike/exercise price) on or before a specified date (expiration/maturity date).

$$V_T = \begin{cases} (S_T - X)^+, & \text{for call;} \\ (X - S_T)^+, & \text{for put.} \end{cases}$$

Problem: How much does these guys worth NOW?





Black-Scholes PDE (1973)

In Black-Scholes world:

- Underlier price follows a Geometric Brownian Motion.
- Risk free rate and underlier volatility are constant.
- Trading is continuous.
- No transaction costs, no taxes.

$$\frac{\partial v}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 v}{\partial S^2} + (r-q)S\frac{\partial v}{\partial S} - rv = 0,$$

subject to the terminal condition:

$$v(S_T, T) = \begin{cases} (S_T - X)^+, & \text{call;} \\ (X - S_T)^+, & \text{put.} \end{cases}$$

Analytical Solution

Introduce the change-of-variable transformation

$$\tau = T - t$$
$$u = Ce^{r\tau}$$
$$x = \ln\left(\frac{S}{K}\right) + \left(r - \frac{1}{2}\sigma^2\right)\tau$$

Then the Black–Scholes PDE becomes a heat equation

$$\frac{\partial u}{\partial \tau} = \frac{1}{2}\sigma^2 \frac{\partial^2 u}{\partial x^2}$$

Analytical Solution

For European vanilla options, the solution to the Black-Scholes PDE is

$$v(S_t,t) = \begin{cases} e^{-q(T-t)}S_tN(d_1) - Xe^{-r(T-t)}N(d_2), & \text{call;} \\ Xe^{-r(T-t)}N(-d_2) - e^{-q(T-t)}S_tN(-d_1), & \text{put} \end{cases}$$

where

$$d_1 = \frac{\ln(S_t/X) + (r - q + \sigma^2/2)(T-t)}{\sigma\sqrt{T-t}},$$

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

$$N(x) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{x} e^{-y^2/2}\,dy\,.$$

Finite Difference Method

Domain Discretization

$(S_n^i, t_n), \quad i = 0, 1, \ldots, I, \quad n = 0, 1, \ldots, N,$ Where $S_n^i = ih, t_n = n\Delta t$

The domain $S \geq 0$ is truncated to $S \in [0, S_{max}]$

$S_{max} = S_n^I = Ih$ and $T = N\Delta t$

Option values at grid points are $v_n^i = v(S_n^i, t_n)$

(For put option) Boundary conditions are:

$V_N^i = (X - ih)^+$ At $t = T$

$V_N^0 = X e^{-r(T - n\Delta t)}$ At $S = 0$: $V_n^I = 0$ At $S = S_{max}$

Explicit Scheme

PARENTAL
ADVISORY
EXPLICIT CONTENT

Explicit Scheme

Evaluate the PDE at (S^i_{n+1}, t_{n+1})

$$\left.\frac{\partial v}{\partial t}\right|_{(S^i_{n+1},t_{n+1})} + \frac{1}{2}\sigma^2 S^2 \left.\frac{\partial^2 v}{\partial S^2}\right|_{(S^i_{n+1},t_{n+1})} + rS\left.\frac{\partial v}{\partial S}\right|_{(S^i_{n+1},t_{n+1})} - r\,v\Big|_{(S^i_{n+1},t_{n+1})} = 0$$

Use backward time, and centered space

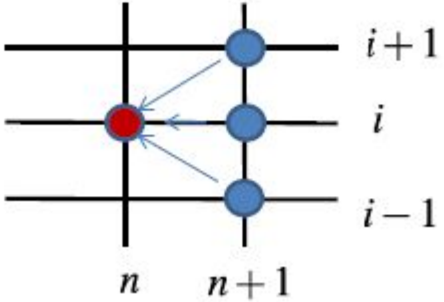
$$\frac{v^i_{n+1} - v^i_n}{\Delta t} + \frac{1}{2}\sigma^2\, i^2\, h^2 \frac{v^{i+1}_{n+1} - 2v^i_{n+1} + v^{i-1}_{n+1}}{h^2} + r\,i\,h \frac{v^{i+1}_{n+1} - v^{i-1}_{n+1}}{2h} - rv^i_{n+1}$$
$$+ O(\Delta t + h^2) = 0.$$

Explicit Scheme

Dropping the truncation error, and re-arranging the terms, we get

$$V_n^i = \frac{1}{2}\left(\sigma^2 i^2 - r i\right)\Delta t V_{n+1}^{i-1} + (1 - \sigma^2 i^2 \Delta t - r\Delta t)V_{n+1}^{i} + \frac{1}{2}\left(\sigma^2 i^2 + r i\right)\Delta t V_{n+1}^{i+1}.$$

The stencil is like:



Stability

We have $V_n^i = a_i V_{n+1}^{i-1} + b_i V_{n+1}^i + c_i V_{n+1}^{i+1}$

A sufficient condition:

$$a_i \geq 0,\ b_i \geq 0,\ c_i \geq 0 \text{ for all } i.$$

Stable?

$$V_n^i = \frac{1}{2}\left(\sigma^2 i^2 - ri\right)\Delta t V_{n+1}^{i-1} + \left(1 - \sigma^2 i^2 \Delta t - r\Delta t\right) V_{n+1}^i + \frac{1}{2}\left(\sigma^2 i^2 + ri\right)\Delta t V_{n+1}^{i+1}$$

$1 - \sigma^2 i^2 \Delta t - r\Delta t$ and $\frac{1}{2}(\sigma^2 i^2 - ri)$ are likely to be negative

Problem!!!

--How?

Approximate $v(S_{n+1}^i, t_{n+1})$ by v_n^i

Explicit Scheme (II)

Evaluate the PDE at (S^i_{n+1}, t_{n+1})

$$\frac{v^i_{n+1} - v^i_n}{\Delta t} + \frac{1}{2}\sigma^2 i^2 h^2 \frac{v^{i+1}_{n+1} - 2v^i_{n+1} + v^{i-1}_{n+1}}{h^2} + rih \frac{v^{i+1}_{n+1} - v^{i-1}_{n+1}}{2h} - rv^i_n + O(\Delta t)$$

$$+O(\Delta t + h^2) = 0$$

Dropping the truncation error, and re-arranging the terms, we get

$$V^i_n = \frac{1}{1 + r\Delta t}\left[\frac{1}{2}\left(\sigma^2 i^2 - ri\right)\Delta t V^{i-1}_{n+1} + (1 - \sigma^2 i^2 \Delta t) V^i_{n+1} + \frac{1}{2}\left(\sigma^2 i^2 + ri\right)\Delta t V^{i+1}_{n+1}\right]$$

Stable?

$$V_n^i = \frac{1}{1+r\Delta t}\left[\frac{1}{2}\left(\sigma^2 i^2 - ri\right)\Delta t V_{n+1}^{i-1} + (1-\sigma^2 i^2 \Delta t)V_{n+1}^i + \frac{1}{2}\left(\sigma^2 i^2 + ri\right)\Delta t V_{n+1}^{i+1}\right]$$

We now requires $1-\sigma^2 i^2 \Delta t > 0$

(Also need to notice the possibility that $\sigma^2 i < r$)

--How?

Consider

$$\left.\frac{\partial v}{\partial S}\right|_{(S_{n+1}^i, t_{n+1})} = \frac{v_{n+1}^{i+1} - v_{n+1}^i}{h} + O(h)$$

Explicit Scheme (III)

Evaluate the PDE at (S^i_{n+1}, t_{n+1})

$$\frac{v^i_{n+1} - v^i_n}{\Delta t} + \frac{1}{2}\sigma^2 i^2 h^2 \frac{v^{i+1}_{n+1} - 2v^i_{n+1} + v^{i-1}_{n+1}}{h^2} + r i h \frac{v^{i+1}_{n+1} - v^i_{n+1}}{h} - r v^i_n + O(\Delta t)$$

$$+ O(\Delta t + h) = 0.$$

Dropping the truncation error, and re-arranging the terms, we get

$$V^i_n = \frac{1}{1 + r\Delta t}\left[\frac{1}{2}\sigma^2 i^2 \Delta t\, V^{i-1}_{n+1} + (1 - \sigma^2 i^2 \Delta t - r i \Delta t) V^i_{n+1} + \frac{1}{2}\left(\sigma^2 i^2 + r i\right) \Delta t\, V^{i+1}_{n+1}\right]$$

Stable? $V_n^i = \frac{1}{1+r\Delta t}\left[\frac{1}{2}\sigma^2 i^2 \Delta t V_{n+1}^{i-1} + (1-\sigma^2 i^2 \Delta t - r i \Delta t) V_{n+1}^i + \frac{1}{2}(\sigma^2 i^2 + ri)\Delta t V_{n+1}^{i+1}\right]$

We now ONLY requires $1-\sigma^2 i^2 \Delta t - r i \Delta t > 0$

Matrix Form

Recall the boundary condition

$$t = T \quad V_N^i = (X - ih)^+ \qquad S = 0\colon \; V_N^0 = X e^{-r(T - n\Delta t)} \qquad S = S_{max} \quad V_n^I = 0$$

$$\begin{bmatrix} V_n^1 \\ V_n^2 \\ \vdots \\ V_n^{I-2} \\ V_n^{I-1} \end{bmatrix} = \begin{bmatrix} b_1 & c_1 & & & \\ a_2 & b_2 & c_2 & & \\ & \ddots & \ddots & & \\ & & a_{I-2} & b_{I-2} & c_{I-2} \\ & & & a_{I-1} & b_{I-1} \end{bmatrix} \begin{bmatrix} V_{n+1}^1 \\ V_{n+1}^2 \\ \vdots \\ V_{n+1}^{I-2} \\ V_{n+1}^{I-1} \end{bmatrix} + \begin{bmatrix} a_1 V_{n+1}^0 \\ 0 \\ \vdots \\ 0 \\ c_{I-1} V_{n+1}^I \end{bmatrix}$$

Matlab Implementation

```
% Explicit Scheme I
% c=(0.5*sig^2*isq+0.5*r*i)*dt;
% b=1-sig^2*isq*dt-r*dt;
% a=(0.5*sig^2*isq-0.5*r*i)*dt;
% Explicit Scheme II
 c=(0.5*sig^2*isq+0.5*r*i)*dt/(1+r*dt);
 b=(1-sig^2*isq*dt)/(1+r*dt);
 a=(0.5*sig^2*isq-0.5*r*i)*dt/(1+r*dt);
% Explicit Scheme III
% c=(0.5*sig^2*isq+r*i)*dt/(1+r*dt);
% b=(1-sig^2*isq*dt-r*i*dt)/(1+r*dt);
% a=(0.5*sig^2*isq)*dt/(1+r*dt);
```

```
for n=N:-1:1
    VGrid(i,n)=a.*VGrid(i-1,n+1)+b.*VGrid(i,n+1)+c.*VGrid(i+1,n+1);
end;
```

Implicit Scheme

Implicit Scheme

Evaluate the PDE at (S_n^i, t_n)

$$\left.\frac{\partial v}{\partial t}\right|_{(S_n^i,t_n)} + \frac{1}{2}\sigma^2 S^2 \left.\frac{\partial^2 v}{\partial S^2}\right|_{(S_n^i,t_n)} + rS\left.\frac{\partial v}{\partial S}\right|_{(S_n^i,t_n)} - r\left.v\right|_{(S_n^i,t_n)} = 0.$$

Use forward time, centered space

$$\frac{v_{n+1}^i - v_n^i}{\Delta t} + \frac{1}{2}\sigma^2 i^2 h^2 \frac{v_n^{i-1} - 2v_n^i + v_n^{i+1}}{h^2} + r\,i\,h\,\frac{v_n^{i+1} - v_n^{i-1}}{2h} - rv_n^i$$

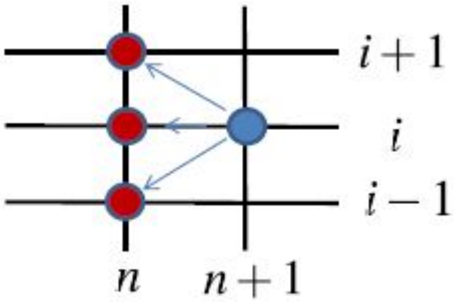
$$+O(\Delta t + h^2) = 0.$$

Implicit Scheme

Dropping the truncation error, and re-arranging the terms, we get

$$V_{n+1}^{i} = -\frac{1}{2}(\sigma^2 i^2 - ri)\Delta t V_n^{i-1} + (1 + \sigma^2 i^2 \Delta t + r\Delta t)V_n^i - \frac{1}{2}(\sigma^2 i^2 + ri)\Delta t V_n^{i+1}$$

The stencil is like:



Stability

We have $a_i V_n^{i-1} + b_i V_n^i + c_i V_n^{i+1} = V_{n+1}^i, \quad i = 1, 2, \ldots, I-1$

In matrix form:

$$\begin{bmatrix} b_1 & c_1 & & & \\ a_2 & b_2 & c_2 & & \\ & \ddots & \ddots & \ddots & \\ & & a_{I-2} & b_{I-2} & c_{I-2} \\ & & & a_{I-1} & b_{I-1} \end{bmatrix} \begin{bmatrix} V_n^1 \\ V_n^2 \\ \vdots \\ V_n^{I-2} \\ V_n^{I-1} \end{bmatrix} = \begin{bmatrix} V_{n+1}^1 \\ V_{n+1}^2 \\ \vdots \\ V_{n+1}^{I-2} \\ V_{n+1}^{I-1} \end{bmatrix} + \begin{bmatrix} -a_1 V_n^0 \\ 0 \\ \vdots \\ 0 \\ -c_{I-1} V_n^I \end{bmatrix}$$

A sufficient condition: " $a_i \leq 0, c_i \leq 0 \quad b_i > |a_i| + |c_i|$ "

Stability

In our scheme:

$$a_i = -\frac{1}{2}(\sigma^2 i^2 - ri)\Delta t V_n^{i-1}$$

We are then motivated to do the change:

$$\left.\frac{\partial V}{\partial S}\right|_{(S_n^i, t_n)} = \frac{V_n^{i+1} - V_n^i}{h} + O(h)$$

Implicit Scheme (II)

$$\frac{v_{n+1}^{i}-v_{n}^{i}}{\Delta t}+\frac{1}{2}\sigma^{2}i^{2}h^{2}\frac{v_{n}^{i-1}-2v_{n}^{i}+v_{n}^{i+1}}{h^{2}}+rih\frac{v_{n}^{i+1}-v_{n}^{i}}{h}-rv_{n}^{i}$$

$$+O(\Delta t+h)=0.$$

Dropping the truncation error, and re-arranging the terms, we get

$$-\frac{1}{2}\sigma^{2}i^{2}\Delta t\,V_{n}^{i-1}+\left(1+\sigma^{2}i^{2}\Delta t+r\Delta t(1+i)\right)V_{n}^{i}-\frac{1}{2}\left(\sigma^{2}i^{2}+ri\right)\Delta t\,V_{n}^{i+1}=V_{n+1}^{i}$$

Matlab Implementation

```
a= -0.5*(sig^2*isq - r*i)*dt;
b=(1 + sig^2*isq*dt + r*dt);
c= -0.5*(sig^2*isq + r*i)*dt;

% Set up sparse tri-diagonal matrix of coefficients
CoeffMatrix=spdiags([c, b, a],-1:1, I-1, I-1)';
ishift=1;
for n=N:-1:1 % backward time recursive
RhsB=VGrid(i+ishift,n+1); % set up right hand side vector
RhsB(1) =RhsB(1) -a(1) *VGrid(0+ishift,n);
RhsB(I-1)=RhsB(I-1)-c(I-1)*VGrid(I+ishift,n);
VGrid(i+ishift,n)=CoeffMatrix\RhsB; % solve linear system
end;
```

No Loop

But need to solve a matrix

Empirical Results

Problem

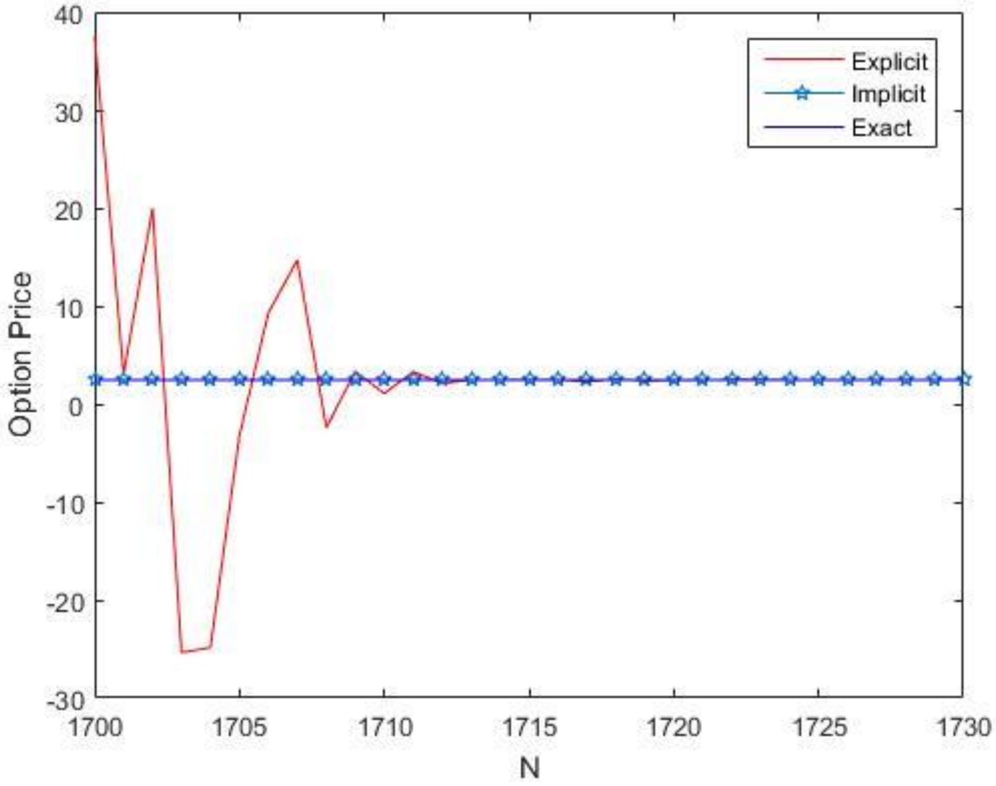
The case: $S_0 = \$100, X = \$98, r = 0.02, T = 0.2, \sigma = 20\%$

$$S_{max} = 3X, \Delta S = 0.5$$

$$\Delta t = T/N, N = 1700, 1701, ..., 1730$$

Why use these N values?

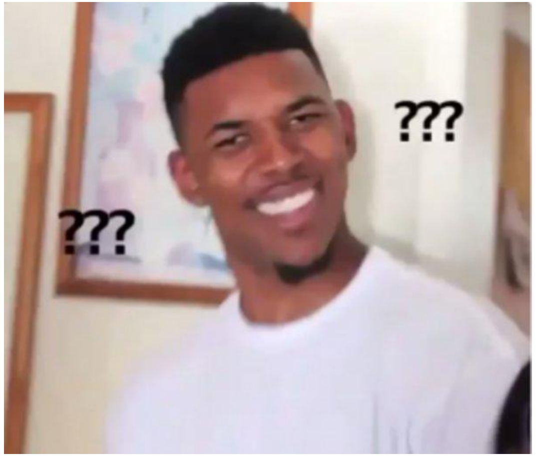
Result



40
30
20
10
0
-10
-20
-30
1700
1705
1710
1715
1720
1725
1730
N
Option Price
Explicit
Implicit
Exact

Stability--Explicit Scheme

```
>> FD_eds_put(100, 98, 0.02, 0.2, 0.2, 1730, 0.5)
Coeff a, Of 587 elements, 0 violated the positivity condition.
Coeff b, Of 587 elements, 122 violated the positivity condition.
Coeff c, Of 587 elements, 0 violated the positivity condition.
At S0=100 exact value=2.457 FD value=2.4691
```



What if $\Delta S = 0.49$?

```
>> FD_eds_put(100, 98, 0.02, 0.2, 0.2, 1730, 0.49)
Coeff a, Of 599 elements, 0 violated the positivity condition.
Coeff b, Of 599 elements, 134 violated the positivity condition.
Coeff c, Of 599 elements, 0 violated the positivity condition.
At S0=100 exact value=2.457 FD value=-1.59523486575866e+18
```

Stability--Explicit Scheme

```
>> FD_eds_put(100, 98, 0.02, 0.2, 0.2, 3000, 0.5)
Coeff a, Of 587 elements, 0 violated the positivity condition.
Coeff b, Of 587 elements, 0 violated the positivity condition.
Coeff c, Of 587 elements, 0 violated the positivity condition.
At S0=100 exact value=2.457 FD value=2.4558
```

```
>> FD_eds_put(100, 98, 0.02, 0.2, 0.2, 3000, 0.49)
Coeff a, Of 599 elements, 0 violated the positivity condition.
Coeff b, Of 599 elements, 0 violated the positivity condition.
Coeff c, Of 599 elements, 0 violated the positivity condition.
At S0=100 exact value=2.457 FD value=2.4709
```

Stability--Implicit Scheme

```
>> FD_ids_put(100, 98, 0.02, 0.2, 0.2, 1730, 0.5)
At S0=100 exact value=2.457 FD value=2.4554
```

```
>> FD_ids_put(100, 98, 0.02, 0.2, 0.2, 1730, 0.49)
At S0=100 exact value=2.457 FD value=2.4705
```

Conclusion

Explicit vs Implicit

Explicit schemes

- ➢ Advantages:
  There is no need to solve a system of equations.
  Easy for implementation in software.
- ➢ Disadvantage: Bad stability and conditional convergence

Implicit schemes

- ➢ Advantage: Good stability and convergence.
- ➢ Disadvantage: Need to solve a system of equations.

Q&A

Thank you!