# Numerical Comparison to the Black-Scholes Model

20180617 You SeungWoo

June 29, 2022

**Abstract**

The Black-Scholes model calculates the price of a an option in the stock market. It can be solved with hands, but also by computing. In this article, the main concern is to find better numerical ways of the model. Analytic solution and some problem solving techniques are introduced. Since the model is PDE contains $s$ and $t$ coordinates, changing to ODE is essential to get the numerical solution. Different method was used to obtain the $s$-coordinate value. Finially, find the absolute error and compare them at each time. Global error also analyzed.

## 1 Introduction

Black-Scholes model is given second PDE as following:

$$\frac{\partial V}{\partial t} + rs\frac{\partial V}{\partial s} + \frac{1}{2}\sigma^2 s^2 \frac{\partial^2 V}{\partial s^2} - rV = 0, \qquad 0 < s < \infty,\ 0 < t < T.$$

$V(s,t)$ is a soulution of the equation which is the price of an option with respected to stock price and time in years. $r$ and $\sigma$ is constant, risk-free interest rate and voltility each. There are many option conditions to satisfy this, but just consider a call option. For the stock price $s$ and the strike price constant $k$, the final price of a call option is determined by them. If $s < k$, then the value of an option is lost, otherwise it will have the value of $s - k$. So the final boundary condition is given as

$$V(s,T) = \begin{cases} s - k & \text{if } s \geq k \\ 0 & \text{if } s < k. \end{cases}$$

This model contains simple boundary condition but PDE looks complicated second order. However, it can turn into one-dimensional heat equation. Therefore the model can be solved analytically. The numerical solution can be applied to both changed and original forms, and different method can be taken each $s$ and $t$ coordinates. Although $s$ can increase infinitely, but it sufficies to take three or four times of $k$. Hence set the boundary $3k \leq S \leq 4k$ for numeircal calculation.

## 2 Analytic Solution

Guess the form of $V(s,t)$. Suppose $V(s,t) = e^{ax+b\tau}U(x,\tau)$, where $x = \ln s,\ \tau = \frac{\sigma^2}{2}(T-t),\ a = \frac{1}{2} - \frac{r}{\sigma^2}$, and $b = -(\frac{1}{2} + \frac{r}{\sigma^2})^2$. Let the final boundary condition $V(s,T) = F(s)$. Then

$$\begin{aligned}
\frac{\partial V}{\partial t} &= -\frac{\sigma^2}{2}\left(be^{ax+b\tau}U + e^{ax+b\tau}\frac{\partial U}{\partial \tau}\right),\\
\frac{\partial V}{\partial s} &= ae^{ax+b\tau}U + e^{ax+b\tau}\frac{\partial U}{\partial x},\\
\frac{\partial^2 V}{\partial s^2} &= a(a-1)e^{a(x-2)+b\tau}U + (2a-1)e^{a(x-2)+b\tau}\frac{\partial U}{\partial x} + e^{a(x-2)+b\tau}\frac{\partial^2 U}{\partial x^2}.
\end{aligned}$$

Replacing $s$ by $e^x$ and canceling out $e^{ax+b\tau}$, then the equation is changed as

$$\begin{aligned}
&-\frac{\sigma^2}{2}\left(bU + \frac{\partial U}{\partial \tau}\right) + ra\left(U + \frac{\partial U}{\partial x}\right) + \frac{\sigma^2}{2}\left(a(a-1)U + (2a-1)\frac{\partial U}{\partial x} + \frac{\partial^2 U}{\partial x^2}\right) - rU = 0\\
\Rightarrow &\frac{\sigma^2}{2}\frac{\partial^2 U}{\partial x^2} + \left(r + \frac{\sigma^2}{2}(2a-1)\right)\frac{\partial U}{\partial x} + \left(-\frac{\sigma^2}{2}b + ra + \frac{\sigma^2}{2}a(a-1) - r\right)U = \frac{\sigma^2}{2}\frac{\partial U}{\partial \tau}\\
\Rightarrow &\frac{\partial^2 U}{\partial x^2} = \frac{\partial U}{\partial \tau},\ \text{ because }\ r + \frac{\sigma^2}{2}(2a-1) = 0 \text{ and } -\frac{\sigma^2}{2}b + ra + \frac{\sigma^2}{2}a(a-1) - r = 0.
\end{aligned}$$

The initial condition is

$$\begin{aligned}
F(s) &= e^{ax}U(x,0)\\
\Rightarrow U(x,0) &= e^{-ax}F(s) = e^{-ax}F(e^x).
\end{aligned}$$

The model is given as following:

$$\frac{\partial U}{\partial \tau} = \frac{\partial^2 U}{\partial x^2}, \qquad -\infty < x < \infty,\ 0 < \tau < \frac{\sigma^2}{2}T.$$

This is the same as one-dimensional heat equation. The solution of heat equation is already known, so it can be solved. The general solution is

$$V(s,t) = \frac{e^{r(T-t)}}{\sigma\sqrt{2\pi(T-t)}}\int_0^\infty e^{-\frac{\left(ln(s/x)+(r-\sigma^2/2)(T-t)\right)^2}{2\sigma^2(T-t)}}F(x)\frac{dx}{x}.$$

For the given initial condition(i.e. for the call option case),

$$\begin{aligned}
V(s,t) &= \frac{e^{r(T-t)}}{\sigma\sqrt{2\pi(T-t)}}\int_k^\infty e^{-\frac{\left(\ln(s/x)+(r-\sigma^2/2)(T-t)\right)^2}{2\sigma^2(T-t)}}(x-k)\frac{dx}{x}\\
&= \frac{e^{r(T-t)}}{\sigma\sqrt{2\pi(T-t)}}\int_{\ln k}^\infty e^{-\frac{\left(-u+\ln s+(r-\sigma^2/2)(T-t)\right)^2}{2\sigma^2(T-t)}}(e^u-k)du\\
&= s\frac{1}{\sqrt{2\pi}}\int_{-\frac{\ln\frac{s}{k}+\left(r+\frac{1}{2}\sigma^2\right)(T-t)}{\sigma\sqrt{T-t}}}^\infty e^{-\frac{1}{2}x_1^2}dx_1 - ke^{-r(T-t)}\frac{1}{\sqrt{2\pi}}\int_{-\frac{\ln\frac{s}{k}+\left(r-\frac{1}{2}\sigma^2\right)(T-t)}{\sigma\sqrt{T-t}}}^\infty e^{-\frac{1}{2}x_2^2}dx_2\\
&= s\Phi(d_1) - ke^{-r(T-t)}\Phi(d_2),
\end{aligned}$$

where $d_1 = \frac{\ln\frac{s}{k}+\left(r+\frac{1}{2}\sigma^2\right)(T-t)}{\sigma\sqrt{T-t}}, d_2 = \frac{\ln\frac{s}{k}+\left(r-\frac{1}{2}\sigma^2\right)(T-t)}{\sigma\sqrt{T-t}} = d_1 - \sigma\sqrt{T-t}$, and $\Phi(x)$ is a cumulative distribution function of a standard normal distribution.

# 3 Numerical Solution

For most cases, it is difficult to obtain differential equation directly. Therefore, approximation is needed for the PDE as a function with respected to $s$ for each fixed $t$ to use the ODE numerical solution. First, define $\tau = T - t$. Then the PDE is

$$\frac{\partial V}{\partial \tau} = rs\frac{\partial V}{\partial s} + \frac{1}{2}\sigma^2 s^2 \frac{\partial^2 V}{\partial s^2} - rV, \qquad 0 < s < \infty,\ 0 < \tau < T, \tag{1}$$

with the initial condition

$$V(s,0) = \begin{cases} s-k & \text{if } s \geq k \\ 0 & \text{if } s < k. \end{cases}$$

Assume that $s$ is bounded by $S$: $0 < s < S$. Set new boundary conditions:

$$\begin{aligned} V(0,\tau) &= 0, \\ V(S,\tau) &\simeq S. \end{aligned}$$

Let $W(s,\tau)$ be a numerical solution. Take finite point $(s_i,\tau_j)$, $i = 0,1,2,\dots,N$, $j = 0,1,2,\dots,M$, and let $W(s_i,\tau_j) = W_{i,j}$. To get each $W_{i,j}$, derive the approximation form with respect to $s$ at first by applying some formulae to given $V(s,\tau_j)$ for fixed $\tau$. Then a numerical function $W(s,\tau_j)$ for fixed $j$ is found, so the PDE turn into ODE with respect to $\tau$. Next, take appropriate method to the ODE. This implies all numerical soultions are found because next numerical function $W(s,\tau_{j+1})$ comes out from the previous numerical function $W(s,\tau_j)$. In this section, just apply Euler's method.
Take the uniform grid $ds = l$ and $d\tau = h$. Introduce the set of points $s_i = il$ for $i = 0,1,2,\dots,N$, and $\tau_j = jh$ for $j = 0,1,2,\dots,M$. Then $s_N = Nl = S$, $\tau_M = Mh = T$. Consider the solution for the constant $k = 50$, $r = 0.12$, $\sigma = 0.09$, $S = 150$, $T = 1$, $N = 300$, $M = 8000$, so that $l = 0.5$, $h = 0.000125$.

## 3.1 Midpoint Formula

For the uniform grid, each partial derivative can be approximated as

$$\begin{aligned} \frac{\partial V}{\partial s}(s_i,\tau_j) &\simeq \frac{W_{i+1,j} - W_{i-1,j}}{2l}, \\ \frac{\partial^2 V}{\partial s^2}(s_i,\tau_j) &\simeq \frac{W_{i+1,j} - 2W_{i,j} + W_{i-1,j}}{l^2} \end{aligned}$$

by the mid-point formula. Plugging that into PDE$^{(1)}$ gives ODE with respected to $\tau$:

$$\frac{\partial V}{\partial \tau}(s_i,\tau_{j+1}) \simeq \frac{\partial W}{\partial \tau}(s_i,\tau_{j+1}) = rs_i\frac{W_{i+1,j} - W_{i-1,j}}{2l} + \frac{1}{2}\sigma^2 s_i^2 \frac{W_{i+1,j} - 2W_{i,j} + W_{i-1,j}}{l^2} - rW_{i,j}, \tag{2}$$

with the initial and boundary conditions

$$\begin{aligned} W(s,0) &= \begin{cases} s-k & \text{if } s \geq k \\ 0 & \text{if } s < k, \end{cases} \\ W(0,\tau) &= 0, \\ W(S,\tau) &= \frac{W_{N-1} - W_{N-2}}{l}\left(S - (N-1)l\right) + W_{N-1} \\ &= 2W_{N-1} - W_{N-2}. \end{aligned} \tag{3}$$

Linear interpolation is applied to $W(S,\tau)$. Let the RHS of (2) be $f_{i,j} = f(W_{i,j})$. Note that $f_{i,j}$ contains the term $W_{i+1,j}$ and $W_{i-1,j}$. This can be solved by the Euler's method:

$$\begin{aligned} W_{i,0} &= 0, \\ W_{i,j+1} &= W_{i,j} + hf_{i,j}, \qquad \text{for } i = 1, 2, \ldots, N-1. \end{aligned}$$

Figure 1 is the solution of midpoint formula produced by Python. Consider the plot between $s = 35$ and $s = 65$, which interval contains the conspicuously changing slope. It shows the stock price with different remaining time.

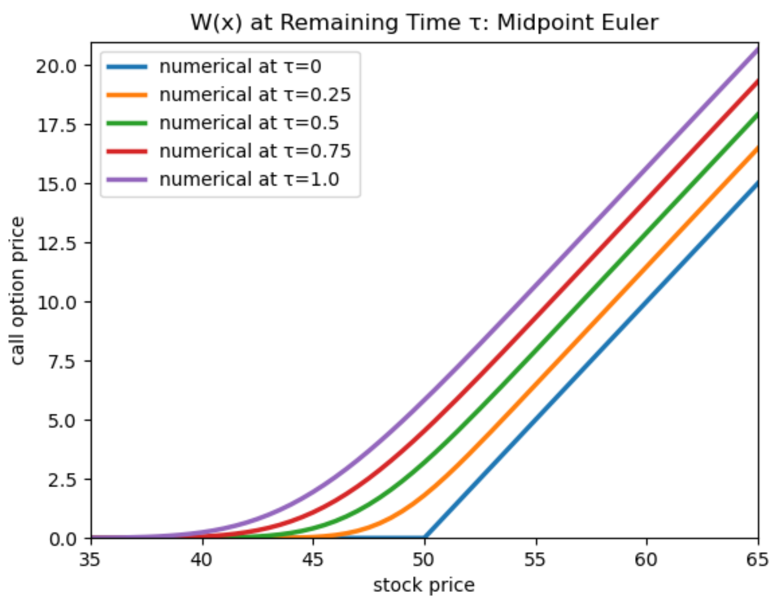



Figure 1: numerical solution of midpoint formula

## 3.2 Cubic Spline

Now introduce a new method to derive the approximation form: using cubic spline. Interpolating given data set $V(s_i, \tau_j)$ for fixed $j$. Let $P_{i,j}(s)$ is a cubic polynomial on the subinterval $[s_i, s_{i+1}]$ for each $i = 0, 1, \ldots, N-1$ and given $j$. Suppose $P'_{0,j}(s_0) = 0$, $P'_{N-1,j}(s_N) = 1$, and $P''_{0,j}(s_0) = 0$, $P''_{N-1,j}(s_N) = 0$, so that it has natural boundary. For the uniform grid, $P_i(s)$ is given as

$$\begin{aligned} P_{i,j}(s) &= a_{i,j} + b_{i,j}(s - s_i) + c_{i,j}(s - s_i)^2 + d_{i,j}(s - s_i)^3, \\ a_{i,j} &= V(s_i, \tau_j), \\ b_{i,j} &= \frac{1}{l}(a_{i+1,j} - a_{i,j}) - \frac{l}{3}(2c_{i,j} + c_{i+1,j}), \\ d_{i,j} &= \frac{1}{3l}(c_{i+1,j} - c_{i,j}), \\ lc_{i-1,j} + 4lc_{i,j} + lc_{i+1,j} &= \frac{3}{l}(a_{i+1,j} - 2a_{i,j} + a_{i-1,j}) \end{aligned}$$

by the definition of spline. Under natural boundary, above linear system equations of $c_i$ produce vector equation $\mathbf{A}\vec{\mathbf{c}} = \vec{\mathbf{z}}$,

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 & \cdots & \cdots & 0 \\ l & 4l & l & \ddots & & \vdots \\ 0 & l & 4l & l & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & l & 4l & l \\ 0 & \cdots & \cdots & 0 & 0 & 1 \end{bmatrix}_{(N+1)\times(N+1)}$$

$$\vec{\mathbf{z}} = \frac{3}{l}\begin{bmatrix} 0 \\ a_{2,j} - 2a_{1,j} + a_{0,j} \\ a_3 - 2a_{2,j} + a_{1,j} \\ \vdots \\ a_{N,j} - 2a_{N-1,j} + a_{N-2,j} \\ 0 \end{bmatrix} \quad \text{and} \quad \vec{\mathbf{c}} = \begin{bmatrix} c_{0,j} \\ c_{1,j} \\ c_{2,j} \\ \vdots \\ c_{N-1,j} \\ c_{N,j} \end{bmatrix}.$$

Since $\mathbf{A}$ is strictly diagonally dominant, $\det \mathbf{A} \neq 0$. Therefore, $\vec{\mathbf{c}}$ is determined uniquely, also each $P_{i,j}$ is produced. Since it is approximation of $V(s, \tau_j)$ on $[s_i, s_{i+1}]$, $\frac{\partial V}{\partial s} \simeq \frac{dP_i}{ds}$ and $\frac{\partial^2 V}{\partial s^2} \simeq \frac{d^2 P_i}{ds^2}$ on $[s_i, s_{i+1}]$. Plugging that into PDE$^{(1)}$ gives ODE with respected to $\tau$:

$$\frac{\partial V}{\partial \tau}(s_i, \tau_{j+1}) \simeq \frac{\partial W}{\partial \tau}(s_i, \tau_{j+1}) = rs_i(b_{i,j}) + \frac{1}{2}\sigma^2 s_i^2(2c_{i,j}) - rW_{i,j},$$

with the same initial and boundary conditions$^{(3)}$. Then each $W_{i,j}$ can be obatined by applying the same algorithm in the midpoint formula. Here is the result.

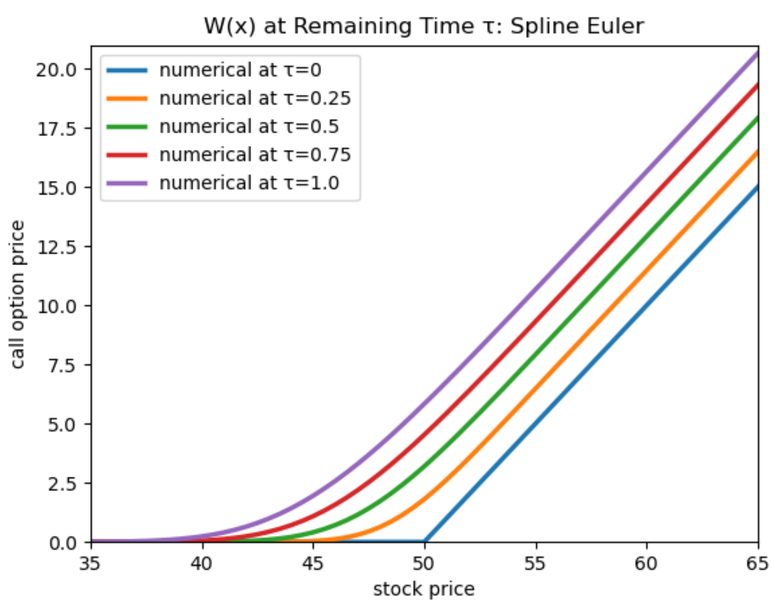
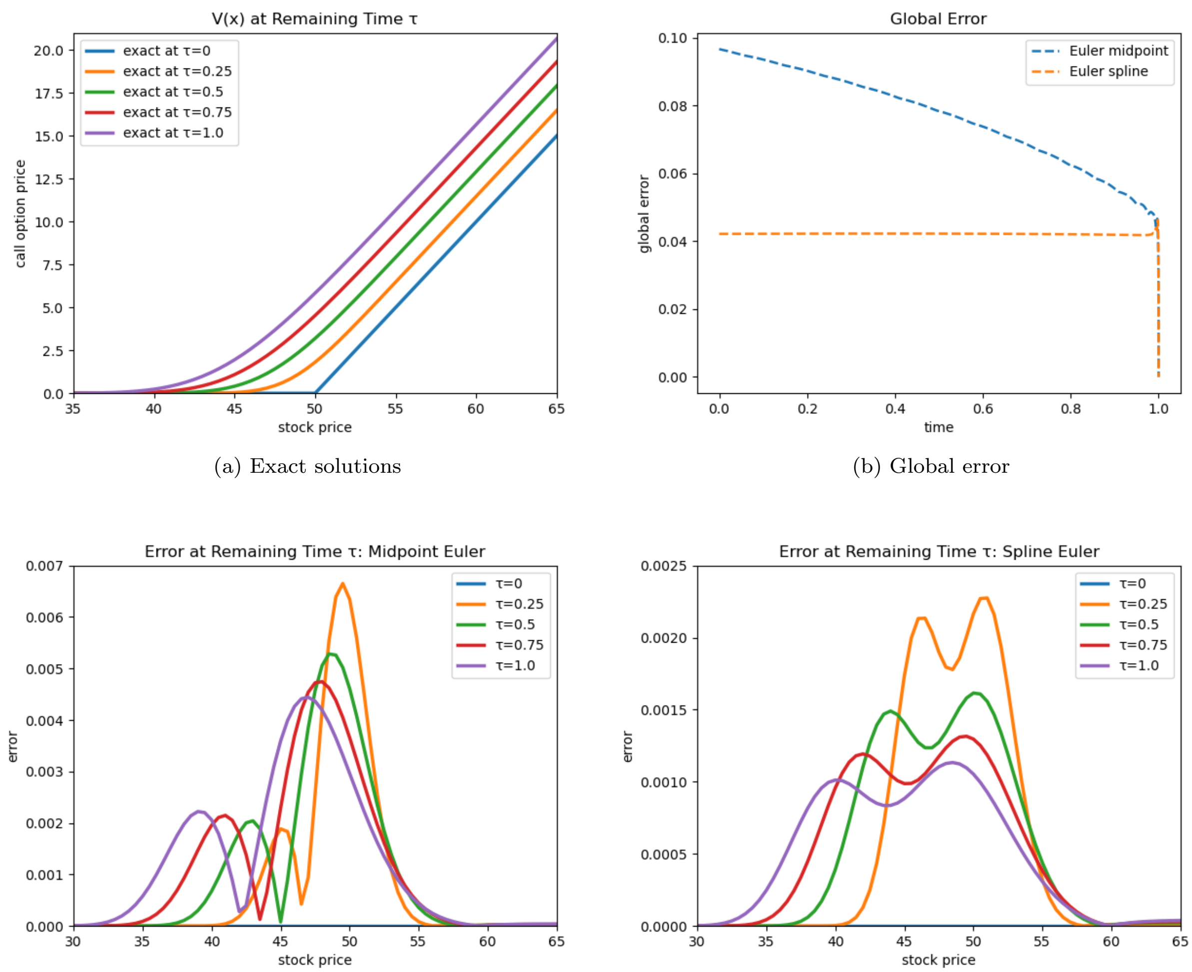



Figure 2: numerical solution of spline

# 4 Comparision

This section compares the forms of numerical and exact solutions, and find the absolute error $|V_{i,j} - W_{i,j}|$ at each $s_i$ and $\tau_j$. The exact solution is also plotted by Python. Above numerical solutions are similar to Figure 3 (a). Now compute the absolute error for each method. First is global error. Next, compare them with $s$-dimensional error. Note the second cases have the slope-changing interval. Each has the same color line at the same remaining time except the global error.




(a) Exact solutions

(b) Global error

(c) Midpoint error

(d) Spline error

Figure 3: Exact and errors

# 5 Conclusion

This article analyzed the Black-Scholes model in analytic and numerical ways. Just Euler method is used for $t$ coordinate, so applying another ODE solving skills such as Runge-Kutta good for next project. From the Comparision section, it seems that spline method is more exact than midpoint formula. However, it should solve the $N+1$ system of equations for just one fixed time, so has exorbitant time complexity if $M$ is large. In addition, this method looks like very unstable. The stability and convergence conditions are not found in here, but in test, spline is more unstable than midpoint. Presumably, a huge $t$-step is required to slightly increase the $s$-step. Also, spline and interpolation does not guarantee that they converge to exact function. The global error does not decrease as $t \to T$. Hence, it seems that spline method is not suitable for large $N$ and $M$. However, It has been shown that appropriate polynomials can have good approximations. So further research needs to focus on finding proper polynomials such as orthogonal or Laguerre, and checking stable conditions of each method.

# Reference


1. Fisher, R. (2021). Comparing Numerical Solution Methods for the Cahn-Hilliard Equation. doi:10.1137/20S1357974.

2. Jiawei He. (2021). A Study on Analytical and Numerical Solutions of Three Types of Black-Scholes Models. *International Journal of Trade, Economics and Finance*(vol.12, 4). 109-114. doi:10.18178/ijtef.2021.12.4.703

3. Nurul Anwar, M. and Sazzad Andallah, L. (2018). A Study on Numerical Solution of Black-Scholes Model. *Journal of Mathematical Finance, 8*, 372-381. doi:10.4236/jmf.2018.82024.

4. Richard L. Burden, J. Douglas Faires, and Annette M. Burden. (2015). *Numerical Analysis*(10th ed.). Boston: Cengage Learning.

5. Stein, Elias M. and Shakarchi, Rami. (2003). *Fourier Analysis: an introduction.* NJ: Princeton University Press.