Degree project

Pricing American and European options under the binomial tree model and its Black-Scholes limit model

Author: Yuankai Yang
Supervisor: Roger Pettersson
Examiner: Astrid Hilbert
Date: 2017-09-28
Course Code: 2MA41E
Subject: Applied Mathematics
Level: Bachelor

Department Of Mathematics
Pricing American and European options under the binomial tree model and its Black-Scholes limit model

Author: Yuankai Yang

Faculty of Technology

Supervisor: Roger Pettersson
Abstract

We consider the \( N \) step binomial tree model of stocks. Call options and put options of European and American type are computed explicitly. With appropriate scaling in time and jumps, convergence of the stock prices and the option prices are obtained as \( N \to \infty \). The obtained convergence is the Black-Scholes model and, for the particular case of European call option, the Black-Scholes formula is obtained. Furthermore, the Black-Scholes partial differential equation is obtained as a limit from the \( N \) step binomial tree model. Pricing of American put option under the Black-Scholes model is obtained as a limit from the \( N \) step binomial tree model.

With this thesis, option pricing under the Black-Scholes model is achieved not by advanced stochastic analysis but by elementary, easily understandable probability computations. Results which in elementary books on finance are mentioned briefly are here derived in more details.

Some important Java codes for \( N \) step binomial tree option prices are constructed by the author of the thesis.

key words: European option, American option, Binomial tree model, Black-Scholes PDE, Black-Scholes option pricing formula.

Acknowledgement

First of all, I would like to say thank you to my supervisor Roger Pettersson for all his support and advice. His encouragement, patience and guidance really helped me much! I still need to thank Jonas Lundberg who was my teacher of Java courses. With this knowledge of Java, I can write Java programs and apply them into different math models.

Contents

1 Introduction to American and European options 4

2 Option pricing under the binomial tree model 5
   2.1 One-step binomial tree model .......................... 5
   2.2 Two-step binomial tree model .......................... 10
   2.3 \( N \)-step binomial tree model ........................ 17
       2.3.1 Examples of \( N \)-step binomial tree model .......... 20

3 The binomial tree model and its Black-Scholes limit model 27
   3.1 Converging results as \( N \to \infty \) ......................... 27
       3.1.1 The Black-Scholes option pricing formula .......... 27
       3.1.2 Derivation of the Black-Scholes Partial Differential Equation 33
   3.2 Numerical comparison of the \( N \)-step binomial model and its Black-Scholes limit .... 37
1 Introduction to American and European options

An option represents a contract which gives the option holder the right, but not the obligation, to purchase or sell the underlying asset at a strike price \( X \), a fixed price of the underlying asset, at or before a specific date, depending on the type of option.

By exercising an option means that the holder of the option buy or sell the stock for the price \( X \).

American options are options that allow holders to exercise the option at or before a specific date, while European options only allow holders to exercise at a specific date. For instance, if we assume the validity period of an option to be \([0, T]\), an American option can be exercised at any moment in the time interval \([0, T]\). For a European option, it is only allowed to exercise at time \( T \) and not before time \( T \).

An owner of the call option has the right but not the obligation to buy the underlying asset at a strike price \( X \). An owner of the put option has the right but not the obligation to sell the underlying asset at the strike price \( X \).

The only difference between American option and European option is the exercise time possibility.

**Notation:** Denote by \( AC \), \( AP \), \( EC \) and \( EP \) options of the type American call, American put, European call and European put, respectively. Let \( C_A \), \( P_A \), \( C_E \) and \( P_E \) be the zero time prices of the \( AC \), \( AP \), \( EC \) and \( EP \), respectively.

By a dividend means that the owner of a stock gets an additional payment in \([0, T]\).

There is a relation between \( C_E \) and \( P_E \), called the put-call-parity [3]:

\[
C_E - P_E = S(0) - Xe^{-rT}.
\]  

(1.1)

**Proposition 1.0.1** Assume that the risk-free interesting rate \( r \) is positive, and there is no dividend. Then the American call option price is equal to the European call option price at any time at or before the time \( T \).

With the ambition to have a self-contained thesis, a proof is included.
Proof of Proposition 1.0.1: The holder has the right to exercise the American call option before the expiry date $T$ with the strike price $X$. We show that it is not optimal to exercise the AC before time $T$. At all time points before $T$, it is better to sell or keep the option but not exercise it. It is sufficient to show that it is not optimal to exercise the AC at time 0. We therefore consider the situation at time 0.

The case $X > S(0)$. Then there is no need to early exercise since the strike price $X$ is more expensive than $S(0)$.

The case $X \leq S(0)$. Then by the put-call-parity (1.1), the fact that European put option prices should always be larger or equal to 0, $P_E \geq 0$, we get:

$$C_E \geq S(0) - X e^{-rT}.$$ 

Besides, clearly, $C_A \geq C_E$ since the American call option is more flexible and provides more opportunities than the European call option. Hence:

$$C_A \geq C_E \geq S(0) - X e^{-rT}.$$ 

Because $r > 0$, $0 < e^{-rT} < 1 \Rightarrow 0 < X e^{-rT} < X$. Therefore:

$$C_A \geq C_E \geq S(0) - X e^{-rT} > S(0) - X. \tag{1.2}$$

The value of exercising the AC is $S(0) - X$ since with the AC you can with the option buy the underlying for the price $X$ and sell it immediately at the market for the price $S(0)$. By (1.2), the value of exercising the AC at time 0, $S(0) - X$, is strictly less than $C_A$, the value of selling or keeping the AC. Therefore it is not optimal to exercise the AC at time 0.

Remark 1.0.2 Assume the risk-free interest rate $r$ is positive and that there is no dividend. Then the American put option price may differ from the European put option price. However, clearly, the value of an AP is always larger or equal to the value of EP since at a same situation, the American put option is more flexible and provides more opportunities than the European put option.

2 Option pricing under the binomial tree model

In [4], a binomial tree model for stock prices was proposed. In finance, the binomial tree model is a very basic model to price options, and it is easily illustrated and understood even with an elementary knowledge of mathematics.

2.1 One-step binomial tree model

A portfolio is a collection of asset holdings. It can for instance be an amount of stocks, a riskfree paper and options.
Assumptions:

1. The stock price at time zero is $S_0 = S(0)$. At time $T$, the stock price $S(T)$ can take the value $S_u = uS_0$ or $S_d = dS_0$ with probabilities $P$ and $1 - P$, respectively.

2. The stock does not pay a dividend.

3. There exists a riskfree paper with continuously compounded interest rate $r$. It means that the value at time $T$ of such riskfree paper is:

$$B(T) = B(0)e^{rT},$$

where $B(0)$ is the initial value of the riskfree paper.

Let $R$ be the simple interest of the riskfree paper in $[0, T]$. It means that $RT$ is the return of the riskfree paper during $[0, T]$, i.e:

$$RT = \frac{B(T) - B(0)}{B(0)} = \frac{B(0)e^{rT} - B(0)}{B(0)} = e^{rT} - 1.$$

4. Non-arbitrage assumption: if the initial value of a portfolio is zero and the future value of the portfolio is known for certainty, then the future value must also be zero.

5. Long and short positions of the riskfree, the stock, and the option is allowed.

Denote by $f_0 = f(0)$, the option price at time zero. Denote by $f_u$, the option price at time $T$ when $S(T) = uS_0$. Denoted by $f_d$, the option price at time $T$ when $S(T) = dS_0$.

<table>
<thead>
<tr>
<th>Stock price</th>
<th>Option price</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S(T) = uS_0$</td>
<td>$f(T) = f_u$</td>
</tr>
<tr>
<td>$S_0, f(0) = f_0$</td>
<td>$1 - P$</td>
</tr>
</tbody>
</table>
Comment of Assumption 4: Assumption 4 implies that $d < e^{rT} < u$. For instance, if $e^{rT} < d < u$ then you at time zero can borrow the riskfree to buy a stock, and at time $T$ sell your stock, pay the loan and keep the profit.

Comment of Assumption 5: By taking a long position at time zero of say a stock means that you buy the stock at time zero and sell it later. By taking a short position in an option at time 0 means that you borrow the option at time 0 and sell it immediately. In the future, here time $T$, you must buy back the option and give it to the original leader.

Now $f_0$, the option price at time zero will be derived for the case of a European option. In the derivation, a suitable portfolio of the riskfree paper and the stock will be constructed:

1. Construct a riskfree portfolio of stock and options, including a **long position** $\Delta$ number of stocks and **short position** in one option at time zero.

2. Then in the one-step binomial tree model, Table 2.1.1 below shows details of this portfolio at time zero and at time $T$.

<table>
<thead>
<tr>
<th>At time 0</th>
<th>At time $T$ when $S(T) = S_u$</th>
<th>At time $T$ when $S(T) = S_d$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Long position in $\Delta$ stocks</strong></td>
<td>Buy $\Delta$ amount of stocks: $-\Delta S_0$</td>
<td>Sell $\Delta$ amount of stocks: $+\Delta S_u$</td>
</tr>
<tr>
<td><strong>Short position in one option</strong></td>
<td>Sell one option: $+f_0$</td>
<td>Buy one option: $-f_u$</td>
</tr>
<tr>
<td><strong>Borrow:</strong> $\Delta S_0 - f_0$</td>
<td>Pay: $-e^{rT}(\Delta S_0 - f_0)$</td>
<td>Pay: $-e^{rT}(\Delta S_0 - f_0)$</td>
</tr>
<tr>
<td><strong>Value:</strong> $-\Delta S_0 + f_0 + \Delta S_0 - f_0 = 0$</td>
<td>$\Delta S_u - f_u - e^{rT}(\Delta S_0 - f_0)$</td>
<td>$\Delta S_d - f_d - e^{rT}(\Delta S_0 - f_0)$</td>
</tr>
</tbody>
</table>

Table 2.1.1: Construction of the riskfree portfolio.

Observe from Table 2.1.1, the value of the riskfree portfolio is zero at time zero. By assumption 4, the time $T$ value of the portfolio is zero for certainty if we can select $\Delta$ so that:

$$\Delta S_u - f_u - e^{rT}(\Delta S_0 - f_0) = \Delta S_d - f_d - e^{rT}(\Delta S_0 - f_0). \quad (2.3)$$

Simplifying (2.3), we get:

$$S_u \Delta - f_u = S_d \Delta - f_d,$$

$$\Delta = \frac{f_u - f_d}{S_u - S_d}. \quad (2.4)$$

The time $T$ value of the portfolio is thus 0, so:

$$\Delta S_u - f_u - e^{rT}(\Delta S_0 - f_0) = 0,$$

$$e^{-rT}(\Delta S_u - f_u) = \Delta S_0 - f_0. \quad (2.5)$$
Putting (2.4) results into (2.5):
\[ S_0 \frac{f_u - f_d}{S_u - S_d} - f_0 = e^{-rT} (S_u \frac{f_u - f_d}{S_u - S_d} - f_u). \] (2.6)

Simplifying (2.6) step by step:

\[ S_0 \frac{f_u - f_d}{S_0(u - d)} - f_0 = e^{-rT} (uS_0 \frac{f_u - f_d}{S_0(u - d)} - f_u), \]
\[ \frac{f_u - f_d}{u - d} - f_0 = e^{-rT} \left( u \frac{f_u - f_d}{u - d} - f_u \right). \]

Multiplying \( u - d \) at both sides yields:

\[ f_u - f_d - f_0(u - d) = e^{-rT} (uf_u - uf_d - f_u u + f_d), \]
\[ f_u - f_d - f_0(u - d) = e^{-rT} (-uf_d + f_u d), \]
\[ f_u - f_d - e^{-rT} (-uf_d + f_u d) = f_0(u - d), \]
\[ e^{-rT}(e^T f_u - e^T f_d - e^T f_a u + e^T f_a d) = f_0(u - d), \]
\[ e^{-rT}[f_a(e^T d) + f_a(u - e^T)] = f_0(u - d), \]

and dividing both sides by \( u - d \):

\[ f_0 = e^{-rT} \left[ f_u \frac{e^T - d}{u - d} + f_d \frac{u - e^T}{u - d} \right]. \] (2.7)

On the right hand side of (2.7), we observe:

\[ \frac{e^T - d}{u - d} + \frac{u - e^T}{u - d} = \frac{e^T - d + u - e^T}{u - d} = \frac{u - d}{u - d} = 1. \]

Hence (2.7) can be re-written as:

\[ f_0 = e^{-rT} (pf_u + (1 - p)f_d), \] (2.8)

where

\[ p = \frac{e^T - d}{u - d}, \quad 1 - p = \frac{u - e^T}{u - d}, \quad e^T = 1 + RT. \] (2.9)
The formula (2.8) above gives the option price of a European option under the one-step binomial tree model.

Example 2.1.1 (European call option):

1. Assume the price of a stock at time 0 is $30. After six months, the price of the stock will have possibility increase to $36 or decrease to $24.

2. Assume the strike price of a European call option is $32, and the exercise day is one half year later ($T = 0.5$).

3. Assume the $r$ is 10%.

Compute the European call option price at time zero.

Solution:

- Stock price at time 0: $S_0 = $30
- Stock price at time T: $S_u = $36, $S_d = $24
- Option price at time T: $f_u = \max(36-32, 0) = $4, $f_d = \max(24-32, 0) = $0

Putting $u$ and $d$ into (2.9):

\[
e^{0.1 \times 0.5} = 1.05127, \quad p = \frac{1.05127 - 0.8}{1.2 - 0.8} = 0.6281.
\]

By (2.8):

\[
f_0 = 0.95123(0.6281 \cdot 4 + 0.3819 \cdot 0) \approx 2.38987 \approx 2.39.
\]

The value of this European call option is $2.39.

Example 2.1.2 (European put option):

1. Assume the price of a stock at time 0 is $20. After three months, the price will increase to $uS_0$ ($u=1.5$), or decrease to $dS_0$ ($d=0.66667$).

2. Assume the strike price of the European put option is $18, and the exercise day is three months later ($T = 0.25$).

3. Assume the $r$ is 10%.

Compute the European put option price at time zero.
Solution:

<table>
<thead>
<tr>
<th>Stock price at time 0</th>
<th>Stock price at time T</th>
<th>Option price at time T</th>
</tr>
</thead>
<tbody>
<tr>
<td>( S_u = 20u ) = $30</td>
<td>( f_u = \max(18-20,0) ) = $0</td>
<td></td>
</tr>
<tr>
<td>( S_0 = $20 )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( S_d = 20d = $13.33334 )</td>
<td>( f_d = \max(18-13.33334,0) ) = $4.67</td>
<td></td>
</tr>
</tbody>
</table>

Putting \( u \) and \( d \) into (2.9):

\[
e^{0.1 \cdot 0.25} = 1.0253, \]
\[
p = \frac{1.0253 - 0.66667}{1.5 - 0.66667} = 0.43.
\]

By (2.8):

\[
e^{-0.1 \cdot 0.25} = 0.97531, \]
\[
f_0 = 0.97531 \cdot (0.43 \cdot 0 + 0.57 \cdot 4.67) \approx 2.67.
\]

The value of this European put option is $2.67.

\[ \square \]

2.2 Two-step binomial tree model

In a two-step binomial tree model, we still assume the stock price is \( S_0 \) at time zero. The time of every step is \( \frac{T}{2} \). From time zero to \( \frac{T}{2} \) and time \( \frac{T}{2} \) to \( T \), we assume

1. In every time step, the stock price will be \( u \) times itself with chance \( P \).
2. In every step, the stock price will be \( d \) times itself with chance \( 1 - P \).
3. The stock does not pay a dividend.
4. \( f_{uu}, f_{ud}, \) and \( f_{dd} \) are possible option price at time \( T \).
5. Non-arbitrage assumption: if the initial value of a portfolio is zero and the future value of the portfolio is known for certainty, then the future value must also be zero.
6. Long and short position of the riskfree, the stock, and the option is allowed.
Comment of Assumption 5: Assumption 5 implies that $d < e^{rT/2} < u$.

\[
\begin{align*}
S(T) &= S_{uu} = u^2 S_0, \quad f(T) = f_{uu} \\
S(T/2) &= u S_0 = S_u, \quad f(T/2) = f_u \\
S(0) &= S_0, \quad f(0) = f_0 \\
S(T/2) &= d S_0 = S_d, \quad f(T/2) = f_d \\
S(T) &= S_{dd} = d^2 S_0, \quad f(T) = f_{dd}
\end{align*}
\]

A two-step binomial tree model is composed of three one-step binomial tree models:

1. $f_u, f_{uu}$ and $f_{ud}$.
2. $f_d, f_{ud}$ and $f_{dd}$.
3. $f_0, f_u$ and $f_d$.

Putting 1, 2 and 3 into (2.8) and (2.9) with $T$ replaced by $T/2$:

\[
\begin{align*}
f_u &= e^{-rT/2}(pf_{uu} + (1 - p)f_{ud}), \quad (2.10) \\
f_d &= e^{-rT/2}(pf_{ud} + (1 - p)f_{dd}), \quad (2.11) \\
f_0 &= e^{-rT/2}(pf_u + (1 - p)f_d), \quad (2.12)
\end{align*}
\]

where

\[
\begin{align*}
p &= \frac{e^{rT/2} - d}{u - d}, \quad (2.13) \\
1 - p &= \frac{u - e^{rT/2}}{u - d}.
\end{align*}
\]
Putting (2.10), (2.11) and (2.13) into (2.12):

\[ f_0 = e^{-rT} \left[ \frac{(e^{rT/2} - d)^2}{(u - d)^2} f_{uu} + 2 \frac{(e^{rT/2} - d)}{(u - d)} f_{ud} + \frac{(u - e^{rT/2})^2}{(u - d)^2} f_{dd} \right]. \]

After simplifying:

\[ f_0 = e^{-rT} \left[ p^2 f_{uu} + 2p(1-p)f_{ud} + (1-p)^2 f_{dd} \right]. \]  

(2.14)

The formula (2.14) above gives the option price for the two-step binomial tree model.

**Example 2.2.1 (European call option):**

1. Assume the price of a stock at time zero is $50, there are two steps of time, every time step is six month \((T/2=0.5)\), \(u\) is 1.2 and \(d\) is 0.8.

2. Assume the strike price of European call option is $50.

3. Assume the \(r\) is 10%.

Compute the European call option price at time zero.

**Solution:**

It is a European option, so we have to exercise it at expiry day. We have:

\[ f_{uu} = \max \{ S_0 u^2 - 50, 0 \} = 22 \]
\[ f_{ud} = \max \{ S_0 ud - 50, 0 \} = 0 \]
\[ f_{dd} = \max \{ S_0 d^2 - 50, 0 \} = 0. \]

By (2.13):

\[ e^{rT/2} = e^{0.1 \cdot 0.5} = 1.05127 \]
\[ p = \frac{1.05127 - 0.8}{1.2 - 0.8} = 0.6282. \]

By (2.10), (2.11) and (2.12):

\[ e^{-rT/2} = e^{-0.1 \cdot 0.5} = 0.9512 \]
\[ f_u = 0.9512 (0.6282 \cdot 22 + 0.3718 \cdot 0) \approx 13.146 \]
\[ f_d = 0.9512 (0.6282 \cdot 0 + 0.3718 \cdot 0) = 0 \]
\[ f_0 = 0.9512 (0.6282 \cdot 13.146 + 0.3718 \cdot 0) \approx 7.8553. \]
As an alternative we can directly use the two-step binomial option pricing formula (2.14):

\[ e^{-rT} = e^{-2 \cdot 0.1 \cdot 0.5} = 0.9048 \]

\[ f_0 = 0.9048 \cdot (0.6283^2 \cdot 22 + 0 + 0) \approx 7.85. \]


\[ \begin{array}{ccc}
0 & \quad & T \\
\quad & S_{uu} = 72, & f_{uu} = 22 \\
& S_u = 60, & f_u = 13.146 \\
S_0 = 50, & f_0 = 7.8553 \quad & S_{ud} = 48, \quad f_{ud} = 0 \\
& S_d = 40, & f_d = 0 \quad & S_{dd} = 32, \quad f_{dd} = 0
\end{array} \]

The value of this European call option is $7.86.

\[ \square \]

**Example 2.2.2 (American call option)**:

1. Assume the price of a stock at time zero is $50, there are two time steps, every time step is six month ($T/2=0.5$), $u$ is 1.2 and $d$ is 0.8.

2. Assume the strike price of American call option is $50.

3. Assume the $r$ is 10%.

Compute the American call option price at time zero.

**Solution:**

American option can be exercised before the expiry day. Therefore we need to consider whether we should exercise it early or not, and (2.14) may be used.
For the American option, we have to start from the last step, and compare non-early exercise option price with early exercise option price of every step. This example has the same data as in Example 2.2.1:

\[ f_{uu} = \max\{S_0u^2 - 50, 0\} = 22 \]
\[ f_{ud} = \max\{S_0ud - 50, 0\} = 0 \]
\[ f_{dd} = \max\{S_0d^2 - 50, 0\} = 0 \]
\[ e^{rT/2} = e^{0.1\cdot0.5} = 1.05127 \]
\[ p = \frac{1.05127 - 0.8}{1.2 - 0.8} = 0.6282, \]

and

\[ e^{-rT/2} = e^{-0.1\cdot0.5} = 0.9512 \]
\[ f_{u}^{NE} = f_{u} = 0.9512(0.6282 \cdot 22 + 0.3718 \cdot 0) \approx 13.146 \]
\[ f_{d}^{NE} = f_{d} = 0.9512(0.6282 \cdot 0 + 0.3718 \cdot 0) = 0. \]

At point B, if early exercise, \( f_{E}^{u} = \max\{60 - 50, 0\} = 10 \) which is smaller than the non-early exercise value \( f_{u} = f_{u}^{NE} = 13.146 \). So we should do non-early exercise at B, and \( f_{u} \) is equal to 13.146.

\[ S_{uu} = 72, \quad f_{uu} = 22 \]

\[ B: S_u = 60, \quad f_{E}^{u} = 10 < \]
\[ f_{u}^{NE} = 13.146 \]

\[ A: S_0 = 50 \]
\[ S_{ud} = 48, \quad f_{ud} = 0 \]

\[ C: S_d = 40, \quad f_{E}^{d} = 0 \]
\[ 0 \leq f_{d}^{NE} = 0 \]
\[ S_{dd} = 32, \quad f_{dd} = 0 \]
At point C, if early exercise, $f_d^E = \max\{40-50, 0\} = 0$ which is equal to the non-early exercise $f_d = f_d^N = 0$ value. It means that it does not matter whether we do early exercise or not at C, and $f_d = 0$.

Next:

**B:** $S_u = 60$, $f_u = 13.146$

**A:** $S_0 = 50$, $f_0$

**C:** $S_d = 40$, $f_d = 0$

By (2.12):

$$f_0 = 0.9512(0.6282 \cdot 13.146 + 0.3718 \cdot 0) \approx 7.8553.$$  

The value of this American call option is $7.86$. Compare with Example 2.2.1 and Example 2.2.2, where the price of the European call option and the American call option are both $7.86$ as they should according to Proposition 1.0.1.

**Example 2.2.3 (American put option):**

1. Assume the price of a stock at time zero is $50$, there are two time steps, every step is six month ($T/2 = 0.5$), $u$ is 1.2 and $d$ is 0.8.
2. Assume the strike price of American put option is $52$, within one year to exercise it.
3. Assume the $r$ is 10%.

"Compute the American put option price at time zero."

**Solution:**

It is an American put option, so we have to start from the last time step, and compare non-early exercise option price with early exercise option price of every step:

$$f_{uu} = \max\{52 - 72, 0\} = 0$$
$$f_{ud} = \max\{52 - 48, 0\} = 4$$
$$f_{dd} = \max\{52 - 32, 0\} = 20.$$
By (2.13):
\[ e^{rT/2} = e^{0.1\cdot0.5} = 1.05127 \]
\[ p = \frac{1.05127 - 0.8}{1.2 - 0.8} = 0.6282. \]

By (2.10) and (2.11):
\[ e^{-rT/2} = e^{-0.1\cdot0.5} = 0.9512 \]
\[ f_u^{NE} = f_u = 0.9512(0.6282 \cdot 0 + 0.3718 \cdot 4) \approx 1.4148 \]
\[ f_d^{NE} = f_d = 0.9512(0.6282 \cdot 4 + 0.3718 \cdot 20) \approx 9.4633. \]

We have:

At point B, if early exercise, \( f_u^E = \max\{52-60,0\} = 0 \) which is less than the non-early exercise \( f_u = f_u^{NE} = 1.4148 \) value. So we should do non-early exercise at B, and \( f_u = 1.4148 \).

At point C, if early exercise, \( f_d^E = \max\{52-40,0\} = 12 \) which is larger than the non-early exercise \( f_d = f_d^{NE} = 9.4633 \) value. If we do early exercise then more profit will be earned than non-early exercise. So we do early exercise at C, and \( f_d = 12 \).
Then:

\[ B: S_u = 60, \quad f_u = 1.4148 \]

\[ A: S_0 = 50, \quad f_0 \]

\[ C: S_d = 40, \quad f_d = 12 \]

By (2.12):

\[ f_0 = 0.9512(0.6282 \cdot 1.4148 + 0.3718 \cdot 12) \approx 5.09. \]

The value of this American put option is $5.09. We can see that the American put option should be early exercised at the first step and the European put option should be exercised at the last step. Hence the American put option price differs from the European put option price which is in line with Remark 1.0.2.

\[ \square \]

2.3 \textit{N-step binomial tree model}

Based on the one-step and the two-step binomial tree model, we can deduce the \textit{N}-step binomial tree model.

For an \textit{N}-step binomial tree model for European option, assume:

1. The stock price at time zero is \( S_0 \). In every time step, the stock price will be \( u \) times itself with chance \( P \) or be \( d \) times itself with chance \( 1 - P \).

2. The stock does not pay a dividend.

3. Non-arbitrage assumption: if the initial value of a portfolio is zero and the future value of the portfolio is known for certainty, then the future value must also be zero.

4. Long and short position of the riskfree, the stock, and the option is allowed.

\textbf{Comment of assumption 3:} Assumption 3 implies that \( d < e^{rT/N} < u \).
At time $T$, the price of the stock is $S_0u^i d^{N-i}$ at time $T$ with $i$ numbers of $u$-jumps and $N-i$ numbers of $d$-jumps. Therefore, at time $T$, the value of a European call option will then be $\max\{S_0u^i d^{N-i} - X, 0\}$. The value of a European put option will then be $\max\{X - S_0u^i d^{N-i}, 0\}$.

By similar arguments as for the time zero option price for the two-steps binomial tree model (2.14), here the time zero option price can be written as:

$$f_0 = e^{-rT}\sum_{i=0}^{N}(\binom{N}{i})p^i(1-p)^{N-i}f_{u^i d^{N-i}},$$

where

$$p = \frac{e^{rT/N} - d}{u - d},$$  \hspace{1cm} (2.15)

and $f_{u^i d^{N-i}}$ is the option price at time $T$ if the stock price at time $T$ is $S_0u^i d^{N-i}$.

In more details, for European call options:

$$f_0 = f^{\text{call}}_0 = e^{-rT}\sum_{i=0}^{N}(\binom{N}{i})p^i(1-p)^{N-i}\max\{S_0u^i d^{N-i} - X, 0\}. \hspace{1cm} (2.16)$$

For European put options:

$$f_0 = f^{\text{put}}_0 = e^{-rT}\sum_{i=0}^{N}(\binom{N}{i})p^i(1-p)^{N-i}\max\{X - S_0u^i d^{N-i}, 0\}. \hspace{1cm} (2.17)$$

The formulas (2.16) and (2.17) above give the option price for the $N$-step binomial tree model.

The way of pricing American option under an $N$-step binomial tree model

We have mentioned how to price American options under a two-step binomial tree model. By the same argument for the $N$-step binomial tree model, we still have to start from last step, and compare non-early exercise option price with early exercise option price of every step.

Example 2.3.1 (Pricing American option under a three-step binomial tree model) :
Compute the American option price at time zero.

Solution:

At points G, H, I and J, we have option prices $f_{uuu}$, $f_{uud}$, $f_{udd}$ and $f_{ddd}$, respectively. By using (2.8) for the one-step binomial tree model with $T$ replaced by $T/3$, we get the non-early exercise option price $f_{NEu}$, $f_{NEud}$ and $f_{NEddd}$ at point D, E and F, respectively, and judge whether we should do early exercise or not at D, E and F.

The American call option (put option) may be early exercised. Observe by Proposition 1.0.1, AC should not be early exercised. Therefore we can get $f_{uuu}$, $f_{ud}$ and $f_{dd}$ at D, E and F, respectively:

1. $f_{uu} = \max\{S_0u^2 - X, 0\}$ or $\max\{X - S_0u^2, 0\}$,
2. $f_{ud} = \max\{S_0ud - X, 0\}$ or $\max\{X - S_0ud, 0\}$,
3. $f_{dd} = \max\{S_0d^2 - X, 0\}$ or $\max\{X - S_0d^2, 0\}$,

compare $f_{uu}$, $f_{ud}$, $f_{dd}$ with $f_{Euu}$, $f_{Eud}$, $f_{Edd}$, respectively:

1. $f_{uu} = \max\{f_{NEu}, f_{Euu}\}$,
2. $f_{ud} = \max\{f_{NEud}, f_{Eud}\}$,
3. $f_{dd} = \max\{f_{NEddd}, f_{Edd}\}$.

Now we get $f_{uu}$, $f_{ud}$ and $f_{dd}$. By the same argument, we can get $f_{Eu}$ and $f_{Ed}$ at point B and C, respectively.
The American call option (put option) may be early exercised, so we get $f^E_u$ and $f^E_d$ at B and C, respectively:

1. $f^E_u = \max\{S_0u - X, 0\}$ or $\max\{X - S_0u, 0\}$,
2. $f^E_d = \max\{S_0d - X, 0\}$ or $\max\{X - S_0d, 0\}$,

compare $f^{NE}_u, f^{NE}_d$ with $f^E_u, f^E_d$, respectively:

1. $f_u = \max\{f^{NE}_u, f^E_u\}$,
2. $f_d = \max\{f^{NE}_d, f^E_d\}$.

By the same argument, we finally get $f_0$ at point A.

For $N$-steps, recall formula (2.9) with $T$ replaced by $T/N$:

$$e^{rT/N} = 1 + RT/N, \quad e^{-rT/N} = \frac{1}{1 + RT/N}. \quad (2.18)$$

By the same argument as in Example 2.3.1, we have as in [3, pp. 181-184]:

$$D^A(N) = f(S(N)),$$

$$D^A(N-1) = \max\{f(S(N-1)), \frac{1}{1 + RT/N}[p \cdot f_N(S(N-1)u) + (1-p) \cdot f_N(S(N-1)d)]\},$$

$$D^A(N-2) = \max\{f(S(N-2)), \frac{1}{1 + RT/N}[p \cdot f_{N-1}(S(N-2)u) + (1-p) \cdot f_{N-1}(S(N-2)d)]\},$$

$$\vdots$$

$$\vdots$$

$$D^A(1) = \max\{f(S(1)), \frac{1}{1 + RT/N}[p \cdot f_2(S(1)u) + (1-p) \cdot f_2(S(1)d)]\},$$

$$D^A(0) = \max\{f(S(0)), \frac{1}{1 + RT/N}[p \cdot f_1(S(0)u) + (1-p) \cdot f_1(S(0)d)]\},$$

where $D^A(N)$ is the option price at time step $N$. Inside $\max\{}$, the first part is the early exercise option value, the second part is the non-early exercise option value.

### 2.3.1 Examples of $N$-step binomial tree model

Based on the algorithms of pricing option price in the $N$-steps option pricing binomial tree model above, some Java programs are written by the author to compute option price.

Example 2.3.2  $N$ steps binomial tree model, with initial price 60 Kr, strike price 50 Kr, $u = 1.2$, $d = 0.8$, $T = 1$, and $r = 0.1$. Compute European and American time 0 option prices by using different $N$.

Solution: Table 2.3.2, Figure 2.3.1 and Figure 2.3.2.

![Graphical representation of option pricing](image)

Table 2.3.2: $S_0 = 60$, $X = 50$, $u = 1.2$, $d = 0.8$, $T = 1$, $r = 0.1$. 

<table>
<thead>
<tr>
<th>$N$</th>
<th>AC</th>
<th>AP</th>
<th>EC</th>
<th>EP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>15.000000000000005</td>
<td>0.4545454545454539</td>
<td>15.000000000000005</td>
<td>0.4545454545454539</td>
</tr>
<tr>
<td>2</td>
<td>15.488165680473367</td>
<td>1.715976331360945</td>
<td>15.488165680473367</td>
<td>1.715976331360945</td>
</tr>
<tr>
<td>3</td>
<td>17.02770971098655</td>
<td>2.605652713906879</td>
<td>17.02770971098655</td>
<td>2.343409083280179</td>
</tr>
<tr>
<td>4</td>
<td>17.78856054174432</td>
<td>3.234512681716564</td>
<td>17.78856054174432</td>
<td>3.0860927817320762</td>
</tr>
<tr>
<td>5</td>
<td>18.754112458061797</td>
<td>4.343553721254898</td>
<td>18.754112458061797</td>
<td>4.04652949557591</td>
</tr>
<tr>
<td>10</td>
<td>21.57220027672573</td>
<td>7.353280091007527</td>
<td>21.57220027672573</td>
<td>6.836548011374892</td>
</tr>
<tr>
<td>50</td>
<td>35.33595824994206</td>
<td>21.36105370734357</td>
<td>35.33595824994206</td>
<td>20.58234754122068</td>
</tr>
<tr>
<td>80</td>
<td>41.04245961398358</td>
<td>27.1953648621172</td>
<td>41.04245961398358</td>
<td>26.28715586790272</td>
</tr>
<tr>
<td>100</td>
<td>43.80197141368239</td>
<td>30.05282698331222</td>
<td>43.80197141368239</td>
<td>29.04610295871666</td>
</tr>
<tr>
<td>150</td>
<td>48.7199339201098</td>
<td>35.20219563107752</td>
<td>48.7199339201098</td>
<td>33.9613711137021</td>
</tr>
<tr>
<td>200</td>
<td>52.01727187879459</td>
<td>38.69220694933731</td>
<td>52.01727187879459</td>
<td>37.26027346462027</td>
</tr>
<tr>
<td>250</td>
<td>54.2250587444314</td>
<td>41.08820474085987</td>
<td>54.2250587444314</td>
<td>39.46778138148589</td>
</tr>
</tbody>
</table>
Figure 2.3.1: $S_0 = 60$, $X = 50$, $u = 1.2$, $d = 0.8$, $T = 1$, $r = 0.1$. 

The option price with different steps $N$. 

option price

N (steps)
Example 2.3.3  $N$ steps binomial tree model, with initial price $50$, strike price $52$, $u = 1.2$, $d = 0.8$, $T = 1$, and $r = 0.1$. Compute European and American time 0 option prices by using different $N$.

Solution: Table 2.3.3 and Figure 2.3.3.
<table>
<thead>
<tr>
<th>N</th>
<th>AC</th>
<th>AP</th>
<th>EC</th>
<th>EP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5.454545454545457</td>
<td>2.7272727272727235</td>
<td>5.454545454545457</td>
<td>2.7272727272727235</td>
</tr>
<tr>
<td>2</td>
<td>7.086167800453516</td>
<td>5.136054421768705</td>
<td>7.086167800453516</td>
<td>4.251700680272106</td>
</tr>
<tr>
<td>3</td>
<td>8.347319660300093</td>
<td>6.0869893592024376</td>
<td>8.347319660300093</td>
<td>5.475647007485472</td>
</tr>
<tr>
<td>4</td>
<td>9.517982500997066</td>
<td>7.337507931491734</td>
<td>9.517982500997066</td>
<td>6.627416030584333</td>
</tr>
<tr>
<td>5</td>
<td>10.0772310518658</td>
<td>7.881095156061486</td>
<td>10.0772310518658</td>
<td>7.175225216342205</td>
</tr>
<tr>
<td>10</td>
<td>13.86264439306841</td>
<td>11.633951994671637</td>
<td>13.86264439306841</td>
<td>10.93756603710354</td>
</tr>
<tr>
<td>20</td>
<td>18.29565572368932</td>
<td>16.2271170453575</td>
<td>18.29565572368932</td>
<td>15.35893659423488</td>
</tr>
<tr>
<td>80</td>
<td>32.160381283160525</td>
<td>30.33448800019209</td>
<td>32.160381283160525</td>
<td>29.214865386079452</td>
</tr>
<tr>
<td>100</td>
<td>34.776879999999958</td>
<td>33.03777032740004</td>
<td>34.776879999999958</td>
<td>31.830284806684464</td>
</tr>
<tr>
<td>150</td>
<td>39.49347371831697</td>
<td>37.956056871713464</td>
<td>39.49347371831697</td>
<td>36.54658717045054</td>
</tr>
<tr>
<td>200</td>
<td>42.52269921936338</td>
<td>41.188885951421297</td>
<td>42.52269921936338</td>
<td>39.5742086862197</td>
</tr>
<tr>
<td>250</td>
<td>44.62149697424877</td>
<td>43.46652086618817</td>
<td>44.62149697424877</td>
<td>41.67398350157303</td>
</tr>
</tbody>
</table>

Table 2.3.3: \( S_0 = 50, \ X = 52, \ u = 1.2, \ d = 0.8, \ T = 1, \ r = 0.1. \)

Figure 2.3.3: \( S_0 = 50, \ X = 52, \ u = 1.2, \ d = 0.8, \ T = 1, \ r = 0.1. \)
Example 2.3.4 We consider a case when the difference between initial price and strike price is very huge. We consider $N$ steps binomial tree models, with initial price $\$ 200$, strike price $\$ 30$, $u = 1.2$, $d = 0.8$, $T = 1$, and $r = 0.1$. Compute European and American time 0 option prices by using different $N$.

Solution: Table 2.3.4 and Figure 2.3.4.

<table>
<thead>
<tr>
<th>$N$</th>
<th>AC</th>
<th>AP</th>
<th>EC</th>
<th>EP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>172.72727272727272</td>
<td>0.0</td>
<td>172.72727272727272</td>
<td>0.0</td>
</tr>
<tr>
<td>2</td>
<td>172.78911564625852</td>
<td>0.0</td>
<td>172.78911564625852</td>
<td>0.0</td>
</tr>
<tr>
<td>3</td>
<td>172.81058037662385</td>
<td>0.0</td>
<td>172.81058037662385</td>
<td>0.0</td>
</tr>
<tr>
<td>4</td>
<td>172.82148065600734</td>
<td>0.0</td>
<td>172.82148065600734</td>
<td>0.0</td>
</tr>
<tr>
<td>5</td>
<td>172.82807570510252</td>
<td>0.0</td>
<td>172.82807570510252</td>
<td>0.0</td>
</tr>
<tr>
<td>10</td>
<td>172.84590394089437</td>
<td>0.004512581683910533</td>
<td>172.84590394089437</td>
<td>0.004512581683910533</td>
</tr>
<tr>
<td>20</td>
<td>173.15387042009950</td>
<td>0.3117366031761193</td>
<td>173.15387042009950</td>
<td>0.3057575480990097</td>
</tr>
<tr>
<td>50</td>
<td>176.09954696711870</td>
<td>3.3051121978822955</td>
<td>176.09954696711870</td>
<td>3.2473805418916126</td>
</tr>
<tr>
<td>80</td>
<td>179.85206265695947</td>
<td>7.1350687464144777</td>
<td>179.85206265695947</td>
<td>6.99880408643831</td>
</tr>
<tr>
<td>100</td>
<td>182.13714411600630</td>
<td>9.4711537861688152</td>
<td>182.13714411600630</td>
<td>9.2835933429960190</td>
</tr>
<tr>
<td>150</td>
<td>186.82492913297062</td>
<td>14.2858909531287653</td>
<td>186.82492913297062</td>
<td>13.9709561245861840</td>
</tr>
<tr>
<td>200</td>
<td>190.34048664150404</td>
<td>17.92554267166686</td>
<td>190.34048664150404</td>
<td>17.4862875930011440</td>
</tr>
<tr>
<td>250</td>
<td>192.84224482785450</td>
<td>20.54690157867447</td>
<td>192.84224482785450</td>
<td>19.9879097870134070</td>
</tr>
</tbody>
</table>

Table 2.3.4: $S_0 = 200$, $X = 30$, $u = 1.2$, $d = 0.8$, $T = 1$, $r = 0.1$. 
Discussion of the results from Example 2.3.2 to Example 2.3.4:

From Table 2.3.2 to Table 2.3.4, we can see that when $N$ has the same value, the American call is always equal to the European call as in Proposition 1.0.1, while the American put differ from the European put as in Remark 1.0.2.

More precisely, observe from Figure 2.3.1 to Figure 2.3.4, we can see that the American call overlaps with the European call, because then they cover each other and the color of the common curve changes. The value of the red straight line is 0 which indicates that the value difference between the American call and the European call is always 0. These two results indicate that American call is equal to European call as in Proposition 1.0.1. On the contrary, the curves of the American put and the European put do not cover each other and the American put curve is always above the European put curve which is in line with Remark 1.0.2.
3 The binomial tree model and its Black-Scholes limit model

In this chapter we consider the limiting model of the $N$-step binomial tree model where $u$ and $d$ depend on $T$ and $N$ in a careful way. The limiting model is recognised as the Black-Scholes model. The limiting European pricing formulas for the $N$-step binomial tree model will be the Black-Scholes formulas. Also, the Black-Scholes PDE is derived from the binomial tree model.

3.1 Converging results as $N \to \infty$

3.1.1 The Black-Scholes option pricing formula

Here we consider the binomial tree model suggested in [6], but here with detail computations and not only for derivation of Black-Scholes PDEs:

\begin{align*}
    u &= u_N = 1 + \sigma \sqrt{T/N}, \quad (3.19) \\
    d &= d_N = 1 - \sigma \sqrt{T/N}, \quad (3.20) \\
    d < e^{rT/N} < u.
\end{align*}

Recall formula (2.15). We have:

\begin{align*}
    p &= p_N = \frac{e^{rT/N} - d}{u - d} = \left[ e^x \approx 1 + x \right] \\
    &\approx \frac{rT/N + \sigma \sqrt{T/N}}{2\sigma \sqrt{T/N}} \\
    &= \frac{1}{2} + \frac{r}{2\sigma \sqrt{T/N}} \\
    q &= q_N = 1 - p \\
    &\approx \frac{1}{2} - \frac{r}{2\sigma \sqrt{T/N}}.
\end{align*}

Let $Y_N$ be the number of times in $[0, T]$ the binomial tree stock price take $u$-jumps under the constructed probability $p_N$. Then $Y_N$ is a binomial random variable with parameters $p_N$ and $N$ ($Y_N \sim \text{Binomial}(N, p_N)$). The stock price at time $T$ for the binomial tree model with $N$ steps under the constructed probability $p_N$ can be represented as $S_T^N = S_0 u_Y^N d_N^{N-Y_N}$.

**Theorem 3.1.1** Under the constructed probability $p_N$, $\ln S_T^N \to \ln S_0 + (r - \sigma^2/2)T + \sigma W_T$ with probability one as $N \to \infty$, where $W_T \sim \mathcal{N}(0, 1)$.

Theorem 3.1.1 implies that $S_T^N = S_0 u_Y^N d_N^{N-Y_N} \approx S_T = S_0 e^{(r - \sigma^2/2)T + \sigma W_T}$, for $N$ large. This in fact the stock price of the Black-Scholes model at time $T$ under a certain constructed probability.
Proof of Theorem 3.1.1: It is sufficient to show that the characteristic function of \( \ln S_T^N \) converges to the characteristic function of \( \ln S_0 + (r - \frac{\sigma^2}{2}) T + \sigma W_T \).

For \( \theta \in \mathbb{R} \),

\[
\Psi_{\ln S_T^N}(\theta) = E[e^{i\theta \ln S_T^N}] = E[e^{i\theta \ln(S_0u^N d_N^{N-Y_N})}]
\]

\[
= E[e^{i\theta \ln(S_0(\frac{u}{d_N})^N d_N^N)}]
\]

\[
= E[e^{i\theta \ln S_0 + i\theta Y_N \ln \frac{u}{d_N} + i\theta \ln d_N}]
\]

\[
= e^{i\theta \ln S_0 + i\theta N \ln d_N} E[e^{i\theta \ln(\frac{u}{d_N}) Y_N}].
\]

Observe

\[
\Psi_{Y_N}(\theta) = E[e^{i\theta Y_N}] = \sum_{k=0}^{N} e^{i\theta k} \binom{N}{k} p_N^k (1 - p_N)^{N-k}
\]

\[
= \sum_{k=0}^{N} \binom{N}{k} (e^{i\theta} p_N)^k (1 - p_N)^{N-k}
\]

\[
= (e^{i\theta} p_N + (1 - p_N))^N
\]

\[
= (1 + (e^{i\theta} - 1) p_N)^N.
\]

Hence

\[
\Psi_{\ln S_T^N}(\theta) = e^{i\theta \ln S_0 + i\theta N \ln d_N} \Psi_{Y_N}(\theta \ln(\frac{u}{d_N}))
\]

\[
= e^{i\theta \ln S_0 + i\theta N \ln d_N} (1 + (e^{i\theta \ln \frac{u}{d_N}} - 1) p_N)^N
\]

\[
= e^{i\theta \ln S_0 + i\theta N \ln d_N + \ln(1 + (e^{i\theta \ln \frac{u}{d_N}} - 1) p_N)}.
\]

Note: \( \frac{u}{d_N} \to 1 \) as \( N \to \infty \), i.e. \( \theta \ln \frac{u}{d_N} \to 0 \) as \( N \to \infty \). Note also \( p_N \to \frac{1}{2} \) as \( N \to \infty \).

Since \( e^x \approx 1 + x + \frac{x^2}{2} \) for small \( x \),

\[
\Psi_{\ln S_T^N}(\theta) \approx e^{i\theta \ln S_0 + i\theta N \ln d_N + \ln(1 + (e^{i\theta \ln \frac{u}{d_N}} + \frac{1}{2}(\theta)^2 \ln^2 \frac{u}{d_N} p_N - \frac{(\theta)^2}{2} \ln^2 \frac{u}{d_N} p_N^2))}
\]

Since \( \ln(1 + x) \approx x - \frac{x^2}{2} \) for small \( x \),

\[
\Psi_{\ln S_T^N}(\theta) \approx e^{i\theta \ln S_0 + i\theta N \ln d_N + N((i\theta \ln \frac{u}{d_N} + \frac{1}{2}(\theta)^2 \ln^2 \frac{u}{d_N} p_N - \frac{(\theta)^2}{2} \ln^2 \frac{u}{d_N} p_N^2) + \frac{1}{2}(\theta)^2 p_N (1 - p_N) \ln^2 \frac{u}{d_N}}
\]

\[
= e^{i\theta \ln S_0 + i\theta N \ln(\frac{u}{d_N}) + \frac{1}{2}(\theta)^2 p_N (1 - p_N) \ln^2 \frac{u}{d_N}}
\]

\[
= e^{i\theta \ln S_0 + i\theta N\ln u_N + (1 - p_N) \ln d_N} + \frac{1}{2}(\theta)^2 p_N (1 - p_N) \ln^2 \frac{u}{d_N}.
\]
Note:

\[
\ln u_N = \ln(1 + \sqrt{T/N}) \approx \sigma \sqrt{T/N} - \frac{1}{2} \sigma^2 T \frac{1}{N},
\]

\[
\ln d_N = \ln(1 - \sqrt{T/N}) \approx -\sigma \sqrt{T/N} - \frac{1}{2} \sigma^2 T \frac{1}{N},
\]

\[
\ln \frac{u_N}{d_N} = \ln\left( \frac{1 + \sqrt{T/N}}{1 - \sqrt{T/N}} \right) = \left[ \frac{1}{1 - x} = 1 + x + x^2 + \ldots, |x| < 1 \right]
\]

\[
\approx \ln[(1 + \sqrt{T/N})(1 + \sqrt{T/N} + \sigma^2 T \frac{1}{N})]
\]

\[
\approx \ln(1 + 2\sigma \sqrt{T/N} + 2\sigma^2 T \frac{1}{N})
\]

\[
\approx 2\sigma \sqrt{T/N} + 2\sigma^2 T \frac{1}{N} - \frac{1}{2}(2\sigma \sqrt{T/N} + 2\sigma^2 T \frac{1}{N})^2
\]

\[
\approx 2\sigma \sqrt{T/N},
\]

\[
p_N \ln u_N \approx \left( \frac{1}{2} + \frac{r}{2\sigma} \sqrt{T/N} \right)(\sigma \sqrt{T/N} - \frac{1}{2} \sigma^2 T \frac{1}{N})
\]

\[
\approx \frac{\sigma}{2} \sqrt{T/N} - \frac{1}{4} \sigma^2 T \frac{1}{N} + \frac{r}{2} T \frac{1}{N},
\]

\[
(1 - p_N) \ln d_N \approx \left( \frac{1}{2} - \frac{r}{2\sigma} \sqrt{T/N} \right)(-\sigma \sqrt{T/N} - \frac{1}{2} \sigma^2 T \frac{1}{N})
\]

\[
\approx \frac{-\sigma}{2} \sqrt{T/N} - \frac{1}{4} \sigma^2 T \frac{1}{N} + \frac{r}{2} T \frac{1}{N},
\]

\[
p_N(1 - p_N) \ln \frac{u_N}{d_N} \approx \left( \frac{1}{2} + \frac{r}{2\sigma} \sqrt{T/N} \right)(\frac{1}{2} - \frac{r}{2\sigma} \sqrt{T/N}) 4\sigma^2 T \frac{1}{N}
\]

\[
= \frac{1}{4}(1 - \frac{r^2}{\sigma^2 N}) 4\sigma^2 T \frac{1}{N}
\]

\[
\approx \sigma^2 T \frac{1}{N}.
\]

Hence,

\[
\Psi_{\ln S_T^N}(\theta) = e^{i\theta \ln S_0 + i\theta N(p_N \ln u_N + (1-p_N) \ln d_N) + \frac{r}{2}(i\theta)^2 p_N(1-p_N) \ln^2 \frac{u_N}{d_N}}
\]

\[
\approx e^{i\theta \ln S_0 + i\theta N(-\frac{1}{2} \sigma^2 T \frac{1}{N} + r \frac{T}{N}) + \frac{N}{4}(i\theta)^2 \sigma^2 T \frac{1}{N}}
\]

\[
= e^{i\theta \ln S_0 + i\theta (r - \frac{\sigma^2}{2}) T + \frac{N}{4}(i\theta)^2 \sigma^2 T}
\]

\[
= e^{i\theta \ln S_0 + i\theta (r - \frac{\sigma^2}{2}) T - \frac{\sigma^2}{4} \sigma^2 T}.
\]
The characteristic function of $\ln S_0 + (r - \frac{\sigma^2}{2})T + \sigma W_T$ is:

$$\Psi_{\ln S_0 + (r - \frac{\sigma^2}{2})T + \sigma W_T}(\theta) = e^{i\theta \ln S_0 + i\theta(r - \frac{\sigma^2}{2})T - \frac{\sigma^2}{2} \theta^2}.$$ 

We conclude that $\Psi_{\ln S_0 + (r - \frac{\sigma^2}{2})T + \sigma W_T}(\theta) \approx \Psi_{\ln S_0 + (r - \frac{\sigma^2}{2})T + \sigma W_T}(\theta)$ for large $N$ and the proof is complete.

Therefore, under a binomial tree model with $u = u_N = 1 + \sigma \sqrt{T/N}$, $d = d_N = 1 - \sigma \sqrt{T/N}$, $1 + RT/N = 1 + R_NT/N = e^{rT/N}$, for the zero time European call option price, by (2.16):

$$f_0 = f_0^{Call,N} = e^{-rT} \left[ \sum_{i=0}^{N} \left( \begin{array}{c} N \\ i \end{array} \right) p^i (1 - p)^{N-i} \max \{ S_0 u^i d^{N-i} - X, 0 \} \right]$$

where $S_N^T = S_0 u^i d^{N-i}$. 

and for the zero time European put price, by (2.17):

$$f_0 = f_0^{Put,N} = e^{-rT} \left[ \sum_{i=0}^{N} \left( \begin{array}{c} N \\ i \end{array} \right) p^i (1 - p)^{N-i} \max \{ X - S_0 u^i d^{N-i}, 0 \} \right]$$

where $S_N^T = S_0 u^i d^{N-i}$. 

By Theorem 3.1.1, we get the limit of $f_0$:

$$\lim_{N \to \infty} f_0^{Call,N} = f_0 = e^{-rT} E[\max(S_T - X, 0)],$$

(3.21)

$$\lim_{N \to \infty} f_0^{Put,N} = f_0 = e^{-rT} E[\max(X - S_T, 0)],$$

(3.22)

where (3.21) is for the European call option, (3.22) is for the European put option and:

$$S_T = S_0 e^{(r - \frac{\sigma^2}{2})T + \sigma W_T}.$$
Now consider the European call option price (3.21) under (3.23). Then $f_0$ can be written as:

\[
\begin{align*}
    f_0 &= f_0^{\text{call}} \\
    &= e^{-rT}E[\max(S_T - X, 0)] \\
    &= e^{-rT}E[\max(S_0e^{(r-\frac{1}{2}\sigma^2)T+\sigma W_T} - X, 0)] = [W_T \overset{\mathcal{L}}{=} \sqrt{T}Z, \ Z \sim \text{Normal}(0,1)] \\
    &= e^{-rT}E[\max(S_0e^{(r-\frac{1}{2}\sigma^2)T+\sigma \sqrt{T}Z} - X, 0)] \\
    &= e^{-rT} \int_{-\infty}^{\infty} \max(S_0e^{(r-\frac{1}{2}\sigma^2)T+\sigma \sqrt{T}z} - X, 0) \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}z^2} dz \\
    &= \max(0, S_0e^{(r-\frac{1}{2}\sigma^2)T+\sigma \sqrt{T}z} - X, 0) \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}z^2} dz \\
    &\quad \text{[Observe: $S_0e^{(r-\frac{1}{2}\sigma^2)T+\sigma \sqrt{T}z} - X, 0 > 0$,]} \\
    &= e^{-rT} \int_{d_-}^{\infty} \max(S_0e^{(r-\frac{1}{2}\sigma^2)T+\sigma \sqrt{T}z} - X, 0) \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}z^2} dz \\
    &= e^{-rT} \int_{d_-}^{\infty} (S_0e^{(r-\frac{1}{2}\sigma^2)T+\sigma \sqrt{T}z} - X) \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}z^2} dz \\
    &= e^{-rT} S_0 e^{(r-\frac{1}{2}\sigma^2)T} \int_{d_-}^{\infty} e^{\sigma \sqrt{T}z - \frac{1}{2}z^2} \frac{1}{\sqrt{2\pi}} dz - e^{-rT} X \int_{d_-}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}z^2} dz \\
    &= \max(0, S_0e^{(r-\frac{1}{2}\sigma^2)T+\sigma \sqrt{T}z} - X) \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}z^2} dz \\
    &= \int_{-\infty}^{\infty} [S_0e^{(r-\frac{1}{2}\sigma^2)T+\sigma \sqrt{T}w} - X, 0 ] \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}w^2} dw - e^{-rT} X \int_{d_-}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}z^2} dz \\
    &= [\text{By symmetry of } y \sim \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}y^2}] \\
    &= \frac{S_0}{\sqrt{2\pi}} e^{-\frac{1}{2}w^2} dw - e^{-rT} X \int_{d_-}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}z^2} dz \\
    &= S_0 \phi(d_- + \sigma \sqrt{T}) - e^{-rT} X \Phi(d_-) \\
    &= S_0 \phi(d_- + \sigma \sqrt{T}) - e^{-rT} X \Phi(d_-),
\end{align*}
\]

where $d_- = \frac{\ln \frac{S_0}{X} + (r-\frac{1}{2}\sigma^2)T}{\sigma \sqrt{T}}$, $d_+ = d_- + \sigma \sqrt{T} = \frac{\ln \frac{S_0}{X} + (r-\frac{1}{2}\sigma^2)T + \sigma^2 T}{\sigma \sqrt{T}} = \frac{\ln \frac{S_0}{X} + (r+\frac{1}{2}\sigma^2)T}{\sigma \sqrt{T}}$, and $\Phi$ is the standard normal distribution function.
Hence we have shown the Black-Scholes option pricing formula for the European call option:

\[ f_{\text{call}}^0 = S_0 \Phi(d_+) - e^{-rT} X \Phi(d_-) \]  

\[ d_+ = \frac{\ln \frac{S_0}{X} + (r + \frac{1}{2} \sigma^2)T}{\sigma \sqrt{T}} \quad \text{and} \quad d_- = \frac{\ln \frac{S_0}{X} + (r - \frac{1}{2} \sigma^2)T}{\sigma \sqrt{T}}. \]

By the put-call-parity (1.1), \( f_{\text{call}}^0 - f_{\text{put}}^0 = S_0 - X e^{-rT} \), we can get the Black-Scholes option pricing formula for a European put option:

\[ f_{\text{call}}^0 - f_{\text{put}}^0 = S_0 - X e^{-rT}, \]

\[ S_0 \Phi(d_+) - e^{-rT} X \Phi(d_-) - f_{\text{put}}^0 = S_0 - X e^{-rT}. \]

The European put price \( f_{\text{put}}^0 \) can be written as:

\[ f_{\text{put}}^0 = S_0 \Phi(-d_+) + e^{-rT} X \Phi(-d_-) \]

\[ = -S_0(1 - \Phi(d_+)) + e^{-rT} X (1 - \Phi(d_-)) \]

\[ = -S_0 \Phi(-d_+) + e^{-rT} X \Phi(-d_-). \]

In other words:

\[ f_{\text{put}}^0 = -S_0 \Phi(-d_+) + e^{-rT} X \Phi(-d_-), \]  

\[ d_+ = \frac{\ln \frac{S_0}{X} + (r + \frac{1}{2} \sigma^2)T}{\sigma \sqrt{T}} \quad \text{and} \quad d_- = \frac{\ln \frac{S_0}{X} + (r - \frac{1}{2} \sigma^2)T}{\sigma \sqrt{T}}. \]

Formula (3.24) and (3.25) are the Black-Scholes option pricing formulas for European call and put option, respectively.

Figure 3.1.5: Standard normal distribution.
Example 3.1.2 (Black-Scholes formula) Assume the price of a no dividend stock at time zero is 50 kr, investor buy a European call option of this stock, the expire date is after six months, and strike price is 48 kr. Assume the volatility is 25%, the risk-free interest year rate is 10%, which means $S_0 = 50$, $X = 48$, $T = 0.5$, $r = 0.1$ and $\sigma = 0.25$. Compute the European call option price at time zero by using the Black-Scholes formula.

Solution:

By (3.24):

\[
d_+ = \frac{\ln(50/48) + 0.5(0.1 + 0.25^2/2)}{0.25\sqrt{0.5}} = \frac{0.11545}{0.1768} = 0.653
\]
\[
d_- = d_+ - \sigma \sqrt{T} = 0.653 - 0.1768 = 0.476,
\]
then:
\[
f_0^{\text{call}} = 50N(0.653) - 48 \cdot 0.9512N(0.476),
\]

By Figure 3.1.5:
\[
f_c = 50 \cdot 0.7422 - 48 \cdot 0.9512 \cdot 0.6808 = 6.03.
\]

The European call option price is 6 kr.

3.1.2 Derivation of the Black-Scholes Partial Differential Equation

A standard way to show the Black-Scholes formula (3.24) and (3.25) is to derive a PDE for the option price:

\[
\frac{\partial V}{\partial t} + \frac{1}{2} \sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS \frac{\partial V}{\partial S} - rV = 0, \quad 0 < t < T,
\]

(3.26)

with terminal condition for the European call option:
\[
V(T,S) = (S - X)^+,
\]

and terminal condition for the European put option:
\[
V(T,S) = (X - S)^+.
\]

Typically (3.26) is obtained by applying an important formula in stochastic calculus, Itô’s formula [5, Chapter 13: Wiener process and Itô lemma, pp. 280-297]. Here instead (3.26) is derived from our $N$-step binomial tree model with (3.19) and (3.20) as suggested in [6] but here with details.
In previous sections, the option price at time zero for the binomial tree model and its limit where denoted by \( f^N_0 \) and \( f_0 \), respectively. Here we denote the option price at time \( t \) for the binomial tree model and its limit as \( V_N(t) \) and \( V(t) \), respectively.

**Theorem 3.1.3** Let \( V_N(t) \) be an \( N \)-step European option price at time \( t \). Then \( V(t) = \lim_{N \to \infty} V_N(t) \) satisfies (3.26).

**Proof of Theorem 3.1.3:**

Recall (3.19), (3.20) and (2.18). We have:

\[
u = 1 + \sigma \sqrt{T/N}, \quad d = 1 - \sigma \sqrt{T/N}, \quad 1 + RT/N = e^{rT/N},
\]

Recall (2.15), by \( u = u_N = 1 + \sigma \sqrt{T/N}, \quad d = d_N = 1 - \sigma \sqrt{T/N} \) and \( e^{rT/N} = 1 + RT/N \):

\[
p = \frac{e^{rT/N} - d}{u - d} = \frac{1 + RT/N - d}{u - d} = \frac{1 + RT/N - 1 + \sigma \sqrt{T/N}}{1 + \sigma \sqrt{T/N} - (1 - \sigma \sqrt{T/N})} = \frac{RT/N + \sigma \sqrt{T/N}}{2\sigma \sqrt{T/N}} = \frac{1}{2} + \frac{R\sqrt{T/N}}{2\sigma}.
\]

For convenience at time \( \frac{T(k+1)}{N} \), consider the option price time \( t = \frac{kT}{N} \) for some \( k \in [0, ..., N - 1] \), now we can check the equation of \( V \), when \( T/N \) is small at time \( \frac{kT}{N} \):

\[
V = V(t, S), \quad V^+ = V(t + T/N, uS), \quad V^- = V(t + T/N, dS).
\]

By the same argument as in (2.8) with \( T \) replaced by \( T/N \), \( V \) is the option price, \( V^+ \) is the upper option price and \( V^- \) is the lower option price:

\[
V = \frac{1}{1 + RT/N} (pV^+ + (1 - p)V^-).
\]

Expanding (3.27) by using Taylor series step by step when \( T/N \) is small, and letting \( \partial_1 = \frac{\partial}{\partial t} \) and \( \partial_2 = \frac{\partial}{\partial S} \) for convenience,

\[
g_+(T/N) = V(t + T/N, uS)
\]
\[ g_+^\prime(T/N) = \partial_1 V + \partial_2 V \frac{\sigma}{2\sqrt{T/N}} S, \]
\[ g_+^\prime\prime(T/N) = \partial_1(\partial_1 V) + \partial_2(\partial_1 V) \frac{\sigma}{2\sqrt{T/N}} S + \partial_1(\partial_2 V) \frac{\sigma}{2\sqrt{T/N}} S + (\partial_2 V)(-\frac{1}{2}\frac{\sigma}{(T/N)^2} S) \]
\[ = \partial_1^2 V + 2\partial_1\partial_2 V \frac{\sigma}{2\sqrt{T/N}} S + \partial_2^2 V \frac{\sigma^2 S^2}{4T/N} - \frac{1}{4}\partial_2^2 V \frac{\sigma^3}{(T/N)^3} S, \]
\[ g_-^\prime(T/N) = V(t + T/N, dS) \]
\[ = V(t + T/N, (1 - \sigma \sqrt{T/N}) S), \]
\[ g_-^\prime\prime(T/N) = \partial_1 V - \partial_2 V \frac{\sigma}{2\sqrt{T/N}} S, \]
\[ g_-^\prime\prime(T/N) = \partial_1(\partial_1 V) + \partial_2(\partial_1 V) \frac{\sigma}{2\sqrt{T/N}} S - \partial_1(\partial_2 V) \frac{\sigma}{2\sqrt{T/N}} S - (\partial_2 V)(-\frac{1}{2}\frac{\sigma}{(T/N)^2} S) \]
\[ = \partial_1^2 V - 2\partial_1\partial_2 V \frac{\sigma}{2\sqrt{T/N}} S + \partial_2^2 V \frac{\sigma^2 S^2}{4T/N} + \frac{1}{4}\partial_2^2 V \frac{\sigma^3}{(T/N)^3} S. \]

Then by Taylor expansion of \( V^+ \) and \( V^- \):

\[ V^+ = g_+(T/N) \]
\[ = V + (\partial_1 V + \partial_2 V \frac{\sigma}{2\sqrt{T/N}} S)(T/N) + \frac{1}{2}(\partial_1^2 V + 2\partial_1\partial_2 V \frac{\sigma}{2\sqrt{T/N}} S + \partial_2^2 V \frac{\sigma^2 S^2}{4T/N} - \frac{1}{4}\partial_2^2 V \frac{\sigma^3}{(T/N)^3} S)(T/N)^2 + o((T/N)^3) \]
\[ = V + \partial_1 V(T/N) + \partial_2 V \frac{\sigma S}{8\sqrt{T/N}} + \frac{1}{2}\partial_1^2 V(T/N)^2 + \frac{1}{2}\partial_1\partial_2 V \frac{\sigma^2 S}{2\sqrt{T/N}} + \frac{1}{4}\partial_2^2 V \frac{\sigma^3 S^2}{(T/N)^2} + \frac{1}{8}\partial_2^2 V \frac{\sigma S^3}{(T/N)^3} + o((T/N)^3) \]
\[ = 3\partial_2 V \frac{\sigma S}{2\sqrt{T/N}} + (\partial_1 V + \frac{1}{8}\partial_2^2 V \frac{\sigma^2 S}{2\sqrt{T/N}})(T/N) + \frac{1}{2}\partial_1\partial_2 V \frac{\sigma S}{2\sqrt{T/N}}(T/N)^2 + \frac{1}{2}\partial_2^2 V(T/N)^2 + o(1) \]

\[ V^- = g_-(T/N) \]
\[ = V + (\partial_1 V - \partial_2 V \frac{\sigma}{2\sqrt{T/N}} S)(T/N) + \frac{1}{2}(\partial_1^2 V - 2\partial_1\partial_2 V \frac{\sigma}{2\sqrt{T/N}} S + \partial_2^2 V \frac{\sigma^2 S^2}{4T/N} + \frac{1}{4}\partial_2^2 V \frac{\sigma^3}{(T/N)^3} S)(T/N)^2 + o((T/N)^3) \]

35
Recall (3.28):

\[ V = \frac{1}{1 + RT/N} \left( pV^+ + (1 - p)V^- \right), \]

which can be expressed as:

\[ (1 + RT/N)V = \left( \frac{1}{2} + \frac{R\sqrt{T/N}}{2\sigma} \right) V^+ + \left( \frac{1}{2} - \frac{R\sqrt{T/N}}{2\sigma} \right) V^- \quad (3.29) \]

One can write (3.29) as:

\[ (1 + R\Delta t)V = \left( \frac{1}{2} + \frac{R\sqrt{\Delta t}}{2\sigma} \right) V^+ + \left( \frac{1}{2} - \frac{R\sqrt{\Delta t}}{2\sigma} \right) V^- \quad (3.30) \]

There is a problem by Taylor expanding the right hand side of (3.30), around \( \Delta t = 0 \) since \( \frac{\Delta t}{\sqrt{\Delta t}} \to \infty \) as \( \Delta t \to 0 \). Therefore, let \( \epsilon = \sqrt{\Delta t} = \sqrt{T/N} \),

\[
\begin{align*}
V_+ &= V(t + T/N, (1 + \sigma\sqrt{T/N})S) \\
&= V(t + \epsilon^2, (1 + \sigma\epsilon)S) = V_+(\epsilon),
\end{align*}
\]

\[
\begin{align*}
V_+^{(\epsilon)} &= \partial_{2}V^{2}\epsilon + \partial_{3}V\sigma S, \\
V_+^{(\epsilon\epsilon)} &= [\partial_{1}(\partial_{1}V)^{2}\epsilon + \partial_{2}(\partial_{1}V)\sigma S]2\epsilon + \partial_{1}V^{2} + [\partial_{1}(\partial_{2}V)^{2}\epsilon + \partial_{2}(\partial_{2}V)\sigma S]\sigma S \\
&= 4\partial_{1}V^{2}\epsilon^{2} + 4\partial_{1}\partial_{1}V^{2}\sigma S\epsilon + 2\partial_{1}V + 2\partial_{1}^{2}V\sigma^{2}S^{2} + 2\partial_{1}V + 2\partial_{1}^{2}V\sigma^{2}S + 4\partial_{1}^{2}V\epsilon^{2},
\end{align*}
\]

and the same for \( V_- \):

\[
\begin{align*}
V_- &= V(t + T/N, (1 - \sigma\sqrt{T/N})S) \\
&= V(t + \epsilon^2, (1 - \sigma\epsilon)S) = V_-(\epsilon),
\end{align*}
\]

\[
\begin{align*}
V_-^{(\epsilon)} &= \partial_{2}V^{2}\epsilon - \partial_{3}V\sigma S, \\
V_-^{(\epsilon\epsilon)} &= [\partial_{1}(\partial_{1}V)^{2}\epsilon - \partial_{2}(\partial_{1}V)\sigma S]2\epsilon + \partial_{1}V^{2} - [\partial_{1}(\partial_{2}V)^{2}\epsilon - \partial_{2}(\partial_{2}V)\sigma S]\sigma S \\
&= 4\partial_{1}V^{2}\epsilon^{2} - 4\partial_{1}\partial_{1}V^{2}\sigma S\epsilon + 2\partial_{1}V + 2\partial_{1}^{2}V\sigma^{2}S^{2} + 2\partial_{1}V + 2\partial_{1}^{2}V\sigma^{2}S - 4\partial_{1}^{2}V\sigma S\epsilon + 4\partial_{1}^{2}V\epsilon^{2}.
\end{align*}
\]

By Taylor expanding, when \( \epsilon \) is small:

\[
\begin{align*}
V_+(\epsilon) &= V_+(0) + V_+^{(\epsilon)}(0)\epsilon + \frac{1}{2}V_+^{(\epsilon\epsilon)}(0)\epsilon^{2} + o(\epsilon^{3})
\end{align*}
\]
\[ V + \partial_2 V \sigma S \epsilon + \frac{1}{2} (2 \partial_1 V + \partial_2^2 \sigma^2 S^2) \epsilon^2 + o(\epsilon^3), \]

\[ V_-(\epsilon) = V_-(0) + V'_-(0) \epsilon + \frac{1}{2} V''_-(0) \epsilon^2 + o(\epsilon^3) \]

\[ = V - \partial_2 V \sigma S \epsilon + \frac{1}{2} (2 \partial_1 V + \partial_2^2 \sigma^2 S^2) \epsilon^2 + o(\epsilon^3). \]

Recall (3.29). Then:

\[ (1 + R \epsilon^2) V = \frac{1}{2} [V_+(\epsilon) + V_-(\epsilon)] + \frac{R \epsilon}{\sigma} \frac{1}{2} [V_+(\epsilon) - V_-(\epsilon)] + o(\epsilon^3) \]

\[ = V + \frac{1}{2} (2 \partial_1 V + \partial_2^2 \sigma^2 S^2) \epsilon^2 + \frac{R \epsilon}{\sigma} \partial_2 V \sigma S \epsilon + o(\epsilon^3) \]

\[ = V + [\partial_1 V + R \partial_2 V S + \frac{1}{2} \partial_2^2 V \sigma^2 S^2] \epsilon^2 + o(\epsilon^3) \]

\[ R \epsilon^2 V = [\partial_1 V + R \partial_2 V S + \frac{1}{2} \partial_2^2 V \sigma^2 S^2] \epsilon^2 + o(\epsilon^3) \]

\[ RV = \partial_1 V + R \partial_2 V S + \frac{1}{2} \partial_2^2 V \sigma^2 S^2. \]

Since \( 1 + RT/N = e^{rT/N} \), \( R = (e^{rT/N-1})^{N-1}/N \to r \) as \( N \to \infty \). Hence,

\[ rV = \partial_1 V + r \partial_2 V S + \frac{1}{2} \partial_2^2 V \sigma^2 S^2. \]

i.e.

\[ \frac{\partial V}{\partial t} + \frac{1}{2} \sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS \frac{\partial V}{\partial S} - rV = 0. \]

We have obtained that when \( N \to \infty \) under certain assumptions of \( u \) and \( d \) by (3.19) and (3.20), the limiting binomial tree option price satisfies the Black-Scholes PDE (3.26). \( \square \)

### 3.2 Numerical comparison of the \( N \)-step binomial model and its Black-Scholes limit

Based on the algorithms of pricing options under the Black-Scholes model by using binomial tree models for many steps and the Black-Scholes option pricing formulas, some Java programs are written by the author to compute option prices at time zero.

**Notation:** AC=American call by using binomial, EC=European call by using binomial, AP=American put by using binomial, EP=European put by using binomial, ECF=European call by using Black-Scholes formula, EPF=European put by using Black-Scholes formula.
**Example 3.2.1** Assume the initial price of a no dividend stock is 50 kr, the expiry day is after six months, and strike price is 48 kr. Assume the volatility is 25%, the risk-free interest year rate is 10%, which means $S_0 = 50$, $X = 48$, $T = 0.5$, $r = 0.1$ and $\sigma = 0.25$. Compute European and American time zero option prices by using the binomial tree option pricing and the Black-Scholes formula.

*Solution:* Table 3.2.5 and Figure 3.2.6.
## Table 3.2.5

<table>
<thead>
<tr>
<th>N</th>
<th>AC</th>
<th>EC</th>
<th>ECF</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6.6212000000000000</td>
<td>6.6212000000000000</td>
<td>5.972881054522705</td>
</tr>
<tr>
<td>2</td>
<td>5.792980368828079</td>
<td>5.792980368828079</td>
<td>5.972881054522705</td>
</tr>
<tr>
<td>3</td>
<td>6.1820893353342745</td>
<td>6.1820893353342745</td>
<td>5.972881054522705</td>
</tr>
<tr>
<td>4</td>
<td>5.92603011405526</td>
<td>5.92603011405526</td>
<td>5.972881054522705</td>
</tr>
<tr>
<td>5</td>
<td>6.0860122697298324</td>
<td>6.0860122697298324</td>
<td>5.972881054522705</td>
</tr>
<tr>
<td>10</td>
<td>5.986871479128721</td>
<td>5.986871479128721</td>
<td>5.972881054522705</td>
</tr>
<tr>
<td>20</td>
<td>5.993658039945514</td>
<td>5.993658039945514</td>
<td>5.972881054522705</td>
</tr>
<tr>
<td>50</td>
<td>6.0860122697298324</td>
<td>6.0860122697298324</td>
<td>5.972881054522705</td>
</tr>
<tr>
<td>80</td>
<td>5.979659196428628</td>
<td>5.979659196428628</td>
<td>5.972881054522705</td>
</tr>
<tr>
<td>100</td>
<td>5.972766402974998</td>
<td>5.972766402974998</td>
<td>5.972881054522705</td>
</tr>
<tr>
<td>150</td>
<td>5.971736321907938</td>
<td>5.971736321907938</td>
<td>5.972881054522705</td>
</tr>
<tr>
<td>200</td>
<td>5.9688760851854</td>
<td>5.9688760851854</td>
<td>5.972881054522705</td>
</tr>
<tr>
<td>250</td>
<td>5.97051156344141</td>
<td>5.97051156344141</td>
<td>5.972881054522705</td>
</tr>
<tr>
<td>300</td>
<td>5.973519641442152</td>
<td>5.973519641442152</td>
<td>5.972881054522705</td>
</tr>
<tr>
<td>500</td>
<td>5.973519641442152</td>
<td>5.973519641442152</td>
<td>5.972881054522705</td>
</tr>
<tr>
<td>1000</td>
<td>5.973070908357869</td>
<td>5.973070908357869</td>
<td>5.972881054522705</td>
</tr>
<tr>
<td>2000</td>
<td>5.972820979814132</td>
<td>5.972820979814132</td>
<td>5.972881054522705</td>
</tr>
<tr>
<td>2500</td>
<td>5.972051156344141</td>
<td>5.972051156344141</td>
<td>5.972881054522705</td>
</tr>
<tr>
<td>3000</td>
<td>5.973519641442152</td>
<td>5.973519641442152</td>
<td>5.972881054522705</td>
</tr>
<tr>
<td>4000</td>
<td>5.972949503553287</td>
<td>5.972949503553287</td>
<td>5.972881054522705</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>N</th>
<th>AP</th>
<th>EP</th>
<th>EPF</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.3354858046576576</td>
<td>2.3354858046576576</td>
<td>1.631804814865397</td>
</tr>
<tr>
<td>2</td>
<td>1.6583658365836583</td>
<td>1.480071386079715</td>
<td>1.631804814865397</td>
</tr>
<tr>
<td>3</td>
<td>1.9931613593431934</td>
<td>1.859921400573213</td>
<td>1.631804814865397</td>
</tr>
<tr>
<td>4</td>
<td>1.7411267999002145</td>
<td>1.599168221828871</td>
<td>1.631804814865397</td>
</tr>
<tr>
<td>5</td>
<td>1.8971375687840601</td>
<td>1.75635274853776</td>
<td>1.631804814865397</td>
</tr>
<tr>
<td>10</td>
<td>1.792763217666823</td>
<td>1.6515726325400155</td>
<td>1.631804814865397</td>
</tr>
<tr>
<td>20</td>
<td>1.806444342102987</td>
<td>1.6655194458893505</td>
<td>1.631804814865397</td>
</tr>
<tr>
<td>50</td>
<td>1.8018751737741083</td>
<td>1.645885351671653</td>
<td>1.631804814865397</td>
</tr>
<tr>
<td>80</td>
<td>1.7979206075148335</td>
<td>1.6393847029803876</td>
<td>1.631804814865397</td>
</tr>
<tr>
<td>100</td>
<td>1.79056776215651</td>
<td>1.636349330041614</td>
<td>1.631804814865397</td>
</tr>
<tr>
<td>150</td>
<td>1.7910493209062839</td>
<td>1.631029167645248</td>
<td>1.631804814865397</td>
</tr>
<tr>
<td>200</td>
<td>1.7885270122683512</td>
<td>1.6281853067191796</td>
<td>1.631804814865397</td>
</tr>
<tr>
<td>250</td>
<td>1.78956127374715</td>
<td>1.6312917975670728</td>
<td>1.631804814865397</td>
</tr>
<tr>
<td>300</td>
<td>1.7905813798750112</td>
<td>1.632722242622639</td>
<td>1.631804814865397</td>
</tr>
<tr>
<td>500</td>
<td>1.791234140843515</td>
<td>1.63311156015811</td>
<td>1.631804814865397</td>
</tr>
<tr>
<td>1000</td>
<td>1.790494622803047</td>
<td>1.632140356291452</td>
<td>1.631804814865397</td>
</tr>
<tr>
<td>2000</td>
<td>1.7901150346111447</td>
<td>1.631861892269305</td>
<td>1.631804814865397</td>
</tr>
<tr>
<td>2500</td>
<td>1.7902600104590325</td>
<td>1.6320726849037022</td>
<td>1.631804814865397</td>
</tr>
<tr>
<td>3000</td>
<td>1.7900278158322012</td>
<td>1.6317163963978156</td>
<td>1.631804814865397</td>
</tr>
<tr>
<td>4000</td>
<td>1.7901679189071666</td>
<td>1.6310761479123371</td>
<td>1.631804814865397</td>
</tr>
</tbody>
</table>

Table 3.2.5: $S_0 = 50$, $X = 48$, $T = 0.5$, $r = 0.1$, $\sigma = 0.25$. 

39
Figure 3.2.6: $S_0 = 50$, $X = 48$, $T = 0.5$, $r = 0.1$, $\sigma = 0.25$.

**Example 3.2.2** Assume the initial price of a no dividend stock is 75 kr, the expiry day is after six months, and strike price is 79 kr. Assume the volatility is 25%, the risk-free interest year rate is 10%, which means $S_0 = 75$, $X = 79$, $T = 0.5$, $r = 0.1$ and $\sigma = 0.25$. Compute European and American time zero option prices by using the binomial tree option pricing and the Black-Scholes formulas.
Solution: Table 3.2.6 and Figure 3.2.7.

<table>
<thead>
<tr>
<th>N</th>
<th>AC</th>
<th>EC</th>
<th>ECF</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5.65657706064252</td>
<td>5.6565776064252</td>
<td>5.214341266368798</td>
</tr>
<tr>
<td>2</td>
<td>5.45681142177255</td>
<td>5.45681142177255</td>
<td>5.214341266368798</td>
</tr>
<tr>
<td>3</td>
<td>5.13865220824673</td>
<td>5.13865220824673</td>
<td>5.214341266368798</td>
</tr>
<tr>
<td>4</td>
<td>5.4282772084250315</td>
<td>5.4282772084250315</td>
<td>5.214341266368798</td>
</tr>
<tr>
<td>5</td>
<td>5.01746304035501</td>
<td>5.01746304035501</td>
<td>5.214341266368798</td>
</tr>
<tr>
<td>10</td>
<td>5.29807381780166</td>
<td>5.29807381780166</td>
<td>5.214341266368798</td>
</tr>
<tr>
<td>20</td>
<td>5.19563531094972</td>
<td>5.19563531094972</td>
<td>5.214341266368798</td>
</tr>
<tr>
<td>50</td>
<td>5.22859705312705</td>
<td>5.22859705312705</td>
<td>5.214341266368798</td>
</tr>
<tr>
<td>80</td>
<td>5.220514940500632</td>
<td>5.220514940500632</td>
<td>5.214341266368798</td>
</tr>
<tr>
<td>100</td>
<td>5.206040870537112</td>
<td>5.206040870537112</td>
<td>5.214341266368798</td>
</tr>
<tr>
<td>150</td>
<td>5.218942306850856</td>
<td>5.218942306850856</td>
<td>5.214341266368798</td>
</tr>
<tr>
<td>200</td>
<td>5.216945620320739</td>
<td>5.216945620320739</td>
<td>5.214341266368798</td>
</tr>
<tr>
<td>250</td>
<td>5.2085040921979754</td>
<td>5.2085040921979754</td>
<td>5.214341266368798</td>
</tr>
<tr>
<td>300</td>
<td>5.216270512924289</td>
<td>5.216270512924289</td>
<td>5.214341266368798</td>
</tr>
<tr>
<td>500</td>
<td>5.21579036415166</td>
<td>5.21579036415166</td>
<td>5.214341266368798</td>
</tr>
<tr>
<td>1000</td>
<td>5.213128290193111</td>
<td>5.213128290193111</td>
<td>5.214341266368798</td>
</tr>
<tr>
<td>1500</td>
<td>5.21491312165897</td>
<td>5.21491312165897</td>
<td>5.214341266368798</td>
</tr>
<tr>
<td>2000</td>
<td>5.214806740757724</td>
<td>5.214806740757724</td>
<td>5.214341266368798</td>
</tr>
<tr>
<td>2500</td>
<td>5.214709558322726</td>
<td>5.214709558322726</td>
<td>5.214341266368798</td>
</tr>
<tr>
<td>3000</td>
<td>5.214656806265594</td>
<td>5.214656806265594</td>
<td>5.214341266368798</td>
</tr>
<tr>
<td>4000</td>
<td>5.21436788776132</td>
<td>5.21436788776132</td>
<td>5.214341266368798</td>
</tr>
<tr>
<td>N</td>
<td>AP</td>
<td>EP</td>
<td>EPF</td>
</tr>
<tr>
<td>-----</td>
<td>---------------</td>
<td>---------------</td>
<td>----------------</td>
</tr>
<tr>
<td>1</td>
<td>5.893752998737757</td>
<td>5.893752998737757</td>
<td>5.361465801925199</td>
</tr>
<tr>
<td>2</td>
<td>6.400951814396194</td>
<td>5.6490184414039275</td>
<td>5.361465801925199</td>
</tr>
<tr>
<td>3</td>
<td>5.992787212544943</td>
<td>5.3119638281970945</td>
<td>5.361465801925199</td>
</tr>
<tr>
<td>4</td>
<td>6.228056208777035</td>
<td>5.598694950580117</td>
<td>5.361465801925199</td>
</tr>
<tr>
<td>5</td>
<td>5.948112058852832</td>
<td>5.183252361289167</td>
<td>5.361465801925199</td>
</tr>
<tr>
<td>10</td>
<td>6.1166647703708</td>
<td>5.4562946967696</td>
<td>5.361465801925199</td>
</tr>
<tr>
<td>20</td>
<td>6.048319836226081</td>
<td>5.347448874989951</td>
<td>5.361465801925199</td>
</tr>
<tr>
<td>50</td>
<td>6.0719989991948715</td>
<td>5.37786169921618</td>
<td>5.361465801925199</td>
</tr>
<tr>
<td>80</td>
<td>6.063408892108720</td>
<td>5.3688131700336665</td>
<td>5.361465801925199</td>
</tr>
<tr>
<td>100</td>
<td>6.05563895551573</td>
<td>5.35410443802157</td>
<td>5.361465801925199</td>
</tr>
<tr>
<td>150</td>
<td>6.063241075071758</td>
<td>5.366692859272588</td>
<td>5.361465801925199</td>
</tr>
<tr>
<td>200</td>
<td>6.060773155783871</td>
<td>5.36459748609143</td>
<td>5.361465801925199</td>
</tr>
<tr>
<td>250</td>
<td>6.057246244586998</td>
<td>5.356041144007351</td>
<td>5.361465801925199</td>
</tr>
<tr>
<td>300</td>
<td>6.060799740527209</td>
<td>5.36708127367028</td>
<td>5.361465801925199</td>
</tr>
<tr>
<td>500</td>
<td>6.059827915668232</td>
<td>5.362491427494148</td>
<td>5.361465801925199</td>
</tr>
</tbody>
</table>
Example 3.2.3 We consider a case when the difference value between initial price and strike price is huge. Assume the initial price of a no dividend stock is 300 kr, the expiry day is after six months, and strike price is 20 kr. Assume the volatility of this stock is 25%, the risk-free interest year rate is 10%, which means $S_0 = 300$, $X = 20$, $T = 0.5$, $r = 0.1$ and $\sigma = 0.25$. Compute European and American time zero option prices by using the binomial tree option pricing and the Black-Scholes formula.
Solution: Table 3.2.7 and Figure 3.2.8.

<table>
<thead>
<tr>
<th>( \text{N} )</th>
<th>( \text{AC} )</th>
<th>( \text{EC} )</th>
<th>( \text{ECF} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>280.95238095238096</td>
<td>280.95238095238096</td>
<td>280.97541150998575</td>
</tr>
<tr>
<td>2</td>
<td>280.9637120761452</td>
<td>280.9637120761452</td>
<td>280.97541150998575</td>
</tr>
<tr>
<td>3</td>
<td>280.96756997281716</td>
<td>280.96756997281716</td>
<td>280.97541150998575</td>
</tr>
<tr>
<td>4</td>
<td>280.969514495657</td>
<td>280.969514495657</td>
<td>280.97541150998575</td>
</tr>
<tr>
<td>5</td>
<td>280.970686247865</td>
<td>280.970686247865</td>
<td>280.97541150998575</td>
</tr>
<tr>
<td>10</td>
<td>280.97304118607894</td>
<td>280.97304118607894</td>
<td>280.97541150998575</td>
</tr>
<tr>
<td>20</td>
<td>280.974224414149</td>
<td>280.974224414149</td>
<td>280.97541150998575</td>
</tr>
<tr>
<td>50</td>
<td>280.9749362061762</td>
<td>280.9749362061762</td>
<td>280.97541150998575</td>
</tr>
<tr>
<td>80</td>
<td>280.975137803837</td>
<td>280.975137803837</td>
<td>280.97541150998575</td>
</tr>
<tr>
<td>100</td>
<td>280.97525063071</td>
<td>280.97525063071</td>
<td>280.97541150998575</td>
</tr>
<tr>
<td>150</td>
<td>280.975292657448</td>
<td>280.975292657448</td>
<td>280.97541150998575</td>
</tr>
<tr>
<td>200</td>
<td>280.9753163994882</td>
<td>280.9753163994882</td>
<td>280.97541150998575</td>
</tr>
<tr>
<td>250</td>
<td>280.97532322450274</td>
<td>280.97532322450274</td>
<td>280.97541150998575</td>
</tr>
<tr>
<td>300</td>
<td>280.975329516231</td>
<td>280.975329516231</td>
<td>280.97541150998575</td>
</tr>
<tr>
<td>500</td>
<td>280.9753775390969</td>
<td>280.9753775390969</td>
<td>280.97541150998575</td>
</tr>
<tr>
<td>700</td>
<td>280.9753775390969</td>
<td>280.9753775390969</td>
<td>280.97541150998575</td>
</tr>
<tr>
<td>1000</td>
<td>280.975387729989</td>
<td>280.975387729989</td>
<td>280.97541150998575</td>
</tr>
<tr>
<td>1500</td>
<td>280.9753956555336</td>
<td>280.9753956555336</td>
<td>280.97541150998575</td>
</tr>
<tr>
<td>2000</td>
<td>280.9753996198242</td>
<td>280.9753996198242</td>
<td>280.97541150998575</td>
</tr>
<tr>
<td>2500</td>
<td>280.9754019978655</td>
<td>280.9754019978655</td>
<td>280.97541150998575</td>
</tr>
<tr>
<td>3000</td>
<td>280.9754038305986</td>
<td>280.9754038305986</td>
<td>280.97541150998575</td>
</tr>
<tr>
<td>4000</td>
<td>280.9754055648465</td>
<td>280.9754055648465</td>
<td>280.97541150998575</td>
</tr>
<tr>
<td>5000</td>
<td>280.9754067379054</td>
<td>280.9754067379054</td>
<td>280.97541150998575</td>
</tr>
<tr>
<td>6000</td>
<td>280.9754075465098</td>
<td>280.9754075465098</td>
<td>280.97541150998575</td>
</tr>
<tr>
<td>7500</td>
<td>280.9754083389477</td>
<td>280.9754083389477</td>
<td>280.97541150998575</td>
</tr>
<tr>
<td>9000</td>
<td>280.9754086762766</td>
<td>280.9754086762766</td>
<td>280.97541150998575</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>( \text{N} )</th>
<th>( \text{AP} )</th>
<th>( \text{EP} )</th>
<th>( \text{EPF} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.0</td>
<td>0.0</td>
<td>2.9887811744807084E-55</td>
</tr>
<tr>
<td>2</td>
<td>0.0</td>
<td>0.0</td>
<td>2.9887811744807084E-55</td>
</tr>
<tr>
<td>3</td>
<td>0.0</td>
<td>0.0</td>
<td>2.9887811744807084E-55</td>
</tr>
<tr>
<td>4</td>
<td>0.0</td>
<td>0.0</td>
<td>2.9887811744807084E-55</td>
</tr>
<tr>
<td>5</td>
<td>0.0</td>
<td>0.0</td>
<td>2.9887811744807084E-55</td>
</tr>
<tr>
<td>10</td>
<td>0.0</td>
<td>0.0</td>
<td>2.9887811744807084E-55</td>
</tr>
<tr>
<td>20</td>
<td>0.0</td>
<td>0.0</td>
<td>2.9887811744807084E-55</td>
</tr>
<tr>
<td>50</td>
<td>0.0</td>
<td>0.0</td>
<td>2.9887811744807084E-55</td>
</tr>
<tr>
<td>80</td>
<td>0.0</td>
<td>0.0</td>
<td>2.9887811744807084E-55</td>
</tr>
<tr>
<td>100</td>
<td>0.0</td>
<td>0.0</td>
<td>2.9887811744807084E-55</td>
</tr>
<tr>
<td>150</td>
<td>0.0</td>
<td>0.0</td>
<td>2.9887811744807084E-55</td>
</tr>
<tr>
<td>200</td>
<td>0.0</td>
<td>0.0</td>
<td>2.9887811744807084E-55</td>
</tr>
<tr>
<td>250</td>
<td>2.919771252680334E-70</td>
<td>2.91839693356954E-70</td>
<td>2.9887811744807084E-55</td>
</tr>
<tr>
<td>300</td>
<td>1.585639851994967E-65</td>
<td>1.575404744855247E-65</td>
<td>2.9887811744807084E-55</td>
</tr>
</tbody>
</table>

43
Table 3.2.7: $S_0 = 300$, $X = 20$, $T = 0.5$, $r = 0.1$, $\sigma = 0.25$, and $E^- = 10^-$. 

Figure 3.2.8: $S_0 = 300$, $X = 20$, $T = 0.5$, $r = 0.1$, $\sigma = 0.25$. 

44
Comparison of binomial option prices with Black-Scholes option price formula from Example 3.2.1 to Example 3.2.3:

From Table 3.2.5 to Table 3.2.7, we can see that the American call is always equal to the European call and they are both very close to ECF when $N$ is huge, which fulfill Proposition 1.0.1. Meanwhile, when $N$ is huge, we can see that the American put is obviously larger than the EP (Remark 1.0.2), and European put is close to EPF.

More precisely, Figure 3.2.6 to Figure 3.2.7 shows that American call is covered by European call (the curve color changed and the value difference between the AC and the EC is always 0), hence American call is equal to European call (Proposition 1.0.1). Besides, the curves of American call and European call tend to the curve of ECF when $N$ is huge. Finally, we can still see that the curve of European put tends to the curve of EPF and the curve of American put is above the curve of European put, it means that $P_A > P_E$ (Remark 1.0.2).

Observe that from the Table 3.2.7 and Figure 3.2.8, when the value difference between the initial stock price and the strike price is very huge, $S_0 \gg X$, the American put option price is close to 0, which means in this situation American put option is almost worthless.

4 Conclusion

In this thesis, we derive the Black-Scholes limit under the $N$-step binomial tree model, and derive the Black-Scholes formulas and the Black-Scholes PDE by using $N$-step binomial tree model limits. Consequently, the American put option can be priced under the Black-Scholes model by using a $N$-step binomial tree model with many steps $N$.

Under the Black-Scholes model, the American put option can not be elementary priced. However, here the reader can price American put option under the Black-Scholes model by elementary binomial tree model option pricing.

5 Appendix

The Java codes can be applied in Eclipse. There are 5 programs written by Java, and every program can run isolated. Author: Yuankai Yang.

1. A Java program for N-step binomial tree model. This program compute American call, American put, European call and European put, and there is an extra function that can show option prices of every step if you are willing to know:
import java.util.Scanner;
public class BinomialTree {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Steps: ");
        int N = scan.nextInt();
        System.out.print("Initial price: ");
        double S = scan.nextDouble();
        System.out.print("Strike price: ");
        double St = scan.nextDouble();
        System.out.print("u: ");
        double u = scan.nextDouble();
        System.out.print("d: ");
        double d = scan.nextDouble();
        System.out.print("T: ");
        double T = scan.nextDouble();
        System.out.print("R: ");
        double R = scan.nextDouble();
        double[] call = BioCall(N, S, St, u, d);
        double[] options = BioOption(call, N, R, u, d, T);
        double[] put = BioPut(N, S, St, u, d);
        double[] option = BioOption(put, N, R, u, d, T);
        double[] Ecall = BioCall(N, S, St, u, d);
        double[] Eoptions = EuropeanBiOption(Ecall, N, R, u, d, T);
        double[] Eput = BioPut(N, S, St, u, d);
        double[] Eoption = EuropeanBiOption(Eput, N, R, u, d, T);
        System.out.println();
        System.out.println("(BinaryTree) American call Option price at 0 is: "+options[0]);
        System.out.println("(BinaryTree) American put Option price at 0 is: "+option[0]);
        System.out.println("(BinaryTree) European call Option price at 0 is: "+Eoptions[0]);
        System.out.println("(BinaryTree) European put Option price at 0 is: "+Eoption[0]);
        System.out.println();
        System.out.print("Do you want show details of every point of American call option price? 
(1 is for Yes, 2 is for No): ");
        int kk = scan.nextInt();
        System.out.println();
        if (kk == 1) {
            for (int i = 0; i < options.length; i++) {
                System.out.println(options[i]);
            }
        } else {
            System.out.println();
        }
    }
}
System.out.print("Do you want show details of every point of American put option price? (1 is for Yes, 2 is for No): ");
int kkk=scan.nextInt();
System.out.println();
if(kkk==1){
    for(int i=0;i<option.length;i++){
        System.out.println(option[i]);
    }
    System.out.println();
}
else{
    System.out.println();
}
System.out.print("Do you want show details of every point of European call option price? (1 is for Yes, 2 is for No): ");
int kkkk=scan.nextInt();
System.out.println();
if(kkkk==1){
    for(int i=0;i<Eoptions.length;i++){
        System.out.println(Eoptions[i]);
    }
    System.out.println();
}
else{
    System.out.println();
}
System.out.print("Do you want show details of every point of European put option price? (1 is for Yes, 2 is for No): ");
int kkkkk=scan.nextInt();
System.out.println();
if(kkkkk==1){
    for(int i=0;i<Eoption.length;i++){
        System.out.println(Eoption[i]);
    }
    System.out.println();
}
else{
    System.out.println();
}
scan.close();

public static double[] BioCall(int N, double S, double St, double u, double d) {
    double[] a = new double[(N+2)*(N+1)/2];
    for(int i=0; i<a.length; i++) {
        a[i] = Big(S * Math.pow(d, Column(i+1) - Row(i+1)) * Math.pow(u, Row(i+1)) - St, 0);
    }
    return a;
}

public static double[] BioPut(int N, double S, double St, double u, double d) {
    double[] a = new double[(N+2)*(N+1)/2];
    for(int i=0; i<a.length; i++) {
        a[i] = Big(St - S * Math.pow(d, Column(i+1) - Row(i+1)) * Math.pow(u, Row(i+1)), 0);
    }
    return a;
}

private static double[] BiOption(double[] a, int N, double R, double u, double d, double T) {
    int le = a.length;
    int b = Column(le);
    int pos = le - b - 1;
    double t = T / N;
    double p = (1 + R * t - d) / (u - d);
    for(int j=0; j<=N; j++) {
        for(int i=pos; i<le-1; i++) {
            double aa = a[i-b];
            a[i-b] = Big(aa, (1/(1+R*t)) *(a[i] * (1-p) + a[i+1] * p));
        }
        pos = pos - b;
        le = le - b - 1;
        b--;
    }
    return a;
}

private static double[] EuropeanBiOption(double[] a, int N, double R, double u, double d, double T) {
    int le = a.length;
    int b = Column(le);
    int pos = le - b - 1;
    double t = T / N;
    double p = (1 + R * t - d) / (u - d);
    for(int j=0; j<=N; j++) {
        for(int i=pos; i<le-1; i++) {
            double aa = a[i-b];
            a[i-b] = Big(aa, (1/(1+R*t)) *(a[i] * (1-p) + a[i+1] * p));
        }
        pos = pos - b;
        le = le - b - 1;
        b--;
    }
    return a;
}
\[ a[i-b] = \frac{1}{1+R*t} \times (a[i] \times (1-p) + a[i+1] \times p); \]

```java
pos = pos - b;
le = le - b - 1;
b--;
return a;
```

```java
private static double Big(double a, double b) {
if (a >= b) {
    return a;
} else {
    return b;
}
}
```

```java
private static int Column(int a) {
    int sum = 0;
    int k = 0;
    while (sum < a) {
        k++;
        sum = sum + k;
    }
    return k - 1;
}
```

```java
private static int Row(int a) {
    int b = a - (1 + Column(a)) * (Column(a)) / 2;
    return b - 1;
}
```

When you run it, you can type the values of the parameters that you want to give. The interface looks like:

```
Steps: 7
Initial price: 30
Strike price: 20
u: 1.2
d: 0.8
T: 0.6
R: 0.1
```

```java
(BinaryTree) American call Option price at 0 is: 12.459543700258086
(BinaryTree) American put Option price at 0 is: 1.3259947213678286
(BinaryTree) European call Option price at 0 is: 12.459543700258086
(BinaryTree) European put Option price at 0 is: 1.2996508484829912
```

Do you want show details of every point of American call option price? (1 is for Yes, 2 is for No):
2. A Java program for the Black-Scholes model by using the binomial approach. This program can compute American call, American put, European call and European put, and there is an extra function that can show option prices of every step if you are willing to know:

```java
import java.util.Scanner;

public class Black {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Steps:");
        int N = scan.nextInt();
        System.out.print("Initial price:");
        double S = scan.nextDouble();
        System.out.print("Strike price:");
        double St = scan.nextDouble();
        System.out.print("sigma:");
        double si = scan.nextDouble();
        System.out.print("T:");
        double T = scan.nextDouble();
        System.out.print("R:");
        double R = scan.nextDouble();
        double[] call = BlackCall(N, S, St, 1 + si * Math.sqrt(T / N), 1 - si * Math.sqrt(T / N));
        double[] options = BlackOptions(call, N, R, 1 + si * Math.sqrt(T / N), 1 - si * Math.sqrt(T / N), T, si);
        double[] put = BlackPut(N, S, St, 1 + si * Math.sqrt(T / N), 1 - si * Math.sqrt(T / N));
        double[] option = BlackOptions(put, N, R, 1 + si * Math.sqrt(T / N), 1 - si * Math.sqrt(T / N), T, si);
        double[] Ecall = BlackCall(N, S, St, 1 + si * Math.sqrt(T / N), 1 - si * Math.sqrt(T / N), T, si);
        double[] Eoptions = EuropeanBlackOptions(Ecall, N, R, 1 + si * Math.sqrt(T / N), 1 - si * Math.sqrt(T / N), T, si);
        double[] Eput = BlackPut(N, S, St, 1 + si * Math.sqrt(T / N), 1 - si * Math.sqrt(T / N));
        double[] Eoption = EuropeanBlackOptions(Eput, N, R, 1 + si * Math.sqrt(T / N), 1 - si * Math.sqrt(T / N), T, si);
        System.out.println();
        System.out.println("(Black by using binarytree)American call Option price at 0 is:");
        System.out.println(options[0]);
        System.out.println("(Black by using binarytree)American put Option price at 0 is:");
        System.out.println(option[0]);
        System.out.println("(Black by using binarytree)European call Option price at 0 is:");
        System.out.println(Eoptions[0]);
        System.out.println("(Black by using binarytree)European put Option price at 0 is:");
        System.out.println(Eoption[0]);
        System.out.println();
        System.out.println("Do you want show details of every point of American call option price? (1 is for Yes, 2 is for No):");
        int kk = scan.nextInt();
        if (kk == 1) {
            System.out.println();
            System.out.println("American call Option price at every step:");
            for (int i = 0; i < N; i++) {
                System.out.println(call[i]);
            }
        }
    }
}
```
for(int i=0;i<options.length;i++)
    System.out.println(options[i]);
    System.out.println();
else{
    System.out.println();
}
System.out.print("Do you want show details of every point of American put option price?
(1 is for Yes, 2 is for No):" + kkk=scan.nextInt();
if(kkk==1){
    for(int i=0;i<option.length;i++)
        System.out.println(option[i]);
    System.out.println();
} else{
    System.out.println();
}
System.out.print("Do you want show details of every point of European call option price?
(1 is for Yes, 2 is for No):" + kkkk=scan.nextInt());
if(kkkk==1){
    for(int i=0;i<Eoptions.length;i++)
        System.out.println(Eoptions[i]);
    System.out.println();
} else{
    System.out.println();
}
System.out.print("Do you want show details of every point of European put option price?
(1 is for Yes, 2 is for No):" + kkkkk=scan.nextInt());
if(kkkkk==1){
    for(int i=0;i<Eoption.length;i++)
        System.out.println(Eoption[i]);
    System.out.println();
} else{
    System.out.println();
}
public static double[] BlackCall(int N, double S, double St, double u, double d){
    double[] a=new double[(N+2)*(N+1)/2];
    for(int i=0;i<a.length;i++){
        a[i]=Big(S*Math.pow(d,Column(i+1)-Row(i+1))*Math.pow(u, Row(i+1))-St,0);
    }
    return a;
}

public static double[] BlackPut(int N, double S, double St, double u, double d){
    double[] a=new double[(N+2)*(N+1)/2];
    for(int i=0;i<a.length;i++){
        a[i]=Big(St-S*Math.pow(d,Column(i+1)-Row(i+1))*Math.pow(u, Row(i+1)),0);
    }
    return a;
}

private static double[] BlackOptions(double[] a, int N, double R, double u, double d, double T, double si){
    int le=a.length;
    int b=Column(le);
    int pos=le-b-1;
    double t=T/N;
    double p=(R*Math.sqrt(t))/(2*si)+0.5;
    for(int i=0;i<b;i++){
        for(int j=pos;j<le-1;j++){
            double aa=a[i-b];
            a[i-b]=Big(aa,(1/(1+R*t))*(a[i]*(1-p)+a[i+1]*p));
        }
        pos=pos-b;
        le=le-b-1;
        b--;
    }
    return a;
}

private static double[] EuropeanBlackOptions(double[] a, int N, double R, double u, double d, double T, double si){
    int le=a.length;

When you run it, you can type the values of the parameters that you want to give. The interface looks like:
3. A Java program for the Black-Scholes model by using the Black-Scholes option pricing formula. This program can compute European call and European put:

```java
import java.util.Scanner;
import org.apache.commons.math3.distribution.NormalDistribution;
public class NormalBlack {
    public static void main(String[] args){
        Scanner scan=new Scanner(System.in);
        System.out.print("Initial price:");
        double S=scan.nextDouble();
        System.out.print("Strike price:");
        double St=scan.nextDouble();
        System.out.print("sigma:");
        double si=scan.nextDouble();
        System.out.print("T:");
        double T=scan.nextDouble();
        double R=scan.nextDouble();
        double D1=(Math.log(S/St)+T*(R+(si*si/2)))/(si*Math.sqrt(T));
        double D2=(Math.log(S/St)+T*(R-(si*si/2)))/(si*Math.sqrt(T));
        NormalDistribution n = new NormalDistribution(0,1);
        double D1C = n.cumulativeProbability(D1);
        double D2C = n.cumulativeProbability(D2);
        double D1P = n.cumulativeProbability(-D1);
        double D2P = n.cumulativeProbability(-D2);
        double call=S*D1C-St*Math.pow(Math.E,-R*T)*D2C;
        double put=-S*D1P+St*Math.pow(Math.E,-R*T)*D2P;
        System.out.println("(Black-Schole formula)European call option is: "+call);
        System.out.println("(Black-Schole formula)European put option is: "+put);
        scan.close();
    }
}
```

Steps: 100
Initial price: 40
Strike price: 35
sigma: 0.2
T: 1
R: 0.1

(Black by using binarytree) American call Option price at 0 is: 8.754333530604834
(Black by using binarytree) American put Option price at 0 is: 0.5066160354277621
(Black by using binarytree) European call Option price at 0 is: 8.754333530604834
(Black by using binarytree) European put Option price at 0 is: 0.42522561202687265

Do you want show details of every point of American call option price? (1 is for Yes, 2 is for No):
When you run it, you can type the values of the parameters that you want to give. The interface looks like:

```
Initial price: 50
Strike price: 40
sigma: 0.25
T: 0.8
R: 0.1
(Black-Schole formula) European call option is: 13.461768371355213
(Black-Schole formula) European put option is: 0.38642222682063787
```

4. A Java program for sketching the relationship between different option prices under the N-step binomial tree model; 5 curves inside:

```java
import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;
import org.knowm.xchart.SwingWrapper;
import org.knowm.xchart.XYChart;
import org.knowm.xchart.XYChartBuilder;

public class graph {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Steps:");
        int N = scan.nextInt();
        System.out.print("Initial price:");
        double S = scan.nextDouble();
        System.out.print("Strike price:");
        double St = scan.nextDouble();
        System.out.print("u:");
        double u = scan.nextDouble();
        System.out.print("d:");
        double d = scan.nextDouble();
```
System.out.print("T: ");  
double T=scan.nextDouble();  
System.out.print("R:");  
double R=scan.nextDouble();  

XYChart chart = new XYChartBuilder().width(1200).height(800).title"The option price with different steps N".xAxisTitle("N(steps)").yAxisTitle("option price").build();  
chart.getStyler().setMarkerSize(1);  

List<Integer> xData = new ArrayList<Integer>();  
List<Double> yData = new ArrayList<Double>();  
List<Double> y1Data = new ArrayList<Double>();  
List<Double> y2Data = new ArrayList<Double>();  
List<Double> y3Data = new ArrayList<Double>();  
List<Double> y4Data = new ArrayList<Double>();  

for(int i=1;i<=N;i++)  
{  
double[] call=BioCall(i,S,St,u,d);  
xData.add(i);  
yData.add(getOption(call,i,R,u,d,T));  
}  

for(int j=1;j<=N;j++)  
{  
double[] put=BioPut(j,S,St,u,d);  
y1Data.add(getOption(put,j,R,u,d,T));  
}  

for(int ii=1;ii<=N;ii++)  
{  
double[] Ecall=BioCall(ii,S,St,u,d);  
y2Data.add(EuropeanGetOption(Ecall,ii,R,u,d,T));  
}  

for(int jj=1;jj<=N;jj++)  
{  
double[] Eput=BioPut(jj,S,St,u,d);  
y3Data.add(EuropeanGetOption(Eput,jj,R,u,d,T));  
}  

for(int iii=1;iii<=N;iii++)  
{  
double[] variance=BioCall(iii,S,St,u,d);  
y4Data.add(BiAmericanEuropean(variance,iii,R,u,d,T));  
}

chart.addSeries("BT American call", xData, yData);//add them into chart  
chart.addSeries("BT American put", xData, y1Data);  
chart.addSeries("BT European call", xData, y2Data);  
chart.addSeries("BT European put", xData, y3Data);  
chart.addSeries("BT,Difference value between American call and European call", xData, y4Data);  
new SwingWrapper<>(chart).displayChart("BinomialTree");
```java
private static double BiAmericanEuropean(double[] a, int N, double R, double u, double d, double T)
{
    return getOption(a,N,R,u,d,T)-EuropeanGetOption(a,N,R,u,d,T);
}

private static double getOption(double[] a, int N, double R, double u, double d, double T)
{
    int le=a.length;
    int b=Column(le);
    int pos=le-b-1;
    double t=T/N;
    double p=(1+R*t-d)/(u-d);
    for(int j=0;j<=N;j++)
    {
        for(int i=pos;i<le-1;i++)
        {
            double aa=a[i-b];
            a[i-b]=Big(aa,(1/(1+R*t))*(a[i]*(1-p)+a[i+1]*p));
        }
        pos=pos-b;
        le=le-b-1;
        b--;
    }
    return a[0];
}

public static double[] BioCall(int N, double S, double St,double u, double d)
{
    double[] a=new double[(N+2)*(N+1)/2];
    for(int i=0;i<a.length;i++)
    {
        a[i]=Big(S*Math.pow(d,Column(i+1)-Row(i+1))*Math.pow(u, Row(i+1))-St,0);
    }
    return a;
}

public static double[] BioPut(int N, double S, double St,double u, double d)
{
    double[] a=new double[(N+2)*(N+1)/2];
    for(int i=0;i<a.length;i++)
    {
        a[i]=Big(St-S*Math.pow(d,Column(i+1)-Row(i+1))*Math.pow(u, Row(i+1))-St,0);
    }
    return a;
}

private static double EuropeanGetOption(double[] a, int N, double R, double u, double d, double T)
{
    int le=a.length;
    int b=Column(le);
    int pos=le-b-1;
    double t=T/N;
    double p=(1+R*t-d)/(u-d);
    for(int j=0;j<=N;j++)
    {
        for(int i=pos;i<le-1;i++)
        {
            double aa=a[i-b];
            a[i-b]=Big(aa,(1/(1+R*t))*(a[i]*(1-p)+a[i+1]*p));
        }
        pos=pos-b;
        le=le-b-1;
        b--;
    }
    return a[0];
}
```

int pos=le-b-1;
double t=T/N;
double p=(1*R*t-d)/(u-d);
for(int j=0;j<=N;j++){
    for(int i=pos;i<le-1;i++){
        a[i-b]=(1/(1+R*t))*(a[i]*(1-p)+a[i+1]*p);
    }
    pos=pos-b;
    le=le-b-1;
    b--;
}
return a[0];
}

private static double Big(double a, double b){
    if(a>=b){
        return a;
    }
    else{
        return b;
    }
}

private static int Column(int a){
    int sum=0;
    int k=0;
    while(sum<a){
        k++;
        sum=sum+k;
    }
    return k-1;
}

private static int Row(int a){
    int b=a-(1+Column(a))*(Column(a))/2;
    return b-1;
}

When you run it, you can type the values of the parameters that you want to give. The interface looks like:
5. A Java program for sketching the relationship between different option prices under the Black-Scholes model by using the binomial approach and the Black-Scholes option pricing formulas; 7 curves inside:

```java
import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;
import org.apache.commons.math3.distribution.NormalDistribution;
import org.knowm.xchart.SwingWrapper;
import org.knowm.xchart.XYChart;
import org.knowm.xchart.XYChartBuilder;

public class graphBlack {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Steps:");
        int N = scan.nextInt();
        System.out.print("Initial price:");
        int S = scan.nextInt();
        System.out.print("Strike price:");
        int K = scan.nextInt();
        System.out.print("u:");
        double u = scan.nextDouble();
        System.out.print("d:");
        double d = scan.nextDouble();
        System.out.print("r:");
        double r = scan.nextDouble();
        System.out.print("T:");
        double T = scan.nextDouble();

        // Code to calculate option prices using binomial approach
        // and Black-Scholes option pricing formulas
    }
}
```
double S=scan.nextDouble();
System.out.print("Strike price:");
double St=scan.nextDouble();
System.out.print("sigma:");
double si=scan.nextDouble();
System.out.print("T:");
double T=scan.nextDouble();
System.out.print("R:");
double R=scan.nextDouble();

double D1=(Math.log(S/St)+T*(R+(si*si/2)))/(si*Math.sqrt(T));
double D2=(Math.log(S/St)+T*(R-(si*si/2)))/(si*Math.sqrt(T));
NormalDistribution n = new NormalDistribution(0,1);
double D1C =n.cumulativeProbability(D1);
double D2C =n.cumulativeProbability(D2);
double D1P =n.cumulativeProbability(-D1);
double D2P =n.cumulativeProbability(-D2);
double Fcall=S*D1C-St*Math.pow(Math.E,-R*T)*D2C;
double Fput=-S*D1P+St*Math.pow(Math.E,-R*T)*D2P;

XYChart chart = new XYChartBuilder().width(1200).height(800)
   .title("The option price with different steps N").
   xAxisTitle("N(steps)").yAxisTitle("option price").build();
chart.getStyler().setMarkerSize(1);
List<Integer> xData =new ArrayList<Integer>();
List<Double> yData =new ArrayList<Double>();
List<Double> y1Data=new ArrayList<Double>();
List<Double> y2Data=new ArrayList<Double>();
List<Double> y3Data=new ArrayList<Double>();
List<Double> y4Data=new ArrayList<Double>();
List<Double> y5Data=new ArrayList<Double>();
List<Double> y6Data=new ArrayList<Double>();
for(int i=1;i<=N;i++){
    double[] call=BlackCall(i,S,St,1+si*Math.sqrt(T/i),1-si*Math.sqrt(T/i));
    xData.add(i);
    yData.add(GetBlackOptions(call,i,R,1+si*Math.sqrt(T/i),1-si*Math.sqrt(T/i),T,si));
}
for(int j=1;j<=N;j++){
    double[] put=BlackPut(j,S,St,1+si*Math.sqrt(T/j),1-si*Math.sqrt(T/j));
    y1Data.add(GetBlackOptions(put,j,R,1+si*Math.sqrt(T/j),1-si*Math.sqrt(T/j),T,si));
}
for(int ii=1;ii<=N;ii++){
    double[] Ecall=BlackCall(ii,S,St,1+si*Math.sqrt(T/ii),1-si*Math.sqrt(T/ii));
}
y2Data.add(EuropeanBlackOptions(Ecall, ii, R, 1 + si * Math.sqrt(T / ii), 1 - si * Math.sqrt(T / ii), T, si));
}
for(int jj=1;jj<=N;jj++){
    double[] Eput=BlackPut(jj, S, St, 1 + si * Math.sqrt(T / jj), 1 - si * Math.sqrt(T / jj));
    y3Data.add(EuropeanBlackOptions(Eput, jj, R, 1 + si * Math.sqrt(T / jj), 1 - si * Math.sqrt(T / jj), T, si));
}
for(int iii=1;iii<=N;iii++){
    y4Data.add(Fcall);
}
for(int jjj=1;jjj<=N;jjj++){
    y5Data.add(Fput);
}
for(int iiii=1;iiii<=N;iiii++){
    double[] variance=BlackCall(iiii, S, St, 1 + si * Math.sqrt(T / iiii), 1 - si * Math.sqrt(T / iiii));
    y6Data.add(AmericanAndEuropean(variance, iiii, R, 1 + si * Math.sqrt(T / iiii), 1 - si * Math.sqrt(T / iiii), T, si));
}

chart.addSeries("BS(by using BT) American call", xData, yData);
chart.addSeries("BS(by using BT) American put", xData, y1Data);
chart.addSeries("BS(by using BT) European call", xData, y2Data);
chart.addSeries("BS(by using BT) European put", xData, y3Data);
chart.addSeries("BS(BS formula) European call(no use of N)", xData, y4Data);
chart.addSeries("BS(BS formula) European put(no use of N)", xData, y5Data);
chart.addSeries("BS, Difference value between American call and European call(by using BT)", xData, y6Data);
new SwingWrapper<>(chart).displayChart("Black-Scholes");

}
return a;
}
private static double AmericanAndEuropean(double[] a, int N, double R, double u,
double d, double T, double si){
return GetBlackOptions(a,N,R,u,d,T,si)-EuropeanBlackOptions(a,N,R,u,d,T,si);
}
private static double GetBlackOptions(double[] a, int N, double R, double u,
double d, double T, double si){
int le=a.length;
int b=Column(le);
int pos=le-b-1;
double t=T/N;
double p=(R*Math.sqrt(t))/(2*si)+0.5;
for(int j=0;j<=N;j++){
for(int i=pos;i<le-1;i++){
double aa=a[i-b];
a[i-b]=Big(aa,(1/(1+R*t))*(a[i]*(1-p)+a[i+1]*p));
}
pos=pos-b;
le=le-b-1;
b--;
}
return a[0];
}
private static double EuropeanBlackOptions(double[] a, int N, double R, double u,
double d, double T, double si){
int le=a.length;
int b=Column(le);
int pos=le-b-1;
double t=T/N;
double p=(R*Math.sqrt(t))/(2*si)+0.5;
for(int j=0;j<=N;j++){
for(int i=pos;i<le-1;i++){
a[i-b]=(1/(1+R*t))*(a[i]*(1-p)+a[i+1]*p);
}
pos=pos-b;
le=le-b-1;
b--;
}
return a[0];
}
private static double Big(double a, double b){
if(a>=b){
return a;
}
} else{
    return b;
}

private static int Column(int a){
    int sum=0;
    int k=0;
    while(sum<a){
        k++;
        sum+=sum+k;
    }
    return k-1;
}

private static int Row(int a){
    int b=a-(1+Column(a))*(Column(a))/2;
    return b-1;
}

When you run it, you can type the values of the parameters that you want to give. The interface looks like:

| Steps: 50 |
| Initial price: 50 |
| Strike price: 46 |
| sigma: 0.2 |
| T: 1 |
| R: 0.1 |
References


