Comparison of Pricing Algorithms for Asian Options

Moira Perben
Joris Esch
Cover illustration: The full grid solution for the pricing of a European style
Asian call option. Results are from Večeř’s PDE with $S_0 = 100$, $K = 90$, 
$\sigma = 0.3$, $r = 0.15$, $M = 25$, $N = 50$, $\theta = 0.5$. 
Contents

1 Introduction 5

2 Monte-Carlo method 6
  2.1 Antithetic variables 6
  2.2 Experiments 6

3 Binomial tree method 6
  3.1 Verification 8
  3.2 Experiments 8

4 Večer’s unified PDE method 9
  4.1 Verification 9
  4.2 Experiments 10

5 Conclusion 10
1 Introduction

Asian options are path-dependant, exotic options. This means that the final payoff depends on the history of the random walk followed by the asset price over a certain period. In an Asian option, specifically, the payoff depends on the average over time of the asset price. Several factors affect the definition of average:

- The period of averaging
- Arithmetic or geometric averaging
- Weighted or unweighted averaging

The difficulty of pricing such an option lies in its path dependency. For pricing, we need a third variable, noted \( I \), in addition to asset price \( S \) and time \( t \), to measure the relevant path-dependant quantity. For an arithmetic Asian option, an appropriate quantity is:

\[
I(t) = \int_0^t S(\tau)d\tau \quad \text{continuous time averaging}
\]

\[
I(m) = \sum_{i=0}^m S(t_i) \quad \text{discrete time averaging}
\]

We can then write the time average of the asset price as \( I(T)/T \) or \( I(T)/M \), with \( T \) the time to expiry and \( M \) the total number of samples. The sampling instances \( t_i \) will be accurately specified in the contract.

The payoff contract for an Asian call \( c(T) \), and an Asian put \( p(T) \) is of the form:

\[
c(T) = (\langle S_T \rangle - K_1 S_T - K_2)^+, \\
p(T) = (K_2 - K_1 S_T - \langle S_T \rangle)^+,
\]

with \( \langle S_T \rangle \) a continuous or discrete average over time of an asset price \( S \). When \( K_1 = 0 \) and \( K_2 > 0 \) this is called a fixed strike or average price option. When \( K_1 > 0 \) and \( K_2 = 0 \) this is called a floating strike or average strike option.

The prices \( c(0) \) and \( p(0) \) of European style options are thus given by

\[
c(0) = e^{-rT} E \left[ (\langle S_T \rangle - K_1 S_T - K_2)^+ \right], \\
p(0) = e^{-rT} E \left[ (K_2 - K_1 S_T - \langle S_T \rangle)^+ \right],
\]

with \( r \) the interest rate and \( T \) the time to expiry. The underlying asset price is assumed to follow geometric Brownian motion:

\[
dS_t = S_t \nu dt + S_t \sigma dW_t, \quad (1)
\]

with \( t \) the time, \( \nu \) the drift and \( \sigma \) the volatility. The increment \( dW_t \) follows a Wiener process: \( dW_t = Z_t \sqrt{dt} \) with \( Z_t \) a random variable following a standard normal distribution.
The put-call parity for Asian options is given by:

\[ c(0) - p(0) = \frac{-S_0}{rT}e^{-rT} - S_0K_1 - K_2e^{-rT}. \]  

(2)

2 Monte-Carlo method

The Monte-Carlo simulation method is used to predict the behavior of the underlying stock price until the expiry of the option. This is done by generating a large number of instances of random paths following the Brownian motion model as in (1). This involves the generation of normal random variables, which was done using the Box-Muller algorithm. During the construction of the path, a running sum of the stock price is maintained. This running sum is used to calculate the final payoff.

Each path will give a different payoff at expiry time \( t = T \). After discounting the payoff, different option prices are calculated at time \( t = 0 \). By the strong law of large numbers, the mean of these option prices will give an estimate of the Asian option price.

2.1 Antithetic variables

In our implementation, we used antithetic variables as a variance reduction technique. Given a random variable \( Z_1 \) from a normal distribution \( N(0, 1) \), a second correlated random variable \( Z_2 = -Z_1 \) is generated. This implies:

\[ \text{Var}(\frac{Z_1 + Z_2}{2}) = \frac{1}{4}[\text{Var}(Z_1) + \text{Var}(Z_2) + 2\text{Cov}(Z_1, Z_2)]. \]

Since the covariance of \( Z_1 \) and \( Z_2 \) is negative, \( \text{Cov}(Z_1, Z_2) = -1 \), and the payoff function is a monotonic function of \( Z_t \), this technique results in a variance reduction.

2.2 Experiments

In a first experiment we use the Monte-Carlo method for the pricing of an Asian average price option. As a reference, we use the results from Table 2 in [1]. The results with the Monte-Carlo method are shown in Table 1, for call prices and in Table 2, for put prices. The put prices were calculated using the put-call parity for Asian options (2) and were later verified using a binomial tree method.

The Monte-Carlo experiments were done with 2000 Monte-Carlo paths.

3 Binomial tree method

A binomial tree method was proposed by Hull [2]. The paper is both detailed and complete, and allows for full reproduction of the results. The tree based method was implemented using the C-language.
<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>$K$</th>
<th>MC $\Delta t = 10^{-2}$</th>
<th>MC $\Delta t = 10^{-3}$</th>
<th>MC $\Delta t = 10^{-4}$</th>
<th>Vecer</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.05</td>
<td>95</td>
<td>11.1635</td>
<td>11.1011</td>
<td>11.0945</td>
<td>11.094</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>6.86387</td>
<td>6.80212</td>
<td>6.79467</td>
<td>6.795</td>
</tr>
<tr>
<td></td>
<td>105</td>
<td>2.80286</td>
<td>2.75306</td>
<td>2.74112</td>
<td>2.744</td>
</tr>
<tr>
<td>0.10</td>
<td>90</td>
<td>15.4682</td>
<td>15.4069</td>
<td>15.3979</td>
<td>15.399</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>7.09646</td>
<td>7.03610</td>
<td>7.02183</td>
<td>7.029</td>
</tr>
<tr>
<td></td>
<td>110</td>
<td>1.45477</td>
<td>1.42986</td>
<td>1.41089</td>
<td>1.415</td>
</tr>
<tr>
<td>0.20</td>
<td>90</td>
<td>15.7113</td>
<td>15.6461</td>
<td>15.6271</td>
<td>15.643</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>8.46689</td>
<td>8.43119</td>
<td>8.39651</td>
<td>8.412</td>
</tr>
<tr>
<td></td>
<td>110</td>
<td>3.62751</td>
<td>3.62751</td>
<td>3.55391</td>
<td>3.560</td>
</tr>
<tr>
<td>0.30</td>
<td>90</td>
<td>16.5732</td>
<td>16.5385</td>
<td>16.4844</td>
<td>16.546</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>10.2964</td>
<td>10.2560</td>
<td>10.1939</td>
<td>10.215</td>
</tr>
<tr>
<td></td>
<td>110</td>
<td>5.82246</td>
<td>5.77815</td>
<td>5.72585</td>
<td>5.736</td>
</tr>
</tbody>
</table>

Table 1: Average price call, for $r = 0.15, S_0 = 100$ and $T = 1$.

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>$K$</th>
<th>MC $\Delta t = 10^{-2}$</th>
<th>MC $\Delta t = 10^{-3}$</th>
<th>MC $\Delta t = 10^{-4}$</th>
<th>Vecer</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.05</td>
<td>95</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>0.00288</td>
<td>0.00477</td>
<td>0.00252</td>
<td>0.00445</td>
</tr>
<tr>
<td></td>
<td>105</td>
<td>0.24288</td>
<td>0.25333</td>
<td>0.25751</td>
<td>0.25699</td>
</tr>
<tr>
<td>0.10</td>
<td>90</td>
<td>0.00074</td>
<td>0.00227</td>
<td>0.00062</td>
<td>0.00140</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>0.23184</td>
<td>0.25974</td>
<td>0.23484</td>
<td>0.23845</td>
</tr>
<tr>
<td></td>
<td>110</td>
<td>3.16346</td>
<td>3.26335</td>
<td>3.24720</td>
<td>3.23150</td>
</tr>
<tr>
<td>0.20</td>
<td>90</td>
<td>0.24649</td>
<td>0.26513</td>
<td>0.23881</td>
<td>0.24537</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>1.56682</td>
<td>1.68640</td>
<td>1.63514</td>
<td>1.62140</td>
</tr>
<tr>
<td></td>
<td>110</td>
<td>5.29180</td>
<td>5.43235</td>
<td>5.41576</td>
<td>5.37650</td>
</tr>
<tr>
<td>0.30</td>
<td>90</td>
<td>1.09766</td>
<td>1.19496</td>
<td>1.14620</td>
<td>1.11840</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>3.35321</td>
<td>3.50120</td>
<td>3.47351</td>
<td>3.42440</td>
</tr>
<tr>
<td></td>
<td>110</td>
<td>7.45411</td>
<td>7.62838</td>
<td>7.61314</td>
<td>7.55250</td>
</tr>
</tbody>
</table>

Table 2: Average price put, for $r = 0.15, S_0 = 100$ and $T = 1$. 
<table>
<thead>
<tr>
<th>$h$</th>
<th>Option Type</th>
<th>$N = 20$</th>
<th>$N = 40$</th>
<th>$N = 60$</th>
<th>$N = 80$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.100</td>
<td>American</td>
<td>6.10417</td>
<td>6.27503</td>
<td>6.35344</td>
<td>6.40041</td>
</tr>
<tr>
<td></td>
<td>European</td>
<td>5.39012</td>
<td>5.40730</td>
<td>5.41280</td>
<td>5.41556</td>
</tr>
<tr>
<td>0.050</td>
<td>American</td>
<td>5.56950</td>
<td>5.63173</td>
<td>5.65443</td>
<td>5.66848</td>
</tr>
<tr>
<td></td>
<td>European</td>
<td>5.33991</td>
<td>5.35528</td>
<td>5.36051</td>
<td>5.36318</td>
</tr>
<tr>
<td>0.010</td>
<td>American</td>
<td>5.34439</td>
<td>5.38896</td>
<td>5.40586</td>
<td>5.41469</td>
</tr>
<tr>
<td></td>
<td>European</td>
<td>5.25356</td>
<td>5.26945</td>
<td>5.27440</td>
<td>5.27700</td>
</tr>
<tr>
<td>0.005</td>
<td>American</td>
<td>5.32495</td>
<td>5.36135</td>
<td>5.37761</td>
<td>5.38632</td>
</tr>
<tr>
<td></td>
<td>European</td>
<td>5.23986</td>
<td>5.25431</td>
<td>5.25956</td>
<td>5.26220</td>
</tr>
<tr>
<td>0.003</td>
<td>American</td>
<td>5.31597</td>
<td>5.35466</td>
<td>5.36671</td>
<td>5.35080</td>
</tr>
<tr>
<td></td>
<td>European</td>
<td>5.23627</td>
<td>5.24842</td>
<td>5.25339</td>
<td>5.25602</td>
</tr>
</tbody>
</table>

Table 3: Average price call, for $r = 0.1$, $\sigma = 0.1$, $T = 1$, $S_0 = 100$, $K = 100$.

### 3.1 Verification

The original paper lists numeric results in Exhibits 4 and 5. We are able to exactly reproduce these results.

### 3.2 Experiments

Accurate results using this binomial method can be achieved only with a small value for $h$. This parameter controls the distribution of possible averages in the array $S_0 e^{m h}$ with $m$ an integer between $m_{\text{min}}$ and $m_{\text{max}}$. To show this critical dependence on $h$, we use the input data from Table 4 in [3]. We test our code for various values of $h$, the discretization step size of the average array, and $N$, the number of time steps. Results are listed in Table 3. The runtime for each of these experiments is well below one second on a modern workstation.

The results clearly expose the deficiencies of the binomial tree method. We find European call option prices that are higher than American option prices, albeit for different values of $h$ and $N$. The values of $h$ and $N$ need to be chosen very carefully to get an accurate answer. We find 5.38 for the American option and 5.25 for the European option for $N = 400$ and $h = 0.0005$.

The inaccuracy of the binomial tree method is also shown in Fig. 1. The figures depict the convergence as a function of $h$ and $N$. Accurate pricing is only possible for small values of $h$. The convergence as a function of $N$ is not monotonic, if it can at all be observed. The calculated price rises steadily from below 2.560 at $N = 60$ to above 5.267 at $N = 200$. At a certain point, a more accurate answer can simply not be found by increasing $N$ without decreasing $h$.

The smallest value of $h$ that is used in the original paper is 0.005. The authors claim penny accuracy for this value of $h$ with a large enough number of $N$. This accuracy was not observed in this example.
Figure 1: Dependency of option price on $h$, left, and $N$, right. The calculated prices of a European style Asian average price option are plotted, with $r = 0.15$, $\sigma = 0.1$, $T = 1$, $S_0 = 100$, $K = 100$, $h = 0.005$ (or varying) and $N = 100$ (or varying).

4 Večer’s unified PDE method

Recently, a PDE based pricing method for Asian options was presented as a specific case in a more general framework by Večer [1]. The framework allows pricing of various common and exotic options [4].

The method does not rely on the methodology outlined in the introduction; no third variable $I$ is introduced.

Time derivatives are approximated using the theta method, a blend of the forward and backward difference schemes. By varying the parameter theta one can choose different numeric difference schemes. For theta zero, we have the fully explicit scheme, for theta one, we have the fully implicit scheme. The Crank-Nicolson scheme corresponds to theta one-half.

The method was implemented in Matlab, because the implicit methods used to solve the PDE give rise to a set of linear equations. These equations are easily, though not efficiently, solved using Matlab.

The boundary conditions at expiry can easily be determined by the payoff function. At the lower limit of the asset price range, we enforce an option price of zero. At the upper limit of the asset price range, linear interpolation is used. The linear assumption effectively corresponds to zero Gamma conditions.

4.1 Verification

We verified numerical data from Table 1 in [4], using 200 space points, $M = 200$, and 400 time points, $N = 400$, for the discrete solution of the PDE. We reached agreement with three digits of accuracy.

We exactly reproduce all data in [5].
4.2 Experiments

The solution of the PDE does not depend on $S_0$, hence we can interpret $S_0$ as a variable. We characterize the relationship between $Z_0$, $S_0$, $X_0$ and $u(0,Z_0)$. We have, for a call option:

\[
X_0 = S_0 - K \\
Z_0 = \frac{X_0}{S_0}
\]

which leads to:

\[
Z_0 = \frac{S_0 - K}{S_0} 
\] (3)

We plot this relationship on the left in Fig. 2. On the right we plot the relationship between $Z_0$ and the option price at $t = 0$. This curve is calculated numerically using the PDE method, on a finite grid. Cubic spline interpolation is used to evaluate the option price at $t = 0$. The arrows on the plot indicate the calculation flow. The calculated call price is $816.514$.

In a second experiment, we studied the convergence of the PDE method. We increased the number of time steps $N$ between 20 and 800. The results are shown in Fig. 3. The error goes down linear with $N$, as expected for the Crank-Nicolson method.

5 Conclusion

We compare various pricing methods for Asian options.

The Monte-Carlo method is capable of accurately pricing Asian options. Pricing of Asian Options using the Monte-Carlo does not present any specific
Figure 3: Convergence of the numeric PDE method with a call price calculation with $S_0 = 100$, $K = 90$, $T = 1$, $\sigma = 0.3$, $r = 0.15$, $M = 200$, $\theta = 0.5$. 
difficulties or problems. The drawbacks of the Monte-Carlo method are known. The method is slow, and results depend on the quality of the random number generator. We do feel however that the Monte-Carlo method remains the tool of choice for providing reference prices to be used in the development and testing of possibly better and faster pricing routines. Almost all articles found in literature use results from the Monte-Carlo method in this way.

We implemented the binomial tree method for the pricing of Asian options, as originally presented [2]. We observed serious drawbacks, inaccuracies and limitations of the proposed method. The convergence as a function of the number of time steps is not satisfactory. The method is, in this form, not usable as an accurate pricing tool.

The PDE based method is very robust. The main weakness of the PDE based methods are the choice of boundary conditions for truly difficult test cases. Also, one cannot easily model American options using PDE methods. Overall, the PDE based method is fast and accurate, exhibits smooth convergence and is a very practical technique for use in real-life applications.

References


