

## 選擇權價格計算方式

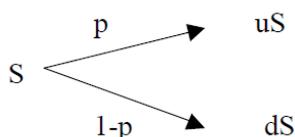
Question: Consider a 5-month Euro. Call option on a non-dividend-paying stock when the stock price is \$50, the strike price is \$50, the risk-free interest rate is 10% per annum, and the volatility is 40% per annum. With our usual notation, this means that  $S_0=50$ ,  $K=50$ ,  $r=0.10$ ,  $\sigma = 0.40$ ,  $T=0.4167$ , and  $q=0$ . Suppose that we divide the life of the option into 100 intervals.

藉由上面的例子，分別來介紹二項樹、三項樹、蒙地卡羅模擬、隱式有限差分法這四種方法的介紹，並且列出這四種方法與 B-S 公式解之值的誤差。

### (1) 二項樹

- 理論<sup>1</sup>

在1979年Cox-Ross-Rubinstein參考Sharpe於1973發展出來的二項式方法，進而推導出二項樹選擇權評價模型，即能彌補其無法計算美式選擇權的缺憾。CRR模型利用無風險投資組合的觀念，以選擇權之標的資產與無風險債券建構一個不論未來股價上漲或下跌皆會等於選擇權價格，其模型推導為假設選擇權存續期間為T，並將期間T分割成n個均等的小區間，在每個小區間中，股價將由原來的股價S上升為uS或下跌為dS。[1]



假設  $p$  為向上變動的機率， $r$  為無風險利率， $\sigma$  為股價變異數，且：

$$R = e^{r\Delta t}, \quad u = e^{\sigma\sqrt{\Delta t}}, \quad d = \frac{1}{u}, \quad p = \frac{R - d}{u - d}$$

此時 CRR 模式下的歐式買權可表示為：

$$C = \left\{ \sum_{i=0}^n \left[ \frac{n!}{i!(n-i)!} \right] p^i (1-p)^{n-i} \times \text{Max} [u^i d^{n-i} S - X, 0] \right\} / R^n$$

- 結論

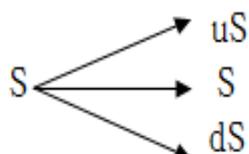
最後透過 Matlab 使用二項樹計算出的歐式買權價格為 6.1038 元  
而利用 B-S model 計算出的歐式買權價格為 6.1151 元  
因此二項樹和 B-S model 計算出的歐式買權價格誤差為  
(6.1038-6.1151)/ 6.1151=-0.0018 (-0.18%)

<sup>1</sup> 許博翔 (2000)。隨機波動性下之障礙選擇權的評價分析。碩士論文，國立中央大學，桃園市。

## (2) 三項樹

### ● 理論<sup>2</sup>

Boyle(1986)提出了三項樹模型，其概念和二項樹模型類似，差異在於二項樹模型假設下一期的股價只有上漲、下跌兩種情況，而三項樹模型則假設了股價的變化有上漲、下跌及持平三種情況。由於股價變化比二項樹多，因此在處理問題時有較大的彈性，比較能處理複雜的產品，且在求解的收斂過程較二項樹快速。[2]



下列是使用了股票變動的分配來分別求出上漲、下跌與不變的機率  
設  $Z = \ln S$

$$\Delta Z \sim N\left[\left(r - q - \frac{\sigma^2}{2}\right)\Delta t, \sigma^2\Delta t\right]$$

$$P_u\Delta Z - P_d\Delta Z = \left(r - q - \frac{\sigma^2}{2}\right)\Delta t \quad \dots\dots\dots(1)$$

$$P_u\Delta Z^2 + P_d\Delta Z^2 = \sigma^2\Delta t \quad \dots\dots\dots(2)$$

$$P_u + P_m + P_d = 1 \quad \dots\dots\dots(3)$$

由(1)(2)(3)可知

$$\begin{bmatrix} \Delta Z & 0 & -\Delta Z \\ \Delta Z^2 & 0 & \Delta Z^2 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} P_u \\ P_m \\ P_d \end{bmatrix} = \begin{bmatrix} \left(r - q - \frac{\sigma^2}{2}\right)\Delta t \\ \sigma^2\Delta t \\ 1 \end{bmatrix}$$

令限制式  $\Delta Z = \sigma\sqrt{3\Delta t}$

$$\text{得 } P_u = \frac{1}{6} + \sqrt{\frac{\Delta t}{12\sigma^2}}\left(r - q - \frac{\sigma^2}{2}\right)$$

$$P_d = \frac{1}{6} - \sqrt{\frac{\Delta t}{12\sigma^2}}\left(r - q - \frac{\sigma^2}{2}\right)$$

$$P_m = 1 - P_u - P_d = \frac{2}{3}$$

### ● 結論

最後透過 Matlab 使用三項樹計算出的歐式買權價格為 6.3638 元  
而利用 B-S model 計算出的歐式買權價格為 6.1151 元  
因此三項樹和 B-S model 計算出的歐式買權價格誤差為  
(6.3638-6.1151)/ 6.1151=0.0407 (4.07%)

<sup>2</sup> Hull, J. C. (2009). *Options, Futures, and Other Derivative Securities*( 7th ed.). Prentice Hall.

### (3) 蒙地卡羅模擬法

● 理論<sup>3</sup>

蒙地卡羅模擬是根據衍生性商品標的資產價格變動之隨機過程進行模擬，並進而求算出此一衍生性商品的價值。假設選擇權的價值（C）不僅受到到期日資產價格（ $S_T$ ）的影響，也受到從發行日至到期日之間資產價格的路徑（ $S_0, \dots, S_T$ ）所影響，即  $C = f(S_0, \dots, S_T)$ ，其中  $f(\cdot)$  為新奇選擇權的收益函數。若短期無風險利率為常數，即  $r(\tau) = r$ ，則此一選擇權的現值為  $C_0 = e^{-rT} \hat{E}[f(S_0, \dots, S_T)]$ ，其中  $\hat{E}$  為風險中立下之期望值，蒙地卡羅模擬法即利用此方程式來進行評價。更明確的說，若資產價格（S）服從幾何布朗運動，則資產價格的瞬間變動為：<sup>[3]</sup>

$$\frac{dS}{S} = \mu dt + \sigma dW$$

$\ln S$  在一段時間內（如從時點  $t$  至  $T$ ）的變動，服從常態分配：

$$\ln S_T - \ln S_t \sim N\left[\left(\mu - \frac{\sigma^2}{2}\right)(T-t), \sigma\sqrt{T-t}\right]$$

在風險中立評價法下（Risk-Neutral Valuation），我們以無風險利率  $r$  代替  $\mu$ ，則

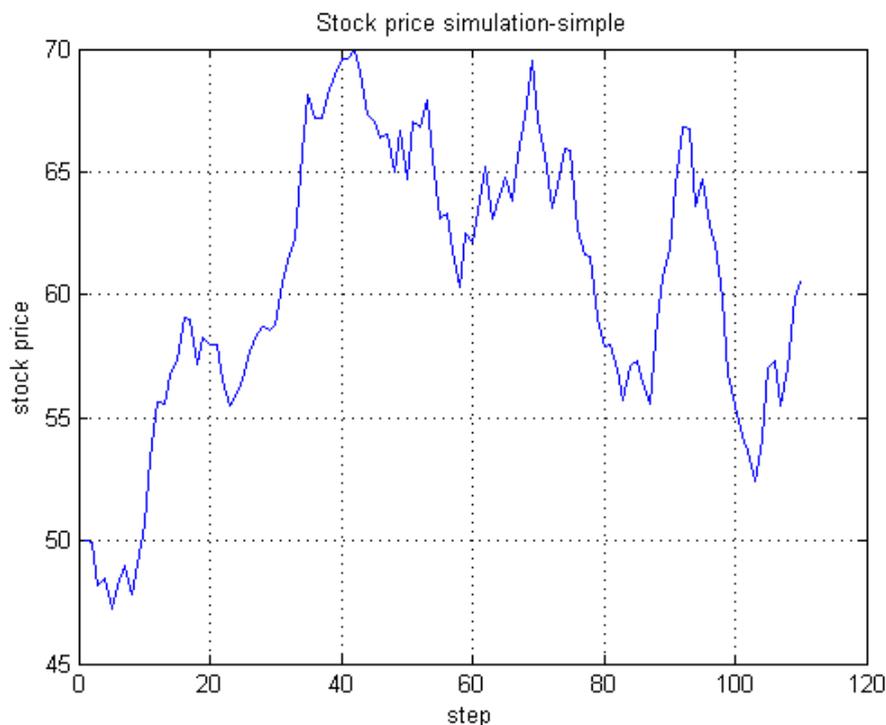
$$\ln S_T \sim N\left[\ln S_t + \left(r - \frac{\sigma^2}{2}\right)(T-t), \sigma\sqrt{T-t}\right]$$

故在時點  $T$  之資產價格（ $S_T$ ）可表示為：

$$S_T = S_t \cdot \exp\left[\left(r - \frac{\sigma^2}{2}\right)(T-t) + \sigma\sqrt{T-t} \cdot \varepsilon\right]$$

其中  $\varepsilon$  為自平均數為 0，標準差為 1 之標準常態分配中所抽取之隨機樣本。在資產價格模擬時，將選擇權的存續期間分割成  $N$  期，每期皆為  $\Delta t$ （ $= T/N$ ），然後自標準常態分配中抽出  $N$  個獨立之隨機樣本後，依據上式即可計算出在時點  $0, \Delta t, 2\Delta t, \dots, T$  時之資產價格，如此便產生一條資產價格模擬之路徑，並可以利用  $V = f(S_0, \dots, S_T)$  求算新奇選擇權之收益值。每一條資產價格模擬路徑與相應的選擇權收益值之計算，稱為一次的模擬試行（Simulation Trial）。在經過大量的模擬試行之後，譬如 100,000 次，可以得到 100,000 條資產價格模擬路徑及其對應的選擇權收益值，即將模擬所得之 100,000 個選擇權收益值予以平均，再以無風險利率折現，便可得出該選擇權價值的估計值。

<sup>3</sup> 林蒼祥、蔡蒔銓（2005）。從物理學到財務金融—漫談跨學科的結合。《物理雙月刊》，27（6），795-802。



- 信賴區間

在試行M次的模擬後，可求得期估計值的平均數 $\mu_{\hat{C}}$ ，以及標準差w，以此可建立一信賴區間，投資人可依據自己的容忍誤差程度不大於此信賴區間，以決定蒙地卡羅模擬的試行次數，容忍誤差程度越小，則求得的選擇權價格越精確，但所花費時間也越長。

$$\mu_{\hat{C}} - 1.96 \frac{w}{\sqrt{M}} < f < \mu_{\hat{C}} + 1.96 \frac{w}{\sqrt{M}}$$

於實務上，容忍誤差是由人為直接訂定，通常是公司依據自身所持有的投資組合，並參考市場狀況調整，決定出一 $\alpha$ 值做為公司的風險管理政策。

- Antithetic variate method

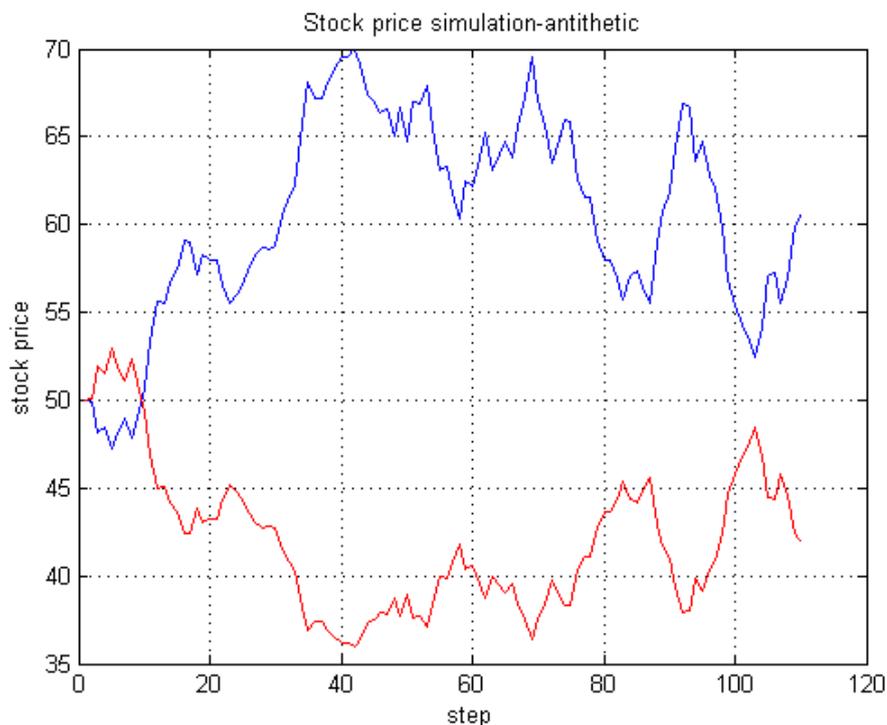
每產生一次隨機值 $\varepsilon$ ，即產生兩條模擬路徑：

$$S_{dt} = S_0 e^{\left(r - \frac{\sigma^2}{2}\right)dt + \sigma\varepsilon\sqrt{dt}}$$

$$S_{dt} = S_0 e^{\left(r - \frac{\sigma^2}{2}\right)dt + \sigma(-\varepsilon)\sqrt{dt}}$$

如此可同時估計得 $\hat{C}_1$ 、 $\hat{C}_2$ ，而選擇權的估計值即為兩者平均值：

$$\hat{C} = \frac{\hat{C}_1 + \hat{C}_2}{2}$$



- Control variate method<sup>4</sup>

找出與選擇權C具有高度相關，且存在公式解的另一衍生性商品G，以 $G_{BS}$ 表示，在蒙地卡羅模擬C的標的資產路徑同時，也模擬G的標的資產路徑，進而求得G的估計值 $\hat{G}$ ，此時C的估計值 $C^*$ 為：

$$C^* = G_{BS} + (\hat{C} - \hat{G})$$

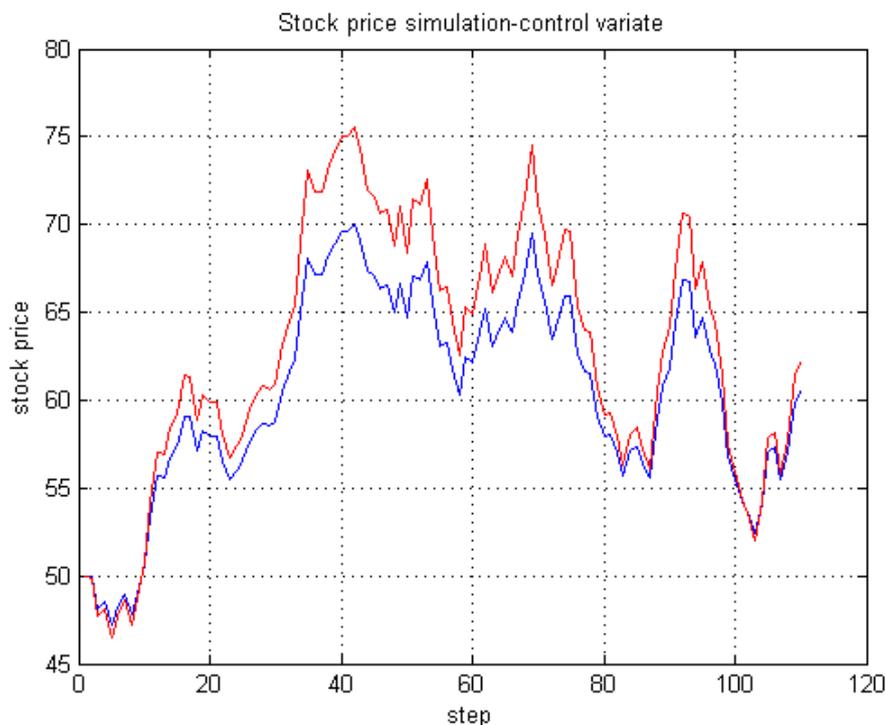
此時 $C^*$ 的變異數為：

$$\text{Var}(C^*) = \text{Var}(\hat{C}) + \text{Var}(\hat{G}) - 2\text{Cov}(\hat{C}, \hat{G})$$

因故，只要 $2\text{Cov}(\hat{C}, \hat{G}) > \text{Var}(\hat{G})$ ，則control variate method即能達到降低估計誤差的效果。

---

<sup>4</sup> Boyle, P. P. (1976). Options: a monte carlo approach. *Journal of Financial Economics*, 4, 323-332.



● 結論

以容忍區間為0.2為例，試行蒙地卡羅基本型、Antithetic variate method及 Control variate method，在Control variate method的模擬中，以波動率為0.5的買權做為高度相關之衍生性商品，當信賴區間小於等於容忍區間時，即停止模擬。由以下結果可以發現在收斂的效率性上，Control variate method明顯優於Antithetic variate method，而基本型的效率則最差。

(A)

最後透過 Matlab 使用蒙地卡羅-simple 計算出的歐式買權價格為 5.9974 元

而利用 B-S model 計算出的歐式買權價格為 6.1151 元

95%信賴區間為 [5.8974 6.1000]

因此蒙地卡羅-simple 和 B-S model 計算出的歐式買權價格誤差為

$$(5.9974-6.1151)/6.1151=-0.0192 (1.92\%)$$

試行次數為 33755 次

(B)

最後透過 Matlab 使用蒙地卡羅- antithetic 計算出的歐式買權價格為 6.1031 元

而利用 B-S model 計算出的歐式買權價格為 6.1151 元

95%信賴區間為 [6.0031 6.2031]

因此蒙地卡羅-antithetic 和 B-S model 計算出的歐式買權價格誤差為

$$(6.1031-6.1151)/6.1151=0.0020 (0.2\%)$$

試行次數為 9768 次

(C)

最後透過 Matlab 使用蒙地卡羅-control 計算出的歐式買權價格為 6.2351 元

而利用 B-S model 計算出的歐式買權價格為 6.1151 元

95%信賴區間為 [6.1351 6.3351]

因此蒙地卡羅-control 和 B-S model 計算出的歐式買權價格誤差為

$(6.2351-6.1151)/6.1151=0.0196$  (1.96%)

試行次數為 2440 次

在使用蒙地卡羅-control 方法的狀況下，當滿足停止模擬條件時(試行次數 2440 次)，蒙地卡羅-control 的標準誤為 0.051，同時蒙地卡羅-antithetic 的標準誤為 0.1，而蒙地卡羅-simple 的標準誤為 0.18。

#### (4) 有限差分法

- 理論<sup>5</sup>

有限差分法最早由 Schwartz 於 1977 年提出，為一種用來解偏微分方程的數值方法，而後 Courtadon 在 1982 年再加以延伸應用在財務領域上。基本概念是將偏微分方程式轉為差分方程式，以離散變量函數求解近似連續變量函數，也就是把連續函數中定義的區域用有限的離散點所構成的網格來代替，經由計算網格上各節點的理論價值，就可推得出差分方程式的解。

- 基本觀念<sup>6</sup>

$$\frac{\partial Y}{\partial X} \rightarrow \frac{\Delta Y}{\Delta X}$$

$$\left(\frac{dY}{dX}\right)_i = \lim_{\Delta X \rightarrow 0} \frac{Y_{i+1}-Y_i}{\Delta X} = \lim_{\Delta X \rightarrow 0} \frac{\Delta Y_i}{\Delta X}$$

$$\text{向前差分法：} \frac{\Delta Y}{\Delta X} = \frac{Y_{i+1}-Y_i}{\Delta X}$$

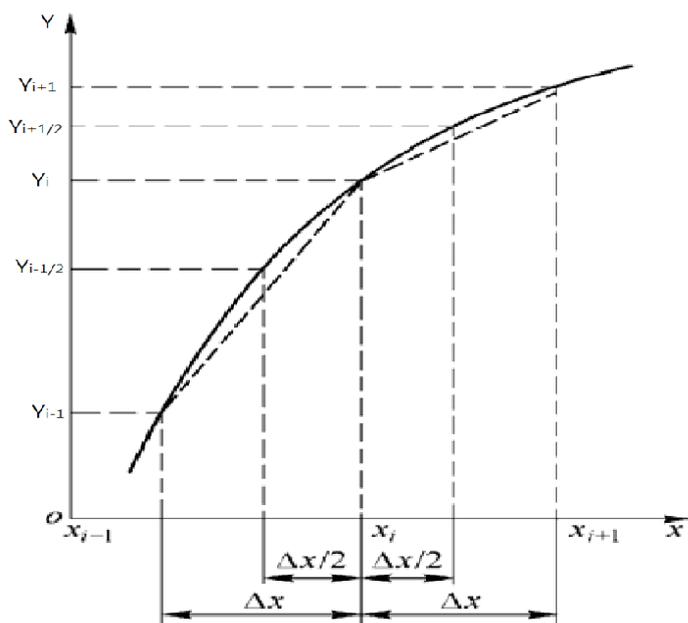
$$\text{向後差分法：} \frac{\Delta Y}{\Delta X} = \frac{Y_i-Y_{i-1}}{\Delta X}$$

$$\text{中心差分法：} \frac{\Delta Y}{\Delta X} = \frac{Y_{i+1}-Y_{i-1}}{2\Delta X}$$

$$\text{二階差分法：} \frac{\Delta^2 Y}{\Delta X^2} = \frac{Y_{i+1}+Y_{i-1}-2Y_i}{\Delta X^2}$$

<sup>5</sup> 何泓璋 (2006)。使用二維隱性有限插分法做美式互換選擇權之評價。碩士論文，國立高雄第一科技大學，高雄市。

<sup>6</sup> 陳凌鶴、蔡蒔銓、曾雅盟 (2010)。財務工程：實務與個案探討。台北市：證券暨期貨市場發展基金會。



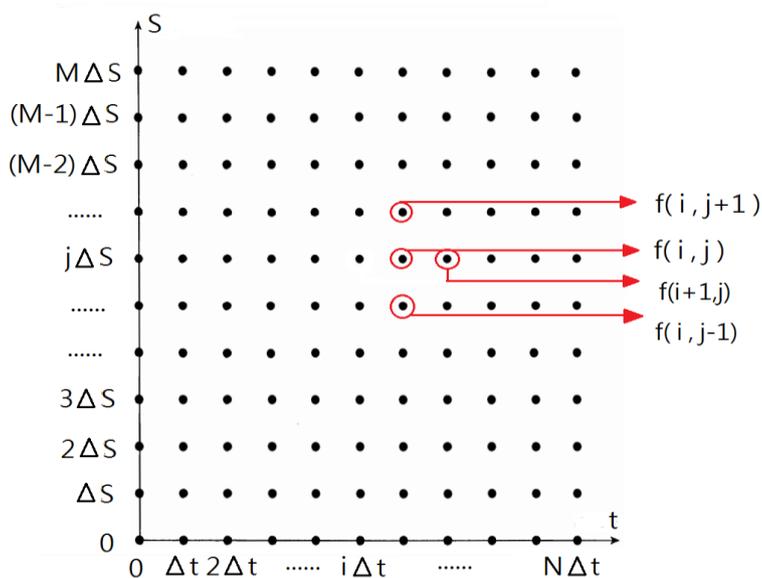
選擇權的微分方程： $\frac{\partial f}{\partial t} + (r - q)S \frac{\partial f}{\partial S} + \frac{1}{2} \sigma^2 S^2 \frac{\partial^2 f}{\partial S^2} = rf$

● 隱式有限差分

$$\frac{\partial f}{\partial S} = \frac{f_{i,j+1} - f_{i,j-1}}{2\Delta S}$$

$$\frac{\partial f}{\partial t} = \frac{f_{i+1,j} - f_{i,j}}{\Delta t}$$

$$\frac{\partial^2 f}{\partial S^2} = \frac{\frac{f_{i,j+1} - f_{i,j}}{\Delta S} - \frac{f_{i,j} - f_{i,j-1}}{\Delta S}}{\Delta S} = \frac{f_{i,j+1} + f_{i,j-1} - 2f_{i,j}}{\Delta S^2}$$



將上述三式代入選擇權微分方程，並令  $S = j\Delta S$  可得：

$$\frac{f_{i+1,j} - f_{i,j}}{\Delta t} + (r - q)j\Delta S \frac{f_{i,j+1} - f_{i,j-1}}{2\Delta S} + \frac{1}{2}\sigma^2 j^2 \Delta S^2 \frac{f_{i,j+1} + f_{i,j-1} - 2f_{i,j}}{\Delta S^2} = rf_{i,j}$$

整理後可得：

$$a_j f_{i,j-1} + b_j f_{i,j} + c_j f_{i,j+1} = f_{i+1,j}$$

其中：  $a_j = \frac{1}{2}(r - q)j\Delta t$

$$b_j = 1 + \sigma^2 j^2 \Delta t + r\Delta t$$

$$c_j = -\frac{1}{2}(r - q)j\Delta t - \frac{1}{2}\sigma^2 j^2 \Delta t$$

\*在決定股票價格間距時，最佳的方法是參考市場選擇權契約的價差。

進一步可得 M-1 條方程式：

$$\begin{bmatrix} b_1 & c_1 & 0 & \cdots & 0 & 0 & 0 \\ a_2 & b_2 & c_2 & \cdots & 0 & 0 & 0 \\ 0 & a_3 & b_3 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 & 0 & 0 \\ 0 & 0 & 0 & \cdots & b_{M-3} & c_{M-3} & 0 \\ 0 & 0 & 0 & \cdots & a_{M-2} & b_{M-2} & c_{M-2} \\ 0 & 0 & 0 & \cdots & 0 & a_{M-1} & b_{M-1} \end{bmatrix} \begin{bmatrix} f_{N-1,1} \\ f_{N-1,2} \\ f_{N-1,3} \\ \vdots \\ f_{N-1,M-3} \\ f_{N-1,M-2} \\ f_{N-1,M-1} \end{bmatrix} = \begin{bmatrix} f_{N,1} \\ f_{N,2} \\ f_{N,3} \\ \vdots \\ f_{N,M-3} \\ f_{N,M-2} \\ f_{N,M-1} \end{bmatrix}$$

● 結論

最後透過 Matlab 使用隱式有限差方計算出的歐式買權價格為 5.9407 元

而利用 B-S model 計算出的歐式買權價格為 6.1151 元

因此三項樹和 B-S model 計算出的歐式買權價格誤差為

$$(5.9407 - 6.1151) / 6.1151 = -0.0285 \text{ (2.85\%)}$$

利用隱式有限差分所求得買權價格

Time to maturity											
s	5	4.5	4	3.5	3	2.5	2	1.5	1	0.5	0
100	50	50	50	50	50	50	50	50	50	50	50
95	45.58	45.54	45.50	45.46	45.42	45.37	45.32	45.27	45.20	45.12	45.00
90	41.05	40.97	40.89	40.81	40.73	40.64	40.54	40.43	40.31	40.17	40.00
85	36.43	36.31	36.19	36.07	35.95	35.82	35.68	35.53	35.37	35.19	35.00
80	31.75	31.59	31.43	31.27	31.10	30.94	30.76	30.58	30.40	30.20	30.00
75	27.03	26.84	26.64	26.43	26.23	26.03	25.82	25.62	25.41	25.21	25.00
70	22.35	22.10	21.86	21.62	21.37	21.13	20.89	20.66	20.43	20.21	20.00
65	17.76	17.47	17.18	16.88	16.59	16.29	16.01	15.73	15.47	15.22	15.00
60	13.38	13.04	12.70	12.35	12.00	11.63	11.27	10.91	10.57	10.26	10.00
55	9.37	9.00	8.62	8.23	7.81	7.37	6.91	6.42	5.92	5.43	5.00
50	5.94	5.58	5.19	4.79	4.35	3.88	3.36	2.77	2.07	1.19	0.00
45	3.30	2.99	2.67	2.34	2.00	1.65	1.29	0.92	0.56	0.23	0.00
40	1.55	1.34	1.13	0.93	0.74	0.55	0.38	0.24	0.12	0.04	0.00
35	0.59	0.48	0.38	0.29	0.21	0.14	0.09	0.05	0.02	0.01	0.00
30	0.17	0.13	0.10	0.07	0.05	0.03	0.02	0.01	0.00	0.00	0.00
25	0.04	0.03	0.02	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00
20	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
15	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
5	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

● 顯式與隱式差分的差異

Method	隱式有限差分	顯式有限差分
差異處		
起始點	(i, j)	(i+1, j)
先求得	$f_{i+1, j}$	$f_{i, j}$
計算方法	由後反推，需解聯立方程式	與二項樹同，亦由後反推
穩定性	穩定	有誤差，不穩定
能否求解美式選擇權	可以求美式選擇權	可求美式選擇權，但需做調整

(5) 各種計算方式比較

Method	Call Price	與B-S Price的誤差
B-S model	6.1151	0%
二項樹	6.1038	0.18%
三項樹	6.3638	4.07%
蒙地卡羅-基本型	5.9974	1.92%
蒙地卡羅-antithetic	6.1031	0.20%
蒙地卡羅-control	6.2351	1.96%
隱式有限差分	5.9407	2.85%

## 選擇權價格計算方式 Matlab 程式碼

- **BS-model**

```
n=100; %共執行 100step  
S=50; %現在股價為 50  
K=50; %選擇權履約價格為 50  
r=0.1; %年利率為 10%  
T=5/12; %選擇權距離到期有 5 個月  
vol=0.4; %波動率為 40%  
dt=T/n; %每個 step 所需經歷時間
```

```
%B-S MODEL [Call, Put] = blsprice(Price, Strike, Rate, Time, Volatility)  
[Call, Put] = blkprice(S, K, r, T, vol);  
bs_call=Call;  
fprintf('利用 B-S model 計算出的歐式買權價格為 %4.4f\n',bs_call)
```

- **二項樹**

```
clear;clc  
%設定參數  
n=100; %共執行 100step  
S=50; %現在股價為 50  
K=50; %選擇權履約價格為 50  
r=0.1; %年利率為 10%  
T=5/12; %選擇權距離到期有 5 個月  
vol=0.4; %波動率為 40%  
dt=T/n; %每個 step 所需經歷時間
```

```
%二項樹 [AssetPrice, OptionValue] = binprice(Price, Strike, Rate, Time,%Increment,Volatility, Flag)  
%Flag: put=0 call=1  
[Price, Option] = binprice(S, K, r, T, dt, vol, 1);  
bin_call=Option(1,1);  
fprintf('利用二項樹計算出的歐式買權價格為 %4.4f\n',bin_call)
```

● 三項樹

```
clear;clc
n=100;      %共執行 100step
S=50;      %現在股價為 50
K=50;      %選擇權履約價格為 50
r=0.1;     %年利率為 10%
T=5/12;    %選擇權距離到期有 5 個月
vol=0.4;   %波動率為 40%
dt=T/n;    %每個 step 所需經歷時間

u = exp(vol*sqrt(3*dt));           %股票的漲幅
d = 1/u;                           %股票的跌幅
pd=-sqrt(dt/(12*vol^2))*(r-0.5*vol^2)+1/6; %股價上漲的機率
pm=2/3;                             %股價不變的機率
pu=sqrt(dt/(12*vol^2))*(r-0.5*vol^2)+1/6; %股價下跌的機率
pr=[pu;pm;pd];                    %將各種機率令為矩陣
SS=zeros(2*n+1,n+1);

XX=zeros(n*2+1,1);
XX(n+1,1)=S;                       %將現時股價放入空矩陣中
SS(:,1)=XX;

%計算出各期股價 SS 矩陣
for i=1:n
    uu=S*u^i;
    dd=S*d^i;
    XX([(n+1)-i,(n+1)+i],1)=[uu,dd];
    SS(:,i+1)=XX;
end

%計算最後到期日的可能選擇權價值 call_100
call_100=SS(:,n+1)-K;
for i=1:size(call_100)
    if call_100(i,1)<0;              %若 S-K<0,則選擇權價值為 0
        call_100(i,1)=0;
    end
end
end
```

```
%從最後一期選擇權價格往前反推，最後得全部期間選擇權價格 CC
CC=zeros(2*n+1,101);
CC(:,n+1)=call_100;
for j=1:n
    tic
    YY=zeros(2*n+1,1);
    for k=1:2*n-j
        eval(['YY',num2str(j),'(k+j,1)=sum(CC((k+j-1:k+2+j-1),(n+1)-j+1).*pr);'])
    end
    eval([' CC(1+j:(2*n+1)-j,(n+1)-j)=YY',num2str(j),'((n+1)-n+j:(n+1)+n-j,1);'])
    clc,process=j,toc
end

%打印出答案
CALL_PRICE=CC(n+1,1);
clc;fprintf('利用三項樹計算出的歐式買權價格為%4.4f\n',CALL_PRICE)
```

- **蒙地卡羅**

```
clear;clc
S=50;           %現在股價為 50
K=50;           %選擇權履約價格為 50
r=0.1;          %年利率為 10%
vol=0.4;        %波動率為 40%
T=5/12;         %距到期期間長度
n=110;          %假設五個月共有 110 個交易日
dt=T/n;         %dt 為一個交易日
inter=0.1;      %設定一個可容忍的區間
vol2=0.5;       %control variate 模擬的高度相關衍生商品
[Call, Put] = blsprice(S, K, r, T, vol2); %高度相關衍生商品
G=Call;

for i=1:inf;
    tic
    MC_ST_posi(1,i)=S; %第一條模擬路徑的初始值
    MC_ST_negi(1,i)=S; %第二條模擬路徑的初始值
    MC_ST_hat(1,i)=S; %高度相關衍生性商品模擬路徑的初始值
```

```

for j=2:n;
    RAN=randn(1,1);    %產生隨機值
    %第一條股價模擬路徑
    MC_ST_posi(j,i)=MC_ST_posi(j-1,i)*exp((r-0.5*vol^2)*dt+vol*RAN*dt^0.5);
    %第二條股價模擬路徑
    MC_ST_negi(j,i)=MC_ST_negi(j-1,i)*exp((r-0.5*vol^2)*dt+vol*(-RAN)*dt^0.5);
    %高度相關衍生性商品模擬路徑
    MC_ST_hat(j,i)=MC_ST_hat(j-1,i)*exp((r-0.5*vol2^2)*dt+vol2*RAN*dt^0.5);
end
%第一條路徑決定的選擇權價值
MS_CT_posi=MC_ST_posi(end,i)-K;    %最後一期買權價值 = S-K
if MS_CT_posi<0                    %若 S-K<0, 則買權價值為 0
    MS_CT_posi=0;
end
%第二條路徑決定的選擇權價值
MS_CT_negi=MC_ST_negi(end,i)-K;    %最後一期買權價值 = S-K
if MS_CT_negi<0                    %若 S-K<0, 則買權價值為 0
    MS_CT_negi=0;
end
%高度相關衍生性商品路徑的選擇權價值
MS_CT_hat=MC_ST_hat(end,i)-K;    %最後一期買權價值 = S-K
if MS_CT_hat<0                    %若 S-K<0, 則買權價值為 0
    MS_CT_hat=0;
end

CALL_simple(i,1)=MS_CT_posi*exp(-r*T);    %基本型的選擇權現值
meanCALL_simple=mean(CALL_simple);        %買權的平均現值-基本型
sdCALL_simple=std(CALL_simple);          %買權現值的標準差-基本型

CALL_antithetic(i,1)=(MS_CT_posi+MS_CT_negi)/2*exp(-r*T); %antithetic 調整後的選擇權現值
meanCALL_antithetic=mean(CALL_antithetic);    %買權的平均現值-antithetic
sdCALL_antithetic=std(CALL_antithetic);        %買權現值的標準差-antithetic

CALL_hat(i,1)=MS_CT_hat*exp(-r*T);    %相關模擬商品的現值
CALL_control=G+CALL_simple-CALL_hat;    %control 調整後的選擇權現值
meanCALL_control=mean(CALL_control);    %買權的平均現值-control
sdCALL_control=std(CALL_control);        %買權現值的標準差-control

```

```
%當信賴區間小於容忍區間時，中止模擬
if i>1
    if inter>2*1.96*sdCALL_control/i^0.5
        break
    end
end
clc,process=i,toc
end

clc
fprintf('總共模擬%4.0f 次\n',i)
fprintf('\n')
%打印出 control 調整後蒙地卡羅的結果
e=meanCALL_control-1.96*sdCALL_control/i^0.5;
f=meanCALL_control+1.96*sdCALL_control/i^0.5;
fprintf('蒙地卡羅-control\n')
fprintf('CALL PRICE=%4.4f\n',meanCALL_control)
fprintf('std. error=%4.4f\n',sdCALL_control/i^0.5)
fprintf('95%%信賴區間為 [%4.4f %4.4f]\n',e,f)
fprintf('\n')
%打印出 antithetic 調整後蒙地卡羅的結果
c=meanCALL_antithetic-1.96*sdCALL_antithetic/i^0.5;
d=meanCALL_antithetic+1.96*sdCALL_antithetic/i^0.5;
fprintf('蒙地卡羅-antithetic\n')
fprintf('CALL PRICE=%4.4f\n',meanCALL_antithetic)
fprintf('std. error=%4.4f\n',sdCALL_antithetic/i^0.5)
fprintf('95%%信賴區間為 [%4.4f %4.4f]\n',c,d)
fprintf('\n')
%打印出蒙地卡羅基本型的結果
a=meanCALL_simple-1.96*sdCALL_simple/i^0.5;
b=meanCALL_simple+1.96*sdCALL_simple/i^0.5;
fprintf('蒙地卡羅-基本型\n')
fprintf('CALL PRICE=%4.4f\n',meanCALL_simple)
fprintf('std. error=%4.4f\n',sdCALL_simple/i^0.5)
fprintf('95%%信賴區間為 [%4.4f %4.4f]\n',a,b)
```

● 隱式有限差分

%時間間距為 0.5 個月，選擇權到期日為 5 個月，股價間距為 5 元，最大值為 100 最小值為 0

```

for j=1:19
    b(j)=1+0.4^2*j^2*(1/24)+0.1*(1/24);          %計算 bj 係數
end
for j=2:19
    aa(j)=0.5*0.1*j*(1/24)-0.5*0.4^2*j^2*(1/24); %計算 aj 係數
end
for j=1:18
    c(j)=-0.5*0.1*(j)*(1/24)-0.5*0.4^2*(j)^2*(1/24); %計算 cj 係數
end
a=aa(1,2:19);
B=flipr(b);
A=flipr(a);
C=flipr(c);          %把 a,b,c 左右顛倒放
B1=diag(B,0);        %把 bj 插到主對角線
C1=diag(C,-1);       %把 cj 差到主對角線下一條
A1=diag(A,1);        %把 aj 差到主對角線上一條
X=B1+C1+A1;
j=19;
c(j)=-0.5*0.1*j*(1/24)-0.5*0.4^2*j^2*(1/24);
%ttm=0 時的選擇權價格 Y0=[45-c(19)*50;40;35;30;25;20;15;10;5;0;0;0;0;0;0;0;0;0];
final=X\Y0;          %結果為 ttm=0.5 時的選擇權價格

TOTALCALL=zeros(19,11); %先把最後結果的矩陣列出來
TOTALCALL(:,11)=[45;Y0(2:end,1)]; %把 ttm=0 時的選擇權價格放到第 11 行
TOTALCALL(:,10)=final; %把 ttm=0.5 時的選擇權價格放到第 10 行

for w=1:9
    Y0=final;
    Y0(1,1)=Y0(1,1)-c(19)*50;
    final=X\Y0; %重複上面計算選擇權價格，將 ttm 為 1 到 5 的選擇權價格算出
    TOTALCALL(:,10-w)=final; %依序將算出來的選擇權價格放到第 9 行到第 1 行
end
d(1:11)=50; %股價為最大值 100 時的選擇權價格
e(1:11)=0; %股價為最小值 0 時的選擇權價格
ANS=[d;TOTALCALL;e]; %最後算出橫軸為 t 縱軸為 s 的選擇權價格
CALL_PRICE=ANS(11,1); %最後求出當股價為 50,到期日為 5 個月的選擇權價格
    
```