Ch 9. Lookback Option

I. Analytic Solutions and Monte Carlo Simulation for Lookback Options

II. Pricing Lookback Options with the Binomial Tree

III. Finite Difference Method for Path Dependent Options

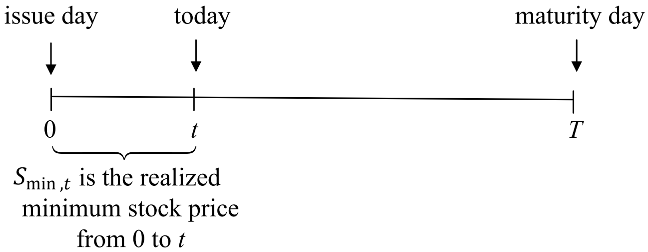
IV. Reset Option

- This chapter introduces the analytic solution, Monte Carlo simulation, binomial tree model, and finite difference method to price lookback options. The application of the finite difference method to price various types of path dependent options is also discussed. Finally, the pricing method for the reset option, which is equal to a lookback option with a limited set of sampling time points, will be introduced.

I. Analytic Solutions and Monte Carlo Simulation for Lookback Options

- European lookback call: $c_T = \max\,(S_T - m_0^T, 0)$, where $m_0^T = \min\limits_{0 \le \tau \le T} S_\tau$.

  Figure 9-1



$$
\begin{aligned}
c_t = {} & S_t e^{-q(T-t)} N(a_1) - S_t e^{-q(T-t)} \tfrac{\sigma^2}{2(r-q)} N(-a_1) \\
& - S_{\min,t} e^{-r(T-t)} (N(a_2) - \tfrac{\sigma^2}{2(r-q)} e^{Y_1} N(-a_3)),
\end{aligned}
$$

where

$a_1 = \frac{\ln(S_t/S_{\min,t}) + (r-q+\frac{\sigma^2}{2})(T-t)}{\sigma\sqrt{T-t}}$, $a_2 = a_1 - \sigma\sqrt{T-t}$, $a_3 = \frac{\ln(S_t/S_{\min,t}) + (-r+q+\frac{\sigma^2}{2})(T-t)}{\sigma\sqrt{T-t}}$,

and $Y_1 = -\frac{2(r-q-\frac{\sigma^2}{2})\cdot\ln(S_t/S_{\min,t})}{\sigma^2}$.

$*$ If $t = 0$, then $S_t = S_0$ and $S_{\min,t} = S_{\min,0} = S_0$.

* $c_0 = e^{-rT}E^Q[\max(S_T - m_0^T, 0)]$
  $= e^{-rT}E^Q[S_T - m_0^T]$
  $= e^{-rT}E^Q[S_T] - e^{-rT}E^Q[m_0^T]$
  $= S_0 - S_0 e^{-rT}E^Q[\frac{m_0^T}{S_0}]$

* Harrison (1985) derives the formula of the following joint cumulative distribution function.
  $P^Q = P^Q(\ln\frac{S_T}{S_0} \geq x, \ln\frac{m_0^T}{S_0} \geq y)$ given $y \leq 0$ and $y \leq x$.
  By setting x to be y, we obtain
  $P^Q = P^Q(\ln\frac{S_T}{S_0} \geq y, \ln\frac{m_0^T}{S_0} \geq y) = P^Q(\ln\frac{m_0^T}{S_0} \geq y)$.
  Finally, the probability density function of $\ln\frac{m_0^T}{S_0}$ can be derived as $\frac{\partial(1-P^Q)}{\partial y}$.

- As to the pricing formulae for the European lookback put with the payoff to be $\max(M_0^T - S_T, 0)$, where $M_0^T = \max_{0\leq u\leq T} S_u$ or other variations of lookback options, please refer to Conze and Viswanathan (1991), "Path Dependent Options: The Case of Lookback Options," *Journal of Finance* 46, pp.1893–1907.

- Note that the above formula is valid only when the lookback mimimum (or maximum) is sampled continuously. However, the continuously sampling is not infeasible in practice. (Strictly speaking, even the quotations of the stock price are not continuous.) It is common to adopt the daily sampling rule in financial markets.

- It is straightforward to apply the Monte Carlo simulation to pricing discretely-sampling lookback option, whose value is the average present value of the payoff of the lookback option associated with each simulated path. When the sampling frequency increases, the results of the Monte Carlo simulation can approach the theoretical value based on the above analytic solution.

II. Pricing Lookback Options with the CRR Binomial Tree

- American lookback put with

$$\text{Payoff}_t = \max(S_{\max,t} - S_t, 0), \text{ where } S_{\max,t} = \max_{0 \le \tau \le t} S_\tau, \text{ for } 0 \le t \le T.$$

Suppose $T = 0.25$ year (3 months)

$$S_0 = S_{\max,0} = 50$$

$$r = 0.1,\ q = 0,\ \sigma = 0.4$$

$$\Delta t = \frac{1}{12} \text{ year (1 month)}$$

$$\Rightarrow \begin{cases} u = 1.1224 \\ d = 0.8909 \\ p = 0.5073 \end{cases}$$

Figure 9-2

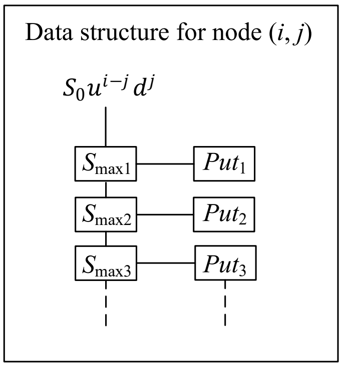
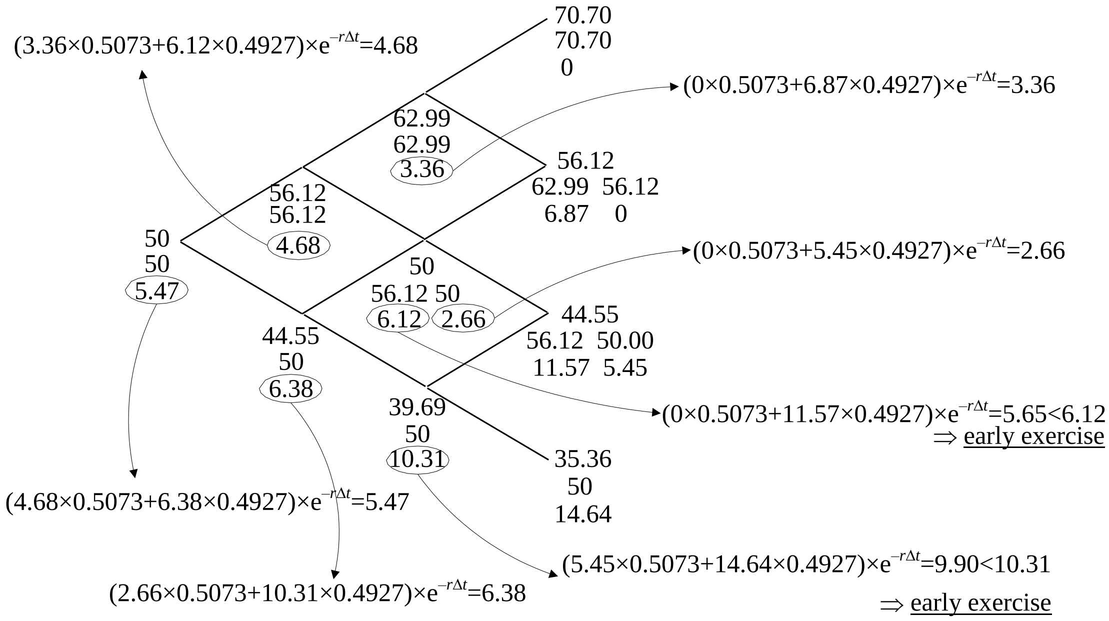


Figure 9-3 This example is from Ch. 26 in Hull (2011)



Algorithm:

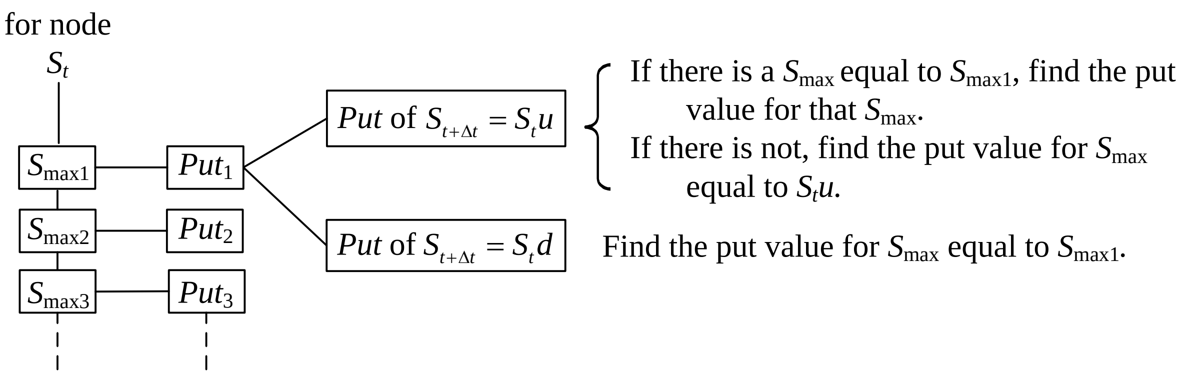
(i) Build the tree, and record possible $S_{\max}$'s for each node. The forward-tracking method to record possible $S_{\max}$'s is explained as follows. For a node with the stock price S_t,

inherit $S_{\max}$'s from parents $\begin{cases} \text{1. If } S_{\max} \text{ from parents} \geq S_t \Rightarrow \text{Insert this } S_{\max} \text{ into its } S_{\max}\text{-list} \\ \text{2. If } S_{\max} \text{ from parents} < S_t \Rightarrow \text{Ignore } S_{\max} \text{ and insert } S_t \text{ into its } S_{\max}\text{-list} \end{cases}$

(ii) For each terminal node, decide the payoff for every $S_{\max}$ of each terminal node.

(iii) Backward induction: see Figure 9-4.

Figure 9-4



III. Finite Difference Method for Path Dependent Options

- In Chapter 5, I introduce the finite difference method (FDM) to price plain vanilly options. In fact, the FDM is a general framework to price various types of path dependent options.

(1) $\frac{dS}{S} = (\mu - q)dt + \sigma dZ$

(2) Define the path-dependent variable (路徑相依變數)

$I(T) = \int_0^T f(S,\tau)d\tau$

or $I(t) = \int_0^t f(S,\tau)d\tau$

or $dI \ = f(S,t)dt$

vanilla option:	$f(S,t) = 0,$	$I(t) = I(T) = 0$
arithmetic average option:	$f(S,t) = S,$	$I(t) = \int_0^t S(\tau)d\tau$
geometric average option:	$f(S,t) = \ln(S),$	$I(t) = \int_0^t \ln(S(\tau))d\tau$

Thus, the option value $V(S(t), I(t), t)$ is the function of S, I, and t.

(3) Payoff at T: $P(S(T), I(T), T)$

vanilla call:	$P(S(T), I(T), T) = \max(S(T) - K, 0)$
arithmetic average call:	$P(S(T), I(T), T) = \max(I(T)/T - K, 0)$
geometric average call:	$P(S(T), I(T), T) = \max(\exp(I(T)/T) - K, 0)$

(4) Construct a riskless portfolio $\pi = -V(S,I,t) + \frac{\partial V}{\partial S}S$

According to the Itô's lemma in the case of three variables,

$$dV = (\frac{\partial V}{\partial t} + \frac{\partial V}{\partial S}(\mu - q)S + \frac{1}{2}\frac{\partial^2 V}{\partial S^2}\sigma^2 S^2 + \frac{\partial V}{\partial I}f(S,t))dt + (\frac{\partial V}{\partial S}\sigma S)dZ$$

$$\begin{aligned}\Rightarrow d\pi &= -dV + \frac{\partial V}{\partial S}dS + q(\frac{\partial V}{\partial S}Sdt) \\ &= -(\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + f(S,t)\frac{\partial V}{\partial I} - q\frac{\partial V}{\partial S}S)dt \\ &= r \cdot \pi \cdot dt\end{aligned}$$

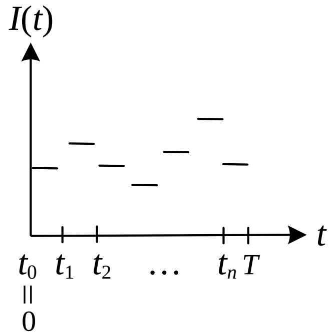
$$\Rightarrow \frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + f(S,t)\frac{\partial V}{\partial I} + (r-q)S\frac{\partial V}{\partial S} - rV = 0$$

Apply the finite difference method on the S-I-t space to solve $V(S,I,t)$ numerically, and the option value today is $V(S(0), I(0), 0)$. Since there are three dimensions, the backward induction is conducted from T to 0 along a rectangular solid. More specifically, for each time point t_i, the option values on a S-I plane are considered.

- When ΔS, ΔI, and Δt approach 0, option values converge to the results based on the continuous sampling rule. However, for most contracts in the real world, the path dependent variable I is not sampled continuously, but sampled discretely. For example, the payoffs of arithmetic average options and lookback options depend only on the daily or weekly closing prices. Next I will introduce how to employ the finite difference method to deal with discretely-sampling path dependent options.

- In the case of discrete sampling, the value of I changes only at sampling points.

Figure 9-5



- For the time points other than the sampling points, the value of I maintains the same, i.e., $dI = 0$, which implies $f(S,t) = 0$ and thus

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

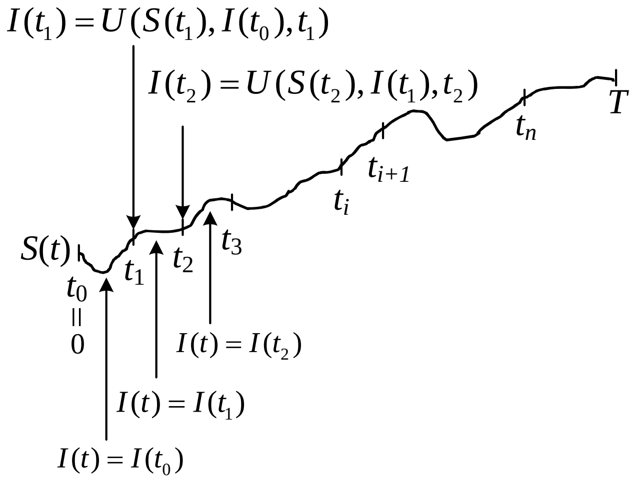
which is independent of I. However, because the payoff depends on the value of I, it is still necessary to apply the finite difference method to solving the above PDE in the S-I-t space.

- At updating (or sampling) points, since $dI \neq 0$, an updating rule of I is considered.

  Case 1: When $t_i \leq t < t_{i+1}$, $I(t) = I(t_i)$ (for time points that are not sampling points)

  Case 2: When $t = t_{i+1}$, $I(t_{i+1}) = U(S(t_{i+1}), I(t_i), t_{i+1})$ (for sampling time points)

Figure 9-6 Illustrating how the updating rule works



- The updating rule for arithmetic average options and lookback options

For arithmetic average options: Define $A_j = \frac{1}{j}\sum_{i=1}^{j} S(t_i)$

$$\Rightarrow A_1 = S(t_1)$$

$$\Rightarrow A_2 = \frac{S(t_1)+S(t_2)}{2} = \frac{1}{2}A_1 + \frac{1}{2}S(t_2)$$

$$\vdots$$

$$\Rightarrow A_j = \frac{j-1}{j}A_{j-1} + \frac{1}{j}S(t_j)$$

previous average is with the weight $(j-1)/j$

stock price on the sampling date is with the weight of $1/j$

(The general rule is to express the evolution of the path dependent variable as a funcion of the current stock price and the previous value of the path dependent variable until the last sampling point.)

(The information of the current time point j is to decide the weight coefficients.)

For lookback options: Define $I(t_j) = \max(S(t_1), \ldots, S(t_j))$

$$\Rightarrow I(t_1) = S(t_1)$$

$$I(t_2) = \max(S(t_2), S(t_1)) = \max(S(t_2), I(t_1))$$

$$I(t_3) = \max(S(t_3), S(t_2), S(t_1)) = \max(S(t_3), I(t_2))$$

$$\vdots$$

- During the backward induction process of the finite difference method, employ the partial differential equation for $dI = 0$ for non-sampling time points, i.e., perform the backward induction based on $\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + (r-q)S\frac{\partial V}{\partial S} - rV = 0$. However, there is a special mapping rule to deal with the option values on sampling points:

$$V(S(t_i), I(t_{i-1}), t_i^-) = V(S(t_i), I(t_i), t_i^+)$$

  ⊙ Time points t_i^- and t_i^+ indicates the same time point t_i actually. You can imagine that there are two surfaces of the S-I plane at the time point t_i, and there is a relationship between the option values on the different surfaces of the S-I plane at that time point.

  ⊙ The relationship can be understood as mapping the option value of $I(t_i^-) = I(t_{i-1})$ to the option value of $I(t_i^+) = I(t_i)$ given the same value of $S(t_i)$.

- The mapping process for each combination of $(S(t_i), I(t_{i-1}))$ at sampling points t_i:

  Step (i):. Based on the updating rule $U(S(t_i), I(t_{i-1}), t_i) = I(t_i)$, and known values of $I(t_{i-1})$, $S(t_i)$, and i, $I(t_i)$ can be solved first.

  Step (ii):. The option value of the node $(S(t_i), I(t_{i-1}), t_i^-)$ is assigned to equal the option value of the node $(S(t_i), I(t_i), t_i^+)$.

- For the steps (i) and (ii) and taking lookback options for example, suppose the option payoff at maturity depends on the maximum of the stock prices at t_1, t_2, t_3, t_4, and t_5.

  Case 1. When the backward induction proceeds to t_5, suppose the goal is to find the option value $V(51, 50, t_5^-)$, where $S(t_5) = 51$, $I(t_4) = 50$.

    ⊙ Since $I(t_5) = \max(I(t_4), S(t_5))$, we can derive $I(t_5) = 51$, and the option value of $V(51, 50, t_5^-)$ is assigned to be the same as the option value of $V(51, 51, t_5^+)$.

  Case 2. When the backward induction proceeds to t_5, suppose the goal is to find the option value $V(50, 51, t_5^-)$, where $S(t_5) = 50$, $I(t_4) = 51$.

    ⊙ Since $I(t_5) = \max(I(t_4), S(t_5))$, we can derive that $I(t_5)$ is still 51 after the sampling point t_5, and the option value of $V(50, 51, t_5^-)$ is assigned to be the same as the option value of $V(50, 51, t_5^+)$.

- For the steps (i) and (ii) and taking arithmetic average options for example, suppose the option payoff at maturity depends on the average stock prices at t_1, t_2, t_3, t_4, and t_5.

  When the backward induction proceeds to t_5, suppose the goal is to find the option value $V(50, 49, t_5^-)$, where $S(t_5) = 50$, $A(t_4) = 49$.

    ⊙ Since $A(t_5) = \frac{4}{5}A(t_4) + \frac{1}{5}S(t_5)$, we can derive $A(t_5) = 49.2$, and the option value of $V(50, 49, t_5^-)$ is assigned to be the same as the option value of $V(50, 49.2, t_5^+)$.

- If there is no such node $(50, 49.2, t_5)$, apply the linear interpolation to derive the option value for $(S(t_5), A(t_5), t_5) = (50, 49.2, t_5^+)$ based on the neighboring two nodes, e.g., node $(50, 49, t_5^+)$ and node $(50, 50, t_5^+)$.

- In summary, three processes that should be implemented at each sampling point:

  1. It is necessary to allocate a 2-dimensional array for recording the option values on the t_i^- surface, $V(S(t_i), I(t_{i-1}), t_i^-)$, separately.

  2. For each $(S(t_i), I(t_{i-1}))$ on this 2-dimensional array, decide the option value $V(S(t_i), I(t_{i-1}), t_i^-)$ based on the aforementioned steps (i) and (ii).

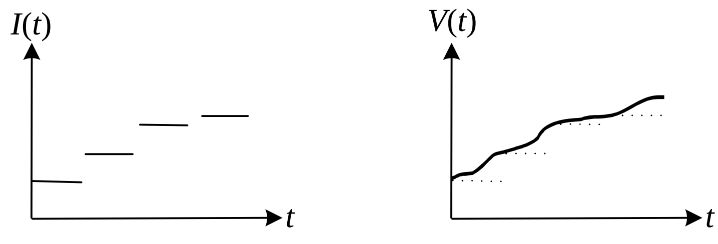
  3. Finally, replace $V(S(t_i), I(t_i), t_i^+)$, which is the result of $V(S(t_i), I(t_i), t_i)$ of the backward induction process at t_i, with $V(S(t_i), I(t_{i-1}), t_i^-)$ and continue the backward induction process.

- For American options, it is necessary to compare the option value and the exercise value, i.e., $V(S(t), I(t), t) = \max(V(S(t), I(t), t), P(S(t), I(t), t))$, for every node at each time point.

- In practice, with the passage of time, investors can make better predictions about the path-dependent variable $I(t_i)$ because partial information of the stock price is realized. Thus, even not reaching the sampling time point t_i, at which $I(t_i)$ will be updated based on the new sampling stock price, option values still change gradually to reflect the realized partial information.

  ⊙ Therefore, although the updating path of I is discontinuous, the option value V is continuous along realized paths of S and t.

  Figure 9-7



  ⊙ In the backward induction process, the option values V also exhibit the above phenomenon. When $V(S(t_i), I(t_{i-1}), t_i^-)$ is derived based on the mapping (or the updating) rule. The affect of updating $I(t_i)$ at the sampling point t_i will be propagated toward the previous sampling date over the non-sampling time points based on the backward induction process, i.e., the value of $V(S(t), I(t), t)$ for $t \in [t_{i-1}, t_i)$ still changes to reflect the update of $I(t_i)$.

IV. Reset Option

- Here call options with the payoff $\max(S_T - K_T, 0)$ are considered for example, for which the initial strike price is K_0 and the strike price K_t is reset downward to be the prevailing stock price at the sampling points if the prevailing stock price is lower than the prevailing strike price.

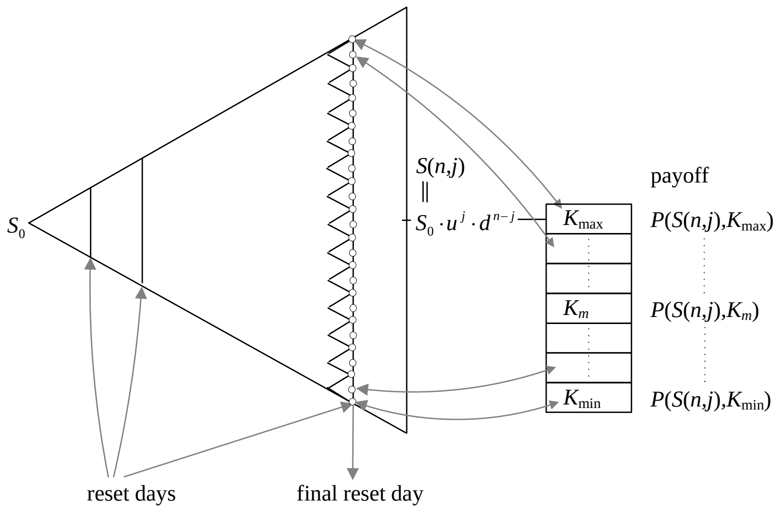
- Reset options are similar to lookback options, but the number of reset time points is much smaller than the number of sampling time points for lookback options. For example, if the time to maturity is 3 months, the reset frequency for reset options may be monthly, but the sampling frequency for lookback options may be daily or continuously.

- The pricing algorithm for reset options is very similar to that for lookback options. It is necessary to construct a K-list for each stock price node to record possible strike prices for that node. Also, you need to develop a backward induction method to reflect the evolution of K appropriately.

  ⊙ The first method to decide a K-list used for all nodes:

  For all nodes on the binomial tree, employ both the stock prices on the final reset day and the stock prices at the time point just prior to the final reset day to construct the K-list. This brute-force method is to capture all possible strike prices for each node on the binomial tree, but this method wastes too much memory space. A modification is to compare possible strike prices with the initial strike price K_0. Only the possible strike prices smaller than the initial strike price K_0 are inserted into the K-list.

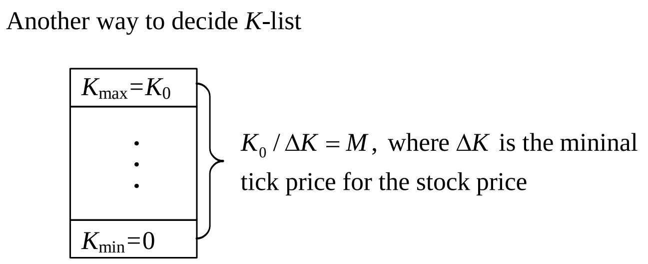
  Figure 9-8



⊙ The second method to decide a K-list used for all nodes:

$K_{\max}$ is assigned to be K_0, $K_{\min}$ is assigned to be 0, and ΔK is the minimal tick movement for the stock price.

Figure 9-9



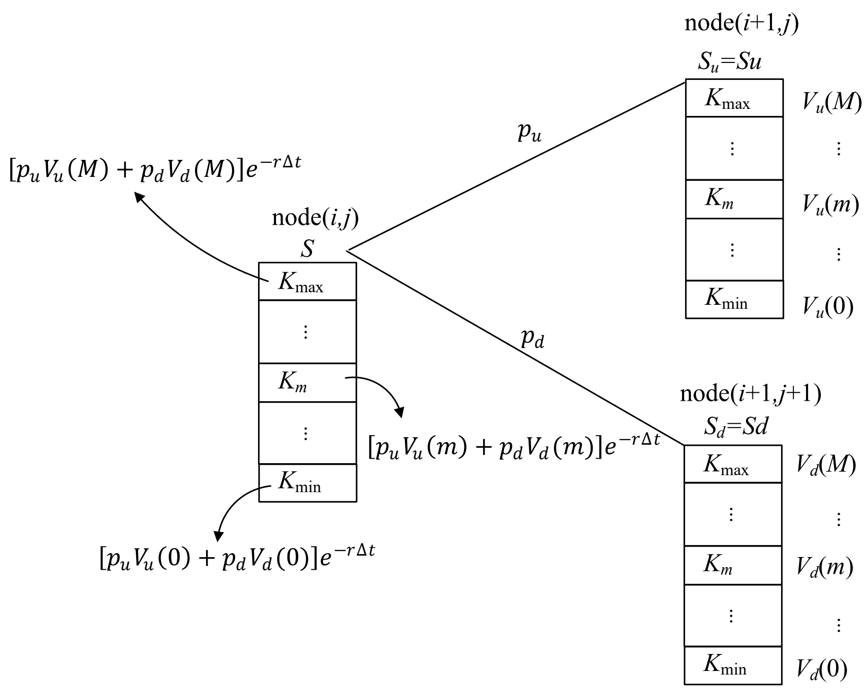
However, this method works only for the reset call, for which the strike price is reset downward.

⊙ The backward induction process:

(i) If $i \neq$ reset day $-$ 1 (i.e., from $i\Delta t$ to $(i+1)\Delta t$, K_t will not change).

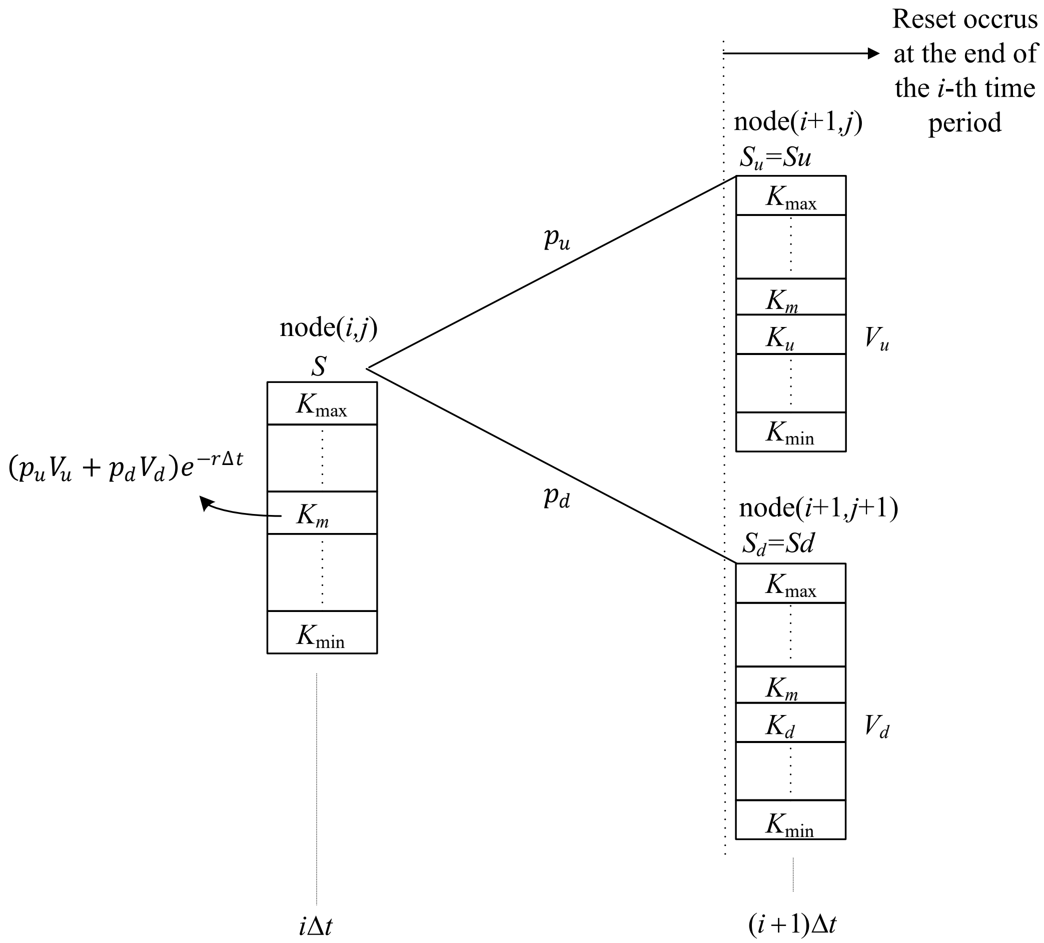
For the option value of each K_m of node(i,j), simply find the option values with the same strike price for the two descendant nodes of node(i,j).

Figure 9-10



(ii) If i = reset day − 1 (i.e., at the next time point $(i+1)\Delta t$, K_t could be updated)

Figure 9-11



First, for each K_m of node(i, j),

K_u=min(K_m,S_u)

K_d=min(K_m,S_d)

Second, find option values V_u and V_d corresponding to K_u and K_d. If there is no such K_u (or K_d), use the option prices corresponding to the strike prices neighboring to K_u (or K_d) to derive the linear interpolation estimate for V_u (or V_d).

⇒ The option value for K_m of node(i, j) is $V(m) = (p_u V_u + p_d V_d)e^{-r\Delta t}$