

A NOTE ON NUMERICAL SOLUTION OF A LINEAR BLACK-SCHOLES MODEL

Md. Kazi Salah Uddin^{1*}, Mostak Ahmed² and Samir Kumar Bhowmik^{1,3}

¹Department of Mathematics, University of Dhaka, Dhaka-1000, Bangladesh

²Department of Mathematics, Jagannath University, Dhaka-1100, Bangladesh

³Department of Mathematics and Statistics, College of Science, Al-Imam Muhammad Ibn Saud
Islamic University, Riyadh, Kingdom of Saudi Arabia

Email: bhowmiksk@gmail.com

Received 18.08.2013

Accepted 02.11.2013

ABSTRACT

Black-Scholes equation is a well known partial differential equation in financial mathematics. In this article we discuss about some solution methods for the Black Scholes model with the European options (Call and Put) analytically as well as numerically. We study a weighted average method using different weights for numerical approximations. In fact, we approximate the model using a finite difference scheme in space first followed by a weighted average scheme for the time integration. Then we present the numerical results for the European Call and Put options. Finally, we investigate some linear algebra solvers to compare the superiority of the solvers.

Keywords: Black-Scholes model; call and put options; exact solution; finite difference schemes.

1. Introduction

The Black-Scholes model [5, 12, 15] is a powerful tool for valuation of equity options. This model is used for finding prices of stocks. Analytical approach and Numerical techniques are two ways for solving the European options. In [14] Mellin transformation was used to solve this model. They required neither variable transformation nor solving diffusion equation. R. Company, A.L. Gonzalez, L. Jodar [16] solved the modified Black-Scholes equation pricing option with discrete dividend. They used a delta-defining sequence of generalized Dirac-Delta function and applied the Mellin transformation to obtain an integral formula. Finally, they approximated the solution by using numerical quadrature approximation.

R. Company, L. Jodar, G. Rubio, R.J. Villanueva [14] found the solution of the model with a wide class of payoff functions that contained not only the Dirac delta type functions but also the ordinary payoff functions with discontinuities of their derivatives. Julia Ankudiova, Matthias Ehrhardt [13] solved non linear Black-Scholes equations numerically. They focused on various models relevant with the Black-Scholes equations with volatility depending on several factors. They also worked on the European Call option and American Call option analytically using transformation into a convection-

* The first author was partially funded by UGC grant for 2011-2012 session, Dhaka, Bangladesh.

diffusion equation with nonlinear term and the free boundary problem respectively. Numerical solutions of similar linear and non-linear PDEs, PIDEs can be found in [1, 3, 4, 6, 7, 9, 10].

In this paper, we discuss analytical solutions of the Black-Scholes equation using Fourier Transformation method for European options. We discuss a finite difference scheme to approximate the solutions. This article is organized in the following way. In Section 2, we discuss the model problem, then we review the analytic solutions in Section 3. We present the weighted average method to approximate the model in Section 4. We finish this study discussing and applying some of linear system solvers, numerical results, as well as some future research directions in Section 5.

2. The Model Problem

The well-known linear Black-Scholes equation (LBS) [12, 15] has been developed by Fischer Black and Myron Scholes in 1973 is

$$V_t + rSV_S + \frac{1}{2}\sigma^2S^2V_{SS} - rV = 0, \quad (1)$$

where $V = V(S, t)$, the pay-off function, $S = S(t)$, the stock price with $S = S(t) \geq 0$, $t =$ time with $t \in (0; T)$ for T is the time of maturity, $r =$ Risk-Free interest rate, $\sigma =$ Volatility Condition. We need to specify the condition so that we can able to find a unique solution of the equation (1). Now we state the terminal and boundary conditions [11] for both the European Call and Put option. We know that the American options can be exercised at any time before expiry; we can find the optimal time t of exercise for which the exercise is optimal. That is the asset price at this time reaches optimal exercise boundary $S_f(t)$.

European Call Option

The solution to the Black-Scholes equation (1) is the value $V(S, t)$ of the European Call option [11] on $0 \leq S < \infty$; $0 \leq t \leq T$. The boundary and terminal conditions are as follows

$$\begin{aligned} V(0, t) &= 0 && \text{for } 0 \leq t \leq T, \\ V(S, t) &\sim S - Ke^{-r(T-t)} && \text{as } S \rightarrow \infty, \\ V(S, T) &= (S - K)^+ && \text{for } 0 \leq S < \infty, \end{aligned} \quad (2)$$

European Put Option

European Put option [11] is the reciprocal of the European Call option and the boundary and terminal conditions are

$$\begin{aligned} V(0, t) &= Ke^{r(T-t)} && \text{for } 0 \leq t \leq T, \\ V(S, t) &\sim 0 && \text{as } S \rightarrow \infty, \\ V(S, T) &= (K - S)^+ && \text{for } 0 \leq S < \infty, \end{aligned} \quad (3)$$

3. Exact Solution of the LBS Model

Let $S = Ke^x$, $t = T - \frac{r}{\sigma^2/2}$, and $v(x, \tau) = \frac{1}{K}V(S, t)$. Now

$$\begin{aligned}\frac{\partial V}{\partial t} &= \frac{\partial V}{\partial \tau} \frac{\partial \tau}{\partial t} + \frac{\partial V}{\partial S} \frac{\partial S}{\partial t} = -\frac{\sigma^2}{2} K \frac{\partial v}{\partial \tau}, \\ \frac{\partial V}{\partial S} &= K \frac{\partial v}{\partial S} = \frac{K}{S} \frac{\partial v}{\partial x},\end{aligned}$$

and

$$\frac{\partial^2 V}{\partial S^2} = K \frac{\partial^2 v}{\partial S^2} = \frac{K}{S^2} \left[\frac{\partial^2 v}{\partial x^2} - \frac{\partial v}{\partial x} \right],$$

Substituting these derivatives in (1) one gets

$$-\frac{\sigma^2}{2} K \frac{\partial v}{\partial \tau} + \frac{\sigma^2}{2} K \left[\frac{\partial^2 v}{\partial x^2} - \frac{\partial v}{\partial x} \right] + rK \frac{\partial v}{\partial x} - rKv = 0,$$

which implies

$$\frac{\partial v}{\partial \tau} = \frac{\partial^2 v}{\partial x^2} + \left(\frac{2r}{\sigma^2} - 1 \right) \frac{\partial v}{\partial x} - \frac{2r}{\sigma^2} v \quad (4)$$

Letting $\frac{2r}{\sigma^2} = \theta$ (4) can be written as

$$\frac{\partial v}{\partial \tau} = \frac{\partial^2 v}{\partial x^2} + (\theta - 1) \frac{\partial v}{\partial x} - \theta v \quad (5)$$

Now consider $\lambda = \frac{1}{2}(\theta - 1)$, $\nu = \frac{1}{2}(\theta + 1) = \lambda + 1$, to get $\nu^2 = \lambda^2 + \theta$, and then $v(x, \tau) = e^{-\lambda x - \nu^2 \tau} u(x, \tau)$. Therefore

$$\begin{aligned}\frac{\partial v}{\partial \tau} &= e^{-\lambda x - \nu^2 \tau} (-\nu^2) u(x, \tau) + \frac{\partial u}{\partial \tau} e^{-\lambda x - \nu^2 \tau}, \\ &= e^{-\lambda x - \nu^2 \tau} \left[-\nu^2 u + \frac{\partial u}{\partial \tau} \right],\end{aligned}$$

$$\frac{\partial v}{\partial x} = e^{-\lambda x - \nu^2 \tau} \left[-\lambda u + \frac{\partial u}{\partial x} \right],$$

$$\frac{\partial v}{\partial x} = e^{-\lambda x - \nu^2 \tau} \left[\lambda^2 u - 2\lambda \frac{\partial u}{\partial x} + \frac{\partial^2 u}{\partial x^2} \right],$$

Inserting these into equation (5) and dividing by $e^{-\lambda x - \nu^2 \tau}$ we get

$$-\nu^2 u + \frac{\partial u}{\partial \tau} = \left[\lambda^2 u - 2\lambda \frac{\partial u}{\partial x} + \frac{\partial^2 u}{\partial x^2} \right] + (\theta - 1) \left[-\lambda u + \frac{\partial u}{\partial x} \right] - \theta u,$$

implies

$$u_\tau = u_{xx} + (-2\lambda + \theta - 1)u_x + (\lambda^2 + \nu^2 - \lambda(\theta - 1))u,$$

and finally we get,

$$u_\tau = u_{xx}. \quad (6)$$

The initial & boundary conditions for the European Call and Put options are respectively

$$\begin{aligned} u(x, 0) &= (e^{(\lambda+1)x} - e^{\lambda x})^+ \quad \text{as } x \in \mathbb{R}, \\ u(x, \tau) &= 0 \quad \text{as } x \rightarrow -\infty, \\ u(x, \tau) &= e^{(\lambda+1)x + \nu^2 \tau} - e^{\lambda x + \lambda^2 \tau} \quad \text{as } x \rightarrow \infty, \end{aligned} \quad (7)$$

and

$$\begin{aligned} u(x, 0) &= (e^{\lambda x} - e^{(\lambda+1)x})^+ \quad \text{as } x \in \mathbb{R}, \\ u(x, \tau) &= e^{\lambda x + \lambda^2 \tau} \quad \text{as } x \rightarrow -\infty, \\ u(x, \tau) &= 0 \quad \text{as } x \rightarrow \infty, \end{aligned} \quad (8)$$

Thus the Black-Scholes equation reduces to the heat equation. Now we have the problem (6) with initial condition $u(x, 0) = g(x) = \max\{(e^{(\lambda+1)x} - e^{\lambda x}), 0\}$, for Call option, and $u(x, 0) = g(x) = \max\{(e^x - e^{(\lambda+1)x}), 0\}$, for Put option.

Applying the Fourier transformation with respect to x in equation (6) we have

$$\frac{\partial}{\partial \tau} \hat{u}(\xi, \tau) = (i\xi)^2 \hat{u}(\xi, \tau),$$

where

$$\hat{u}(\xi, \tau) = \int_{-\infty}^{\infty} u(x, \tau) e^{2\pi i x \xi} dx.$$

Thus the solution in the Fourier domain can be presented as

$$\hat{u} = \hat{u}(\xi, 0) e^{-\xi^2 \tau}.$$

Now the initial function is $u(x, 0) = g(x)$. Hence applying the inverse Fourier transformation

we write the exact solution as

$$u(x, \tau) = \frac{1}{\sqrt{4\pi\tau}} \int_{-\infty}^{\infty} g(\xi) e^{-\frac{(x-\xi)^2}{4\tau}} d\xi$$

Call Option

$$\begin{aligned} u(x, \tau) &= \frac{1}{\sqrt{4\pi\tau}} \int_{-\infty}^{\infty} \max\{e^{(\lambda+1)\xi} - e^{\lambda\xi}, 0\} e^{-\frac{(x-\xi)^2}{4\tau}} d\xi \\ &= \frac{1}{\sqrt{4\pi\tau}} \int_0^{\infty} (e^{(\lambda+1)\xi} - e^{\lambda\xi}) e^{-\frac{(x-\xi)^2}{4\tau}} d\xi \end{aligned}$$

$$\begin{aligned}
&= \frac{1}{\sqrt{4\pi\tau}} \int_0^\infty e^{(\lambda+1)\xi} e^{-\frac{(x-\xi)^2}{4\tau}} d\xi - \frac{1}{\sqrt{4\pi\tau}} \int_0^\infty e^{\lambda\xi} e^{-\frac{(x-\xi)^2}{4\tau}} d\xi \\
&= I_{\lambda+1} - I_\lambda.
\end{aligned}$$

where

$$I_\mu = \frac{1}{\sqrt{4\pi\tau}} \int_0^\infty e^{\mu\xi} e^{-\frac{(x-\xi)^2}{4\tau}} d\xi.$$

Let

$$\eta = \frac{x - \xi + 2\mu\tau}{\sqrt{2\tau}},$$

implies,

$$d\eta = -\frac{d\xi}{\sqrt{2\tau}}.$$

Hence

$$\begin{aligned}
I_\mu &= -\frac{1}{\sqrt{4\pi\tau}} e^{\mu x + \mu^2 \tau} \int_{\frac{x+2\mu\tau}{\sqrt{2\tau}}}^{-\infty} e^{-\frac{\eta^2}{2}} \sqrt{2\tau} d\eta \\
&= e^{\mu x + \mu^2 \tau} \left[\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\frac{x+2\mu\tau}{\sqrt{2\tau}}} e^{-\frac{\eta^2}{2}} d\eta \right] \\
&= e^{\mu x + \mu^2 \tau} \Phi\left(\frac{x + 2\mu\tau}{\sqrt{2\tau}}\right).
\end{aligned}$$

Therefore,

$$I_\mu = e^{(\lambda+1)x + (\lambda+1)^2 \tau} \Phi\left(\frac{x + 2\mu\tau}{\sqrt{2\tau}}\right) - e^{\lambda x + \lambda^2 \tau} \Phi\left(\frac{x + 2\mu\tau}{\sqrt{2\tau}}\right)$$

Introducing new variables d_1 ; d_2 and converting the variables back into the original Black-Scholes variables, we have

$$d_1 = \left(\frac{x+2\mu\tau}{\sqrt{2\tau}}\right) \text{ and } d_2 = \left(\frac{x+2\mu\tau}{\sqrt{2\tau}}\right)$$

where

$$\begin{aligned}
S &= Ke^x, \text{ or, } x = \ln\left(\frac{S}{K}\right), \\
t &= T - \frac{r}{\frac{\sigma^2}{2}}, \text{ or, } r = \frac{\sigma^2}{2}(T - t), \\
\lambda &= \frac{1}{2}(\theta - 1), \text{ or, } \lambda = \left(\frac{r}{\sigma^2} - \frac{1}{2}\right), \\
\nu &= \frac{1}{2}(\theta + 1), \text{ or, } \nu = \left(\frac{r}{\sigma^2} + \frac{1}{2}\right),
\end{aligned}$$

Therefore,

$$\begin{aligned}
 d_1 &= \frac{\ln\left(\frac{S}{K}\right) + 2(T-t)\left(\frac{\sigma^2}{2}\right)\left(\frac{r}{\sigma^2} + \frac{1}{2}\right)}{\sqrt{2(T-t)\left(\frac{\sigma^2}{2}\right)}} \\
 &= \frac{\ln\left(\frac{S}{K}\right) + 2(T-t)\left(r + \frac{\sigma^2}{2}\right)}{\sigma\sqrt{(T-t)}} \\
 d_2 &= \frac{\ln\left(\frac{S}{K}\right) + 2(T-t)\left(\frac{\sigma^2}{2}\right)\left(\frac{r}{\sigma^2} - \frac{1}{2}\right)}{\sqrt{2(T-t)\left(\frac{\sigma^2}{2}\right)}} \\
 &= \frac{\ln\left(\frac{S}{K}\right) + 2(T-t)\left(r - \frac{\sigma^2}{2}\right)}{\sigma\sqrt{(T-t)}}
 \end{aligned}$$

And finally we have

$$\begin{aligned}
 V(S, t) &= Kv(x, \tau) \\
 &= Ke^{-\lambda x - v^2 \tau} u(x, \tau) \\
 &= Ke^{\lambda x - v^2 \tau} e^{vx + v^2 \tau} \Phi(d_1) - Ke^{\lambda x - v^2 \tau} e^{\lambda x + \lambda^2 \tau} \Phi(d_2) \\
 &= Ke^{x(v-\lambda)} \Phi(d_1) - Ke^{v^2 \tau + \lambda^2 \tau} \Phi(d_2) \\
 &= Ke^x \Phi(d_1) - Ke^{\tau\left(-\frac{2r}{\sigma^2}\right)} \Phi(d_2) \\
 &= Ke^x \Phi(d_1) - Ke^{-r(T-t)} \Phi(d_2) \\
 &= S\Phi(d_1) - Ke^{-r(T-t)} \Phi(d_2).
 \end{aligned}$$

Therefore

$$V_c(S, t) = S\Phi(d_1) - Ke^{-r(T-t)}\Phi(d_2).$$

This gives the values for the European Call option where $\Phi(d)$ the standard normal distribution of the function is d . We show $V(S, t)$ in Figure 1 corresponding parameters $K = 100, \sigma = .2, r = .1, T = 1 \text{ year}, dt = .001$.

Put Option

A similar calculation for the European Put option and the pay-off function for the Put option in absence of dividend is

$$V_p(S, t) = Ke^{-r(T-t)}\Phi(-d_2) - S\Phi(-d_1).$$

Figure 1 corresponding parameters $K = 100, \sigma = .2, r = .1, T = 1 \text{ year}, dt = .001$.

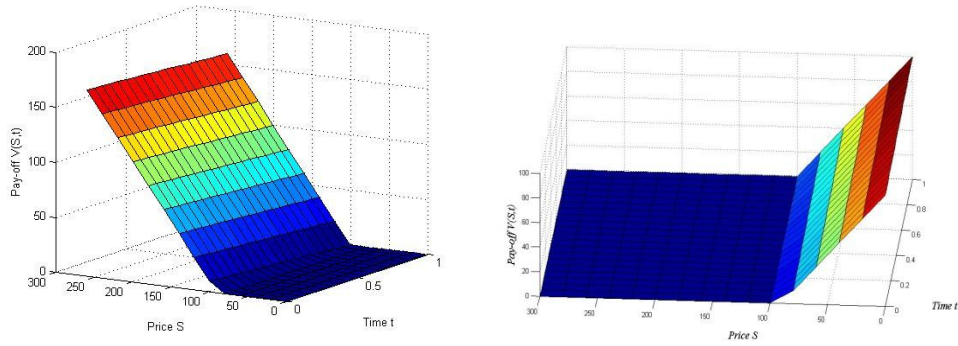


Figure 1: Analytic Solution of European Call option (left figure), the European Put option (right figure).

4. Numerical Approximations to the Transformed LBS Model

In this section we motivate ourselves to solve the model numerically. We use a finite difference scheme to solve the transform problem; the heat equation. Finally we use the back substitution of the coordinate transformation to get the solution of the problem related to the partial differential equation (1).

Finite Difference Formulation

We have the model

$$u_\tau = u_{xx}, \quad x \in \mathbb{R}, \quad 0 \leq \tau \leq \hat{T} \quad (9)$$

with the initial and boundary conditions for call option as

$$\begin{aligned} u(x, 0) &= (e^{(\lambda+1)x} - e^{\lambda x})^+ \quad \text{as } x \in \mathbb{R}, \\ u(x, \tau) &= 0 \quad \text{as } x \rightarrow -\infty, \\ u(x, \tau) &= e^{(\lambda+1)x + v^2 \tau} - e^{\lambda x + \lambda^2 \tau} \quad \text{as } x \rightarrow \infty, \end{aligned} \quad (10)$$

To approximate the model numerically we truncate the infinite space domain to a finite domain, and impose relevant boundary conditions that fit the model. Here we consider $x \in [-1, 1]$. Applying the difference quotients for spatial derivatives only we obtain the following relation [2]

$$\frac{\partial U_i(\tau)}{\partial \tau} = \frac{U_{i-1} - 2U_i + U_{i+1}}{h^2}, \quad (11)$$

where

$$i = -(N-1), -(N-2), \dots, 0, \dots, (N-1),$$

and

$$\begin{cases} U(x_i, \tau) = U_{-N} = 0 \quad \text{as } x \rightarrow -1. \\ U(x_i, \tau) = U_N = e^{(\lambda+1)x + v^2 \tau} - e^{\lambda x + \lambda^2 \tau} \quad \text{as } x \rightarrow 1. \end{cases} \quad (12)$$

Here we have used centered difference scheme. Applying boundary conditions equation (11) gives the following system

$$\frac{dM(\tau)}{d\tau} = AM + C, \quad (13)$$

where

M and C are the column vectors of order $2N - 1$ and A is a matrix of order $(2N - 1) \times (2N - 1)$ and

$$\begin{aligned} M &= [U_{-N+1}, U_{-N+2}, \dots, \dots, U_{N-2}, U_{N-1}]^T, \\ C &= \frac{1}{h^2} [U_{-N}, 0, 0, \dots, \dots, 0, 0, U_N]^T, \\ A &= \frac{1}{h^2} \begin{pmatrix} -2 & 1 & 0 & \dots & 0 \\ 1 & -2 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & 1 & -2 \end{pmatrix}. \end{aligned}$$

Thus we have a system of odes of the form

$$M_\tau = AM + C, \quad (14)$$

with a suitable initial condition. Now we would like to discrete the equation (14) with respect to time. One may start with a simple scheme. One of the trivial choices is to use the forward Euler scheme. Firstly, we discretize (14) explicitly and we have

$$M^{n+1} = (I + \Delta\tau A)M^n + \Delta\tau C. \quad (15)$$

The difficulty of using the scheme is that it needs very little step size to converge, as a result the scheme is a slow one, and is not of interest in this advance study. Besides the scheme is conditionally stable with the Courant-Friedrichs-Lewy (CFL) condition $\left|\frac{\Delta\tau}{h^2}\right| < 1$ and the order of the accuracy of the scheme is $O(k) + O(h^2)$.

We want a fast and efficient scheme, so we want larger time stepping, and interested in using implicit techniques. We discretize (14) implicitly and have

$$\frac{M^{n+1} - M^n}{\Delta\tau} = AM^{n+1} + C,$$

gives

$$(I - \Delta\tau A)M^{n+1} = M^n + \Delta\tau C, \quad (16)$$

which is a system of linear equations with unknowns M^{n+1} . The advantage of using (16) is that the scheme is unconditionally stable [5]. It is accurate of order $O(k) + O(h^2)$. It is faster than the explicit Euler scheme since (16) allows us to use large time steps.

The weighted average method

If we introduce an weight δ and with the discretization of (14) we have

$$\begin{aligned} \frac{M^{n+1} - M^n}{\Delta\tau} &= A(\delta M^{n+1} + (1 - \delta)M^n) + C, \\ M^{n+1} - M^n &= \Delta\tau A(\delta M^{n+1} + (1 - \delta)M^n) + \Delta\tau C, \\ \text{or, } (I - \Delta\tau\delta A)M^{n+1} &= (I + (1 - \delta)\Delta\tau A)M^n + \Delta\tau C. \end{aligned} \quad (17)$$

This system is also a linear one with unknowns M^{n+1} , where δ varies from 0 to 1. This method turns to the explicit method when $\delta = 0$ i.e., equations (15) & (17) are same and implicit method when $\delta = 1$ i.e., equations (16) & (17) are same. For $0 \leq \delta \leq \frac{1}{2}$, the scheme is stable if and only if $\left| \frac{\Delta \tau}{h^2} \right| \leq \frac{1}{2}(1 - 2\delta)^{-1}$ and unconditionally stable for $\frac{1}{2} \leq \delta \leq 1$. The order of the accuracy of the scheme is $O(k) + O(h^2)$. In Figure 2 we present the approximate solutions. It is easy to notice that they agree with the exact solutions shown in Figure 1. Here for time investigation various other one step and multistep schemes can be used. The implementations are similar. So all linear algebra tools we apply in the next section can also be applied to the other schemes as well. For time constraint we only investigate (17).

5. Numerical Results and Discussions

The system of linear equations (17) generated by the discretization of the Black-Scholes model can be solved by many conventional processes. For a large scale linear system, scientists rarely use direct methods as they are computationally costly.

Here, in this section, it is our motivation to solve the system of equation (17) using various iterative techniques. Here we first investigate which linear solver converges swiftly. To that end, we consider Jacobi iterative method, Gauss-Seidel iterative method and successive over relaxation method to start with. In terms of matrices, the Jacobi method can be expressed as

$$x^{(k)} = D^{-1}(L + U)x^{(k-1)} + D^{-1}b,$$

The the Gauss-Seidel method

$$x^{(k)} = (D - L)^{-1}(Ux^{(k-1)} + b),$$

and the Successive Over Relaxation (SOR) algorithm can be written as

$$x^{(k)} = (D - \omega L)^{-1}[\omega U + (1 - \omega)D]x^{(k-1)} + \omega(D - L)^{-1}b,$$

where in each case the matrices D , $-L$, and $-U$ represent the diagonal, strictly lower triangular, and strictly upper triangular parts of A , respectively. To optimize the convergence of the SOR method we fix the weight (ω) by using the following formula [8]:

$$\omega = \frac{2}{1 + \sqrt{1 - [\rho(T_j)]^2}}$$

Where $T_j = D^{-1}(L + U)$.

Then we investigate Preconditioned Conjugate Gradient (PCG) Method, Generalized Minimal Residual (GMRES) Method and BiConjugate Gradient Stabilized (BICGSTAB) Method with a diagonal preconditioning [5].

Here for all computations we consider $K = 100$, $\sigma = .2$, $r = .1$, $T = 1$ year, $dt = 0.001$. The results are presented in Table 1 & Table 2 with different weights δ . Observing Table 1 & Table 2, we notice that GMRES with $\delta = \frac{1}{4}$ performs the best

among all others iterative solvers in terms of number of iterations as well as CPU time taken. Thus, analyzing Table 1 & Table 2, it is easy to see that a right choice of weight (δ) plays an important role while solving the full discrete model.

This study has been motivated only for the one dimensional linear Black-Scholes model. We investigate a finite difference scheme only. A study on a non-linear Black-Scholes model with higher order accurate schemes as well as multi-dimensional models are more interesting and challenging, which left as future research interests.

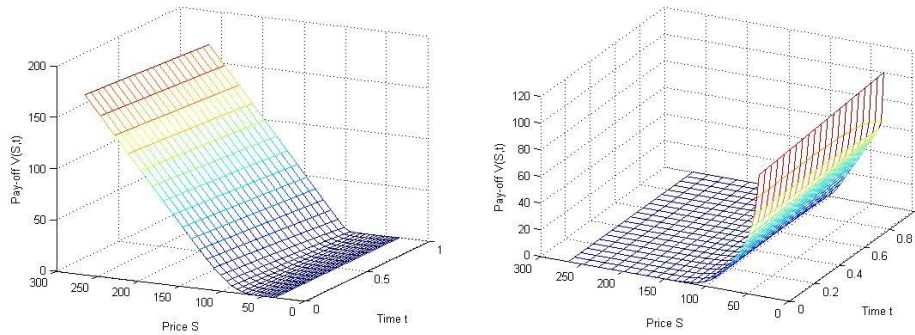


Figure 2: Numerical solution of Call option (left figure), Put option (right figure)

Table 1: Iteration counts for the numerical solution of (17) (in one step) with tolerance 10^{-6} for different weights and different methods. Here we consider the parameters $K = 100, \delta = .2, r = .1, T = 1 \text{ year}, dt = 0.001$.

h	Weight	No. of iterations					
		Jacobi	Gauss-Seidel	Gauss-Seidel SOR	PCG	GMRES	Bi-CGSTAB
1/128	$\delta = 1/4$	105	61	28	29	24	22
	$\delta = 1/2$	187	105	40	41	32	31
	$\delta = 3/4$	264	146	49	50	37	38
	$\delta = 1$	337	185	56	57	41	41
1/256	$\delta = 1/4$	372	203	57	56	40	41
	$\delta = 1/2$	670	363	80	78	51	54
	$\delta = 3/4$	945	511	98	95	58	64
	$\delta = 1$	1204	651	114	109	63	68
1/512	$\delta = 1/4$	1343	718	114	108	62	73
	$\delta = 1/2$	2410	1290	161	149	75	102
	$\delta = 3/4$	3381	1815	197	181	83	123
	$\delta = 1$	4291	2309	228	206	88	135
1/1024	$\delta = 1/4$	4845	2570	228	205	86	120
	$\delta = 1/2$	8604	4594	323	284	97	182
	$\delta = 3/4$	11978	6431	396	342	100	212
	$\delta = 1$	15109	8149	457	391	100	255

Table 2: CPU (Processor: Intel(R) Atom(TM) CPU N455 @1.66GHz, RAM: 1GB DDR3, Hard Disk: SATA 250GB) program execution time in seconds to generate the data shown in Table 1.

h	Weight	No. of iterations					
		Jacobi	Gauss-Seidel	Gauss-Seidel SOR	PCG	GMRES	Bi-CGSTAB
1/128	$\delta = 1/4$	0.1858	0.1359	0.3101	1.0084	0.4592	0.2198
	$\delta = 1/2$	0.2001	0.1430	0.2723	0.1378	0.2082	0.2103
	$\delta = 3/4$	0.2633	0.1781	0.2879	0.2319	0.2148	0.2514
	$\delta = 1$	0.3163	0.2074	0.3051	0.1901	0.2259	0.2858
1/256	$\delta = 1/4$	1.1103	0.7979	1.5475	0.6975	0.4918	0.5853
	$\delta = 1/2$	1.6953	1.0814	1.8525	0.5781	0.4859	0.7787
	$\delta = 3/4$	2.0608	1.4022	1.6970	0.7646	0.6741	0.9492
	$\delta = 1$	2.5304	1.5966	1.8407	0.7916	0.5164	1.0116
1/512	$\delta = 1/4$	12.4278	8.6257	10.5701	2.7356	1.5892	4.0263
	$\delta = 1/2$	18.9933	13.0422	11.5689	3.6496	2.1284	5.0929
	$\delta = 3/4$	25.4323	16.8498	12.4470	4.4150	2.3200	6.0725
	$\delta = 1$	31.2429	20.0590	13.2400	5.0081	2.1938	6.5567
1/1024	$\delta = 1/4$	148.7260	134.6243	79.3490	19.1500	7.0460	22.0985
	$\delta = 1/2$	247.5813	212.7182	88.3009	26.3526	7.9299	33.2071
	$\delta = 3/4$	330.9538	281.5428	94.1031	31.2387	8.1273	39.2230
	$\delta = 1$	410.7900	347.2522	101.8654	36.0884	8.1368	46.5401

REFERENCES

- [1] M. Bakker. One-dimensional Galerkin methods and super convergence at interior nodal points. *SIAM Journal on Numerical Analysis*, 21(1):101–110, Feb 1984.
- [2] S. K. Bhowmik. Stable numerical schemes for a partly convolutional partial integrodifferential equation. *Applied Mathematics and Computation*, 217(8):4217–4226, 2010.
- [3] S. K. Bhowmik. Numerical approximation of a convolution model of dot theta-neuron networks. *Applied Numerical Mathematics*, 61:581–592, 2011.
- [4] S. K. Bhowmik. Stability and convergence analysis of a one step approximation of a linear partial integro-differential equation. *Numerical Methods for Partial Differential Equation*, 27(5):11791200, September 2011.

- [5] S. K. Bhowmik. Fast and efficient numerical methods for an extended Black-Scholes model. arXiv:1205.6265, 2013.
- [6] S. K. Bhowmik, D. B. Duncan, M. Grinfeld, and G. J. Lord. Finite to infinite steady state solutions, bifurcations of an integro-differential equation. *Discrete and Continuous Dynamical Systems*, 16(1):57 – 71, 2011.
- [7] S. K. Bhowmik and C. C. Stolk. Preconditioners based on windowed fourier frames applied to elliptic partial differential equations. *Journal of Pseudo-Differential Operators and Applications*, 2(3):317–342, 2011.
- [8] R.L. Burden, J. Douglas Faires, *Numerical Analysis, Seventh Ed.*, Thomson Learning, 2001.
- [9] S. Dhawan, S. Kapoor, and S. Kumar. Numerical method for advection diffusion equation using fem and b-splines. *Journal of Computational Science*, 3(5):429437, 2012.
- [10] S. Dhawan, R. Kumar, and S. Kumar. Multiquadratic quasi interpolation for burgerfisher equation. *Int. J. of Appl. Math. and Mech.*, 9(11):41–50, 2013.
- [11] P.D.M. Ehrhardt and A. Unterreiter. The numerical solution of nonlinear BlackScholes equations. *Technische Universitat Berlin*, 28, 2008.
- [12] M. Scholes F. Black. The pricing of options and corporate liabilities. *J. Pol. Econ*, 81:637–659., 1973.
- [13] M. Ehrhardt J. Ankudinova*. On the numerical solution of nonlinear BlackScholes equations. *Computers and Mathematics with Applications*, 56:799–812, 2008.
- [14] J.C. Cortos* R. Sala L. Jodar, R Sevilla-Peris. A new direct method for solving the Black-Scholes equation. *Applied Mathematics Letters*, 18, 2005.
- [15] R.C. Merton. Theory of rational option pricing. *Bell J. Econ.*, 4(1):141–183, 1973.
- [16] L. Jodar R. Company*, A.L. Gonzalez. Numerical solution of modified BlackScholes equation pricing stock options with discrete dividend. *Mathematical and Computer Modelling*, 44, 2006.