Math. Commun. **30** (2025), 191–205
DOI: 10.64785/mc.30.2.4

Mathematical Communications

An efficient robust computational method for solving Black-Scholes PDEs

Saurabh Bansal and Srinivasan Natesan*

Department of Mathematics, Indian Institute of Technology Guwahati, Guwahati - 781 039, India

Abstract. In this article, we propose a computational method for the numerical solution of Black-Scholes PDEs arising in option pricing. First, we discretize the time-domain by uniform mesh and apply the Crank-Nicolson method to approximate the time variable. Then, we use the streamline-diffusion finite element method (SDFEM) for the spatial derivative on different nonuniform meshes. The proposed method is of second-order convergent in both variables. For comparison purposes, we use the backward-Euler scheme for the time derivative, which will be of first-order convergent. Numerical experiments are carried out to verify theoretical results.

AMS subject classifications: 65M12, 65M15, 65M50, 65M60
Keywords: option pricing; Black-Scholes equation; streamline-diffusion finite element method; butterfly option
Received August 14, 2023; accepted April 16, 2025

1. Introduction

An option is a financial instrument whose value is determined by the underlying asset value. The contract buyer obtains the right, but not the duty, to buy or sell an asset at a defined price on or before the contract maturity date. An American option may be exercised at any time prior to the maturity date, but a European option may only be executed on the date of maturity [13].

According to Black and Scholes [4], a parabolic partial differential equation (PDE) of second-order with respect to the underlying asset price and the expiry time determines the value of a European option. The famous Black-Scholes PDE is given by

$$\begin{cases} \dfrac{\partial \widehat{U}}{\partial t} + \dfrac{1}{2}\sigma^2 S^2 \dfrac{\partial^2 \widehat{U}}{\partial S^2} + rS\dfrac{\partial \widehat{U}}{\partial S} - r\widehat{U} = 0, \quad (S,t) \in \mathbb{R}_+ \times (0,T), \\ \widehat{U}(S,T) = \widehat{U}_T(S), \quad S \in \mathbb{R}_+, \end{cases}$$

where $\widehat{U}_T(S)$ is the payoff function of the option with T being the maturity time, r represents the risk-free interest rate and σ is the volatility function of the underlying asset.

In the literature, several finite difference and finite element methods are available to determine the European and American options. To cite a few, Chacur et al. [5] applied the finite element technique to deal with the real option pricing problem. Kadalbajoo et al. [7] presented a cubic B-spline collocation technique and it is second-order accurate in both spatial and temporal variables. Later Mohammadi [10] described a uniform quintic B-spline collocation method that is third-order and second-order accurate in space and time variables, respectively. Rao et al. [12] presented a numerical method using the HODIE (High-Order Difference approximation with Identity Expansion) scheme in the spatial variable that exhibits second-order convergence in both variables. Recently, Bansal and Natesan [2, 3] used the Richardson extrapolation method to solve a generalized Black-Scholes equation pricing European options.

Our aim in this article is to devise a robust computational method for the numerical solution of Black-Scholes PDEs arising in option pricing. In fact, the method consists of the Crank-Nicolson method for the time-derivative and streamline-diffusion finite element method (SDFEM) for solving Black-Scholes PDEs. We discretize the time-domain by uniform meshes and apply the Crank-Nicolson scheme for the

Email addresses: sbansal@iitg.ac.in (S. Bansal), natesan@iitg.ac.in (S. Natesan)
*Corresponding author.

time derivative. To discretize the spatial domain we use several nonuniform meshes, and the SDFEM for the spatial derivatives. This numerical schemes provides the second-order of convergence on both time and spatial variables.

The rest of the article is organized as follows. First of all, a Black-Scholes PDE with boundary conditions and a terminal condition is introduced in Section 2, and we also make some modifications so that the Black-Scholes PDE can be solved easily by the numerical scheme. Section 3 addresses different types of meshes. Section 4 deals with the numerical scheme. Section 5 is devoted to error analysis. In Section 6, numerical experiments are carried out. Finally, some conclusions are given in Section 7.

Notations: In this paper, $||.||$ denotes a standard L^2-norm over Ω and $\mathcal{C}$ denotes a generic positive constant.

2. Problem formulation

2.1. Model problem

In this article, we will deal with the following type Black-Scholes equation:

$$\begin{cases} \dfrac{\partial \widehat{U}}{\partial \tau} + \dfrac{1}{2}\sigma^2 S^2 \dfrac{\partial^2 \widehat{U}}{\partial S^2} + (r-\delta)S\dfrac{\partial \widehat{U}}{\partial S} - r\widehat{U} = 0, & (S,\tau) \in \mathbb{R}_+ \times [0,T)\,, \\ \widehat{U}(S,T) = \widehat{U}_T(S), & S \in \mathbb{R}_+, \\ \widehat{U}(S,\tau) = \widehat{U}_b(S,\tau), & (S,\tau) \in \partial\mathbb{R}_+ \times [0,T]\,, \end{cases} \tag{1}$$

where δ is the dividend of the underlying asset, $\widehat{U}_T(S)$ is the payoff function of our option, and $\widehat{U}_b(S,\tau)$ denotes the boundary conditions. The existence of a unique classical solution for model problem (1) is well known from [8, 9]. We will consider the final condition and boundary conditions for the following options [12].

2.1.1. Long butterfly spread option

Combining the long and short selling of European call options over the same underlying asset and with the same maturity date results in a long call butterfly spread option. There are two options with strike prices of K_1 and K_3 (long positions) and two options with strike prices of K_2 (short positions). The payoff function $\widehat{U}(S,T)$ for a long call butterfly spread option is given by

$$\widehat{U}(S,T) = \max(S-K_1,0) - 2\max(S-K_2,0) + \max(S-K_3,0), \quad S > 0,$$

while for the long put butterfly spread option it is given by

$$\widehat{U}(S,T) = \max(K_1-S,0) - 2\max(K_2-S,0) + \max(K_3-S,0), \quad S > 0,$$

and the following are the boundary conditions:

$$\begin{cases} \widehat{U}(0,\tau) = 0, & \tau \in [0,T]\,, \\ \lim\limits_{S\to\infty} \widehat{U}(S,\tau) = 0, & \tau \in [0,T]\,. \end{cases}$$

2.1.2. Portfolio of options with the butterfly spread delta function

We also take into account a portfolio of options using the butterfly spread delta function as the final condition. The payoff function for this case is given by

$$\widehat{U}(S,T) = \begin{cases} 1, & \text{if } S \in (S_1,S_2) \\ -1, & \text{if } S \in (S_2,S_3)\,, \\ 0, & \text{else} \end{cases}$$

and the following are the boundary conditions:

$$\begin{cases} \widehat{U}(0,\tau) = 0, \quad \tau \in [0,T]\,, \\ \lim\limits_{S\to\infty} \widehat{U}(S,\tau) = 0, \quad \tau \in [0,T]\,. \end{cases}$$

2.2. Transformation

Since (1) is degenerate and backward in time (as we have the final condition $V(S,T)$), so first we transform it into non-degenerate and forward in time using the transformation

$$\begin{cases} S = e^x, \\ \tau = T - t, \\ \widetilde{V}(x,t) = \widehat{U}(e^x, T-t). \end{cases}$$

This will give us

$$\begin{cases} \dfrac{\partial \widetilde{V}}{\partial t} - \dfrac{1}{2}\sigma^2 \dfrac{\partial^2 \widetilde{V}}{\partial x^2} - \left(r - \delta - \dfrac{1}{2}\sigma^2\right)\dfrac{\partial \widetilde{V}}{\partial x} + r\widetilde{V} = 0, \qquad (x,t) \in \mathbb{R}\times(0,T]\,, \\ \widetilde{V}(x,0) = \widehat{U}_T(e^x), \qquad x \in \mathbb{R}, \\ \widetilde{V}(x,t) = 0, \qquad (x,t) \in \partial\mathbb{R}\times[0,T]\,. \end{cases} \tag{2}$$

2.3. Localization

In (2), one can see that a spatial variable is unbounded; hence, in order to apply the numerical method we are going to truncate x from $\mathbb{R}$ to $\Omega := (x_{\min}, x_{\max}) \subset \mathbb{R}$ as in [14], where $x_{\max}$ and $x_{\min}$ are suitably chosen numbers in $\mathbb{R}$. Now, the problem that needs to be solved is:

$$\begin{cases} \dfrac{\partial w}{\partial t} - \dfrac{1}{2}\sigma^2 \dfrac{\partial^2 w}{\partial x^2} - \left(r - \delta - \dfrac{1}{2}\sigma^2\right)\dfrac{\partial w}{\partial x} + rw = f(x,t), \quad (x,t) \in \Omega\times(0,T]\,, \\ w(x,0) = w_0(x), \qquad x \in \overline{\Omega}, \\ w(x,t) = 0, \qquad (x,t) \in \partial\Omega\times[0,T]\,. \end{cases} \tag{3}$$

Hence, we have

$$\mathcal{L}w := \frac{\partial w}{\partial t} + \widetilde{\mathcal{L}}w = f,$$

where

$$\widetilde{\mathcal{L}}w := -\frac{1}{2}\sigma^2\frac{\partial^2 w}{\partial x^2} - \left(r - \delta - \frac{1}{2}\sigma^2\right)\frac{\partial w}{\partial x} + rw.$$

Here, we assume $0 < \sigma, r - \delta \ll 1$, and with the condition that $r - \delta > \dfrac{1}{2}\sigma^2$.

Here we consider the values of σ and $r - \delta$ very small; generally, the solution of this type of PDEs contains a layer at boundaries $x = x_{\min}$ and $x = x_{\max}$, where the width and nature of the boundary layer are defined by the solution to the following problem:

$$-\frac{1}{2}\sigma^2 w'' - \left(r - \delta - \frac{1}{2}\sigma^2\right)w' + rw = 0. \tag{4}$$

Now for (4), the auxiliary equation is given by

$$\frac{1}{2}\sigma^2\lambda^2 + \left(r - \delta - \frac{1}{2}\sigma^2\right)\lambda - r = 0. \tag{5}$$

Since the product of roots of (5) is $-\frac{2r}{\sigma^2}$, with both σ and r positive, this implies

$$-\frac{2r}{\sigma^2} < 0.$$

Hence we have the roots of opposite sign. Consider $\lambda_1 < 0$ and $\lambda_2 > 0$ are two roots of equation (5) such that

$$\lambda_{1,2} = \frac{-\left(r-\delta-\frac{1}{2}\sigma^2\right) \pm \sqrt{\left(r-\delta-\frac{1}{2}\sigma^2\right)^2 + 2\sigma^2 r}}{\sigma^2},$$

consider η_L and η_R such that

$$\begin{cases} \eta_L = |\lambda_1| = \dfrac{\left(r-\delta-\frac{1}{2}\sigma^2\right) + \sqrt{\left(r-\delta-\frac{1}{2}\sigma^2\right)^2 + 2\sigma^2 r}}{\sigma^2}, \\ \eta_R = \lambda_2 = \dfrac{-\left(r-\delta-\frac{1}{2}\sigma^2\right) + \sqrt{\left(r-\delta-\frac{1}{2}\sigma^2\right)^2 + 2\sigma^2 r}}{\sigma^2}. \end{cases}$$

Here the values η_L and η_R decide the width of boundary layers at $x = x_{\min}$ and $x = x_{\max}$, respectively. At $x = x_{\min}$, layer behaves like $e^{-\eta_L(x+x_{\min})}$, while at $x = x_{\max}$ it is characterized by $e^{-\eta_R(x_{\max}-x)}$. In order to perform error analysis in layer-adapted meshes, we need a pertinent breakdown of the solution. We divide the solution w of transformed model (3) itself into the left and right singular components v_L and v_R, respectively, and the regular component, v such that $w = u + u_L + u_R$. In [11], a brief analysis of the comprehensive decomposition and the associated bounds for both components is provided.

3. Mesh discretization

In this section, we will discretize both the spatial and temporal variable domains.

3.1. Temporal discretization

For the temporal variable, we discretize the time interval $[0, T]$ by using uniform meshes as

$$\Omega^N = \left\{ t_n = n\Delta t, \quad \Delta t = \frac{T}{N},\ n = 0, 1, \ldots, N \right\}.$$

For the spatial discretization we consider various types of layer-adapted meshes.

3.2. Spatial discretization

To discretize the spatial variable, we will employ nonuniform meshes, including the Shishkin mesh, the Bakhvalov-type mesh, and the exponentially-graded mesh.

3.2.1. Shishkin mesh

The Shishkin mesh is a highly efficient piecewise uniform mesh. It is essential to ensure that M is a multiple of 4 and greater than or equal to 8. For $j = L, R$, the values of σ_j are presented by

$$\sigma_j = \frac{\tau_j}{\eta_j} \ln M \leq \frac{(x_{\max} - x_{\min})}{4}, \quad j = L, R,$$

and in the Shishkin mesh, the mesh size is given by

$$h_i = \begin{cases} \dfrac{4\sigma_L}{M}, & i = 1, 2, \ldots, \dfrac{M}{4}, \\ \dfrac{2\left(x_{\max} - x_{\min} - \sigma_L - \sigma_R\right)}{M}, & i = \dfrac{M}{4} + 1, \ldots, \dfrac{3M}{4}, \\ \dfrac{4\sigma_R}{M}, & i = \dfrac{3M}{4} + 1, \ldots, M. \end{cases}$$

Therefore, we have

$$h_i = \begin{cases} \mathcal{O}\left((\eta_L M)^{-1} \ln M\right), & i = 1, 2, \ldots, \dfrac{M}{4}, \\ \mathcal{O}\left(M^{-1}\right), & i = \dfrac{M}{4} + 1, \ldots, \dfrac{3M}{4}, \\ \mathcal{O}\left((\eta_R M)^{-1} \ln M\right), & i = \dfrac{3M}{4} + 1, \ldots, M, \end{cases}$$

here $x_i = x_{i-1} + h_i,\ i = 1, 2, \ldots, M$.

3.2.2. Bakhvalov-type mesh

For the Bakhvalov-type mesh, we require $M \geq 8$, where M is divisible by 4. Following this, we segment our domain Ω into distinct subdomains as $\Omega = \Omega_L \bigcup \Omega_C \bigcup \Omega_R$, where $\Omega_L = (x_{\min}, x_{\min} + \sigma_L),\ \Omega_C = (x_{\min} + \sigma_L, x_{\max} - \sigma_R),\ \Omega_R = (x_{\max} - \sigma_R, x_{\max})$ with the values of $\sigma_j,$ for $j = L, R$ are presented by

$$\sigma_j = \frac{\tau_j}{\eta_j} \ln \eta_j \leq \frac{(x_{\max} - x_{\min})}{4},$$

and $\tau_j \geq 1,\ j = R, L,$ are parameters chosen by the user. Now we will look at two mesh generation functions Φ_L and Φ_R given as

$$\begin{cases} \Phi_R(s) = -\ln\left(1 - 4(1-s)\left(1 - \eta_R^{-1}\right)\right), \\ \Phi_L(s) = -\ln\left(1 - 4s\left(1 - \eta_L^{-1}\right)\right). \end{cases}$$

This will give us mesh discretization as

$$x_i = \begin{cases} x_{\min} + \dfrac{\tau_L}{\eta_L}\Phi_L\left(s_i\right), & i = 0, 1, \ldots, \dfrac{M}{4}, \\ x_{\min} + \sigma_L + 2\left(s_i - \dfrac{1}{4}\right)\left(x_{\max} - x_{\min} - \sigma_L - \sigma_R\right), & i = \dfrac{M}{4} + 1, \ldots, \dfrac{3M}{4}, \\ x_{\max} - \dfrac{\tau_R}{\eta_R}\Phi_R\left(s_i\right), & i = \dfrac{3M}{4} + 1, \ldots, M, \end{cases}$$

with $s_i = i/M,\ i = 0, \ldots, M$. Here $h_i = (x_i - x_{i-1})$ is the length of the segment $I_i = (x_{i-1}, x_i)$ and we have

$$h_i = \begin{cases} \mathcal{O}\left(\eta_L^{-1}\right), & i = 1, 2, \ldots, \dfrac{M}{4}, \\ \mathcal{O}\left(M^{-1}\right), & i = \dfrac{M}{4} + 1, \ldots, \dfrac{3M}{4}, \\ \mathcal{O}\left(\eta_R^{-1}\right), & i = \dfrac{3M}{4} + 1, \ldots, M. \end{cases}$$

3.2.3. Exponentially graded mesh

The exponentially graded mesh is an example of a graded mesh category. We must choose an integer M for the exponentially graded mesh such that M is divisible by 4 and M is greater than or equal to 8. Here the mesh generating functions Φ_L and Φ_R are given by

$$\begin{cases} \Phi_L(s) = -\ln\left(1 - 4s\theta_L\right), \\ \Phi_R(s) = -\ln\left(1 - 4(1-s)\theta_R\right), \end{cases}$$

with θ_j, for $j = L, R$ are given by

$$\theta_j = 1 - e^{(-q\eta_j)/(2\tau_j)} \in (0, 1).$$

Then we have mesh discretization for the exponentially graded mesh as

$$x_i = \begin{cases} x_{\min} + \dfrac{\tau_L}{q\eta_L}\Phi_L(s_i), & i = 0, 1, \ldots, \dfrac{M}{4} - 1, \\ x_{\frac{M}{4}-1} + \left(\dfrac{x_{\frac{3M}{4}+1} - x_{\frac{M}{4}-1}}{M/2+2}\right)\left(i - \frac{M}{4} + 1\right), & i = \dfrac{M}{4}, \ldots, \dfrac{3M}{4}, \\ x_{\max} - \dfrac{\tau_R}{q\eta_R}\Phi_R(s_i), & i = \dfrac{3M}{4} + 1, \ldots, M, \end{cases}$$

with $s_i = i/M$, where $i = 0, \ldots, M$.

Here $h_i = (x_i - x_{i-1})$ is the length of the segment $I_i = (x_{i-1}, x_i)$ and

$$h_i = \begin{cases} \mathcal{O}\left(\eta_L^{-1}\right), & i = 1, 2, \ldots, \dfrac{M}{4} - 1, \\ \mathcal{O}\left(M^{-1}\right), & i = \dfrac{M}{4}, \ldots, \dfrac{3M}{4}, \\ \mathcal{O}\left(\eta_R^{-1}\right), & i = \dfrac{3M}{4} + 1, \ldots, M. \end{cases}$$

Assumptions: Throughout this paper, we assume the following:

$$\begin{cases} \dfrac{1}{\eta_j}\ln\eta_j \le \mathcal{C}M^{-1}, \quad j = L, R, \\ \max\{\sigma, r - \delta\} \le \mathcal{C}M^{-1}, \\ q\tau_j \ge 2, \quad j = L, R. \end{cases} \tag{6}$$

4. Numerical scheme

In this section, we develop a fully-discrete scheme by applying the Crank-Nicolson method to the temporal variable and SDFEM to the spatial variable. First of all, the weak formulation of (3) is written as:

$$\begin{cases} \text{Find } w : (0, T] \to H_0^1(x_{\min}, x_{\max}), \text{ such that} \\ (w_t, v) + B(w, v) = (f, v), \quad \forall v \in H_0^1(x_{\min}, x_{\max}), \end{cases} \tag{7}$$

where the bilinear form $B(w, v)$ is given by

$$B(w, v) = \frac{\sigma^2}{2}(w', v') - \left(r - \delta - \frac{\sigma^2}{2}\right)(w', v) + (rw, v), \quad \forall v \in H_0^1(x_{\min}, x_{\max}),$$

and $(\cdot, \cdot)$ is the standard inner product in $L^2(x_{\min}, x_{\max})$. Now, at time $t_n = n\Delta t$, if w_h^n denotes the fully discretized solution, then SDFEM and Crank-Nicolson discretization (CN-SDFEM) for (3) are:

$$\begin{cases} \text{For } n = 1, \ldots, N \text{ find } w_h^n \in V_h, \text{ such that} \\ \left(w_h^n - w_h^{n-1}, v_h\right) + \Delta t B_{SD}\left(\dfrac{w_h^n + w_h^{n-1}}{2}, v_h\right) = \Delta t\left(\dfrac{f^n + f^{n-1}}{2}, v_h\right) \\ -\Delta t \displaystyle\sum_{i=1}^{M} \delta_i \left(\frac{f^n + f^{n-1}}{2}, \left(r - \delta - \frac{\sigma^2}{2}\right) v_h'\right)_{I_i} \\ + \displaystyle\sum_{i=1}^{M} \delta_i \left(w_h^n - w_h^{n-1}, \left(r - \delta - \frac{\sigma^2}{2}\right) v_h'\right)_{I_i}, \quad \forall v_h \in V_h, \end{cases} \tag{8}$$

with $w_h^0(x)$ suitably approximated, and $\delta_i,\ i = 1,\ldots,M$, is the stabilization parameter, where a bilinear form $B_{SD}(.,.)$ is given by

$$\begin{aligned} B_{SD}\left(w_h, v_h\right) =& \frac{\sigma^2}{2}\left(w_h', v_h'\right) - \left(r - \delta - \frac{\sigma^2}{2}\right)\left(w_h', v_h\right) + \left(r w_h, v_h\right) \\ & - \sum_{i=1}^{M} \delta_i \left(-\frac{\sigma^2}{2} w_h'' - \left(r - \delta - \frac{\sigma^2}{2}\right) w_h' + r w_h, \left(r - \delta - \frac{\sigma^2}{2}\right) v_h'\right)_{I_i}, \end{aligned} \tag{9}$$

where $V_h \in H_0^1(x_{\min}, x_{\max})$ is the finite element space made up of continuously piecewise linear functions.

The streamline-diffusion norm is defined by

$$|||v_h|||_{SD} = \left(\frac{\sigma^2}{2}\left\|v_h'\right\|^2 + r\left\|v_h\right\|^2 + \sum_{i=1}^{M}\left\|\sqrt{\delta_i}\left(r - \delta - \frac{\sigma^2}{2}\right) v_h'\right\|_{L^2(I_i)}^2\right)^{1/2}. \tag{10}$$

Lemma 1. *If the following inequality is satisfied by the stabilization parameter δ_i*

$$0 \le \delta_i \le \frac{1}{r},\ i = 1,\ldots,M,$$

then the bilinear form $B_{SD}(.,.)$ will satisfy the following coercive property:

$$B_{SD}\left(v_h, v_h\right) \ge \frac{1}{2}|||v_h|||_{SD}^2, \quad \forall v_h \in V_h.$$

Proof. For the proof, one can refer to [1]. □

In this paper, following [1], we consider the value of the stabilization parameter δ_i as

$$\delta_i = \begin{cases} 0, & i = 1, 2, \ldots, \dfrac{M}{4}, \\ \delta_c, & i = \dfrac{M}{4} + 1, \ldots, \dfrac{3M}{4}, \\ 0, & i = \dfrac{3M}{4} + 1, \ldots, M, \end{cases}$$

where the value of δ_c is given by

$$\delta_c = \begin{cases} M^{-1}, & \text{if } \left(r - \delta - \frac{1}{2}\sigma^2\right) < M^{-1}, \\ \left(r - \delta - \frac{1}{2}\sigma^2\right)^{-2} M^{-3}, & \text{if } \left(r - \delta - \frac{1}{2}\sigma^2\right) \ge M^{-1}. \end{cases}$$

Theorem 1. *For each $n = 1, 2, \ldots, N$, the solution w_h^n of (8) satisfies the following stability bound with the condition $4N \le M$:*

$$\left\|w_h^n\right\|^2 + \frac{\Delta t}{4}\sum_{i=1}^{n}|||w_h^i + w_h^{i-1}|||_{SD}^2 \le 3^n\left(\left\|w_h^0\right\|^2 + \Delta t\left(\frac{1}{r} + \frac{1}{M^{3/2}}\right)\sum_{i=1}^{n}\left\|f^i + f^{i-1}\right\|^2\right).$$

Proof. For the proof, one can refer to [1]. □

Remark 1. If in place of the Crank-Nicolson scheme we apply the Backward Euler scheme for discretization of the temporal variable and SDFEM for the spatial variable (BE-SDFEM), then (8) will be written as:

$$\begin{cases} \text{For } n = 1, \ldots, N \text{ find } w_h^n \in V_h, \text{ such that} \\ \left(w_h^n - w_h^{n-1}, v_h\right) + \Delta t B_{SD}\left(w_h^n, v_h\right) = \Delta t\left(f^n, v_h\right) - \Delta t\sum_{i=1}^{M}\delta_i\left(f^n, \left(r - \delta - \frac{\sigma^2}{2}\right) v_h'\right)_{I_i} \\ + \sum_{i=1}^{M}\delta_i\left(w_h^n - w_h^{n-1}, \left(r - \delta - \frac{\sigma^2}{2}\right) v_h'\right)_{I_i}, \quad \forall v_h \in V_h, \end{cases}$$

where B_{SD} is the same as in (9).

In this case equivalent to Theorem 1, we have the following stability bound for BE-SDFEM:

$$\left\|w_h^n\right\|^2 + \frac{\Delta t}{2}\sum_{i=1}^{n}\left|\!\left|\!\left|w_h^i\right|\!\right|\!\right|_{SD}^2 \le \left\|w_h^0\right\|^2 + 2\Delta t\left(\frac{1}{r}+\frac{2}{M}\right)\sum_{i=1}^{n}\left\|f^i\right\|^2.$$

5. Uniform error estimate

The numerical error can be expressed as the sum of interpolation error and discretization error as

$$w(t_n) - w_h^n = \left(w(t_n) - \mathcal{Q}_h w(t_n)\right) + \left(\mathcal{Q}_h w(t_n) - w_h^n\right), \tag{11}$$

where $w(t_n)$ denotes the exact solution of (7) at $t = t_n$ and $\mathcal{Q}_h w$ is a linear interpolant of w. Since the H^1-norm in (10) prevents us from getting second-order convergence, we will use a discrete version of SD-norm (10)

$$\left|\!\left|\!\left|v_h\right|\!\right|\!\right|_{SD,d} = \left(\frac{\sigma^2}{2}\sum_{i=1}^{M}h_i\sum_{j=1}^{2}b_j|v_h'(x_{i,j})|^2 + r\left\|v_h\right\|^2 + \sum_{i=1}^{M}\left\|\sqrt{\delta_i}\left(r-\delta-\frac{\sigma^2}{2}\right)v_h'\right\|_{L^2(I_i)}^2\right)^{1/2},$$

in which the term corresponding to the H^1 norm is estimated by two-point Gaussian quadrature rule, and b_j and $x_{i,j}$ are corresponding weight functions and Gaussian points in each subinterval I_i, respectively.

Theorem 2. *The following bound is satisfied by the interpolation error in exponentially graded and Bakhvalov-type meshes if conditions given in (6) hold true:*

$$\left(\Delta t\sum_{i=1}^{N}\left|\!\left|\!\left|w(t_i) - \mathcal{Q}_h w(t_i)\right|\!\right|\!\right|_{SD,d}^2\right)^{\frac{1}{2}} \le \mathcal{C}M^{-2}.$$

Proof. From [1], for the Bakhvalov-type mesh, if conditions in (6) are fulfilled, then we have

$$\left|\!\left|\!\left|u(t_i) - \mathcal{Q}_h u(t_i)\right|\!\right|\!\right|_{SD,d} \le \mathcal{C}M^{-2},\ \forall i = 1,2,\ldots,N, \tag{12}$$

and

$$\left|\!\left|\!\left|u_j(t_i) - \mathcal{Q}_h u_j(t_i)\right|\!\right|\!\right|_{SD,d} \le \mathcal{C}M^{-2},\ \forall i = 1,2,\ldots,N,\ \text{and}\ j = L,R. \tag{13}$$

Using (12) and (13), we get

$$\begin{aligned}\Delta t\sum_{i=1}^{N}\left|\!\left|\!\left|w(t_i) - \mathcal{Q}_h w(t_i)\right|\!\right|\!\right|_{SD,d}^2 &\le \Delta t\sum_{i=1}^{N}\Big(\left|\!\left|\!\left|u(t_i) - \mathcal{Q}_h u(t_i)\right|\!\right|\!\right|_{SD,d}^2 + \left|\!\left|\!\left|u_L(t_i) - \mathcal{Q}_h u_L(t_i)\right|\!\right|\!\right|_{SD,d}^2 \\ &\quad + \left|\!\left|\!\left|u_R(t_i) - \mathcal{Q}_h u_R(t_i)\right|\!\right|\!\right|_{SD,d}^2\Big) \\ &\le \mathcal{C}\Delta t\sum_{i=1}^{N}M^{-4} \\ &\le \mathcal{C}M^{-4}.\end{aligned}$$

Similarly, one can prove this for the exponentially graded mesh. □

Remark 2. The interpolation error bound for the Shishkin mesh, that is an equivalent form of Theorem 2, is

$$\left(\Delta t\sum_{i=1}^{N}\left|\!\left|\!\left|w(t_i) - \mathcal{Q}_h w(t_i)\right|\!\right|\!\right|_{SD,d}^2\right)^{1/2} \le \mathcal{C}M^{-2}\ln^2 M.$$

Theorem 3. *For the Bakvalov-type mesh, if conditions in (6) are satisfied, then discretization error satisfies the following bound:*

$$\left(\Delta t \sum_{i=1}^{N} \left\| \frac{\mathcal{Q}_h w\left(t_i\right) - w_h^i}{2} + \frac{\mathcal{Q}_h w\left(t_{i-1}\right) - w_h^{i-1}}{2} \right\|_{SD}^2 \right)^{1/2} \leq \mathcal{C}\left((\Delta t)^2 + M^{-2}\right).$$

Proof. For the proof, one can refer to [1]. □

Theorem 4. *Suppose that assumptions in (6) hold. Then the error estimate of our scheme (8) satisfies*

$$\left(\Delta t \sum_{i=1}^{N} \left\| \frac{w\left(t_i\right) - w_h^i}{2} + \frac{w\left(t_{i-1}\right) - w_h^{i-1}}{2} \right\|_{SD,d}^2 \right)^{1/2} \leq \mathcal{C}\left((\Delta t)^2 + M^{-2}\right).$$

Proof. The proof follows from Theorems 2 and 3 after using triangle inequality on (11). □

6. Numerical experiments

For all examples considered here, the exact solution are not available, therefore we measure the error using the double mesh principle, and for CN-SDFEM, it is given by

$$E_{CN}^{M,N} = \left(\Delta t \sum_{i=1}^{N} \left\| \frac{e\left(x, t_i\right)}{2} + \frac{e\left(x, t_{i-1}\right)}{2} \right\|_{SD,d}^2 \right)^{1/2},$$

while for BE-SDFEM, the error is given by

$$E_{BE}^{M,N} = \left(\Delta t \sum_{i=1}^{N} |||e\left(x, t_i\right)|||_{SD,d}^2 \right)^{1/2},$$

where $e(x, t_i) = u^{M,N}(x, t_i) - u^{2M,2N}(x, t_i)$.

Here N and M denote the total number of intervals in the temporal variable and the spatial variable used for discretization in respective directions. Additionally, the convergence order is computed by

$$R^{M,N} = \log_2\left(\frac{E^{M,N}}{E^{2M,2N}}\right).$$

Example 1. Consider the model problem given in (3) for the long call butterfly spread option with $f(x, t) = (1 - x^2)\exp(t^2)$, where we take $\Omega = (-1, 1)$, $K_1 = 0.5$, $K_2 = 1$, $K_3 = 1.5$, $r = 0.06$, $T = 1$, $\sigma = 10^{-5}$ and $r - \delta = 10^{-10}$.

M	N	Shishkin mesh		Bakhvalov-type mesh		Exponentially graded	
		Error	Order	Error	Order	Error	Order
32	8	2.1904e − 03		2.1905e − 03		2.2128e − 03	
64	16	5.4686e − 04	2.0020	5.4656e − 04	2.0028	5.5956e − 04	1.9835
128	32	1.3682e − 04	1.9990	1.3657e − 04	2.0007	1.4038e − 04	1.9950
256	64	3.4319e − 05	1.9951	3.4139e − 05	2.0002	3.5296e − 05	1.9918
512	128	8.6395e − 06	1.9900	8.5351e − 06	1.9999	8.9186e − 06	1.9846
1024	256	2.1898e − 06	1.9802	2.1366e − 06	1.9981	2.2829e − 06	1.9659

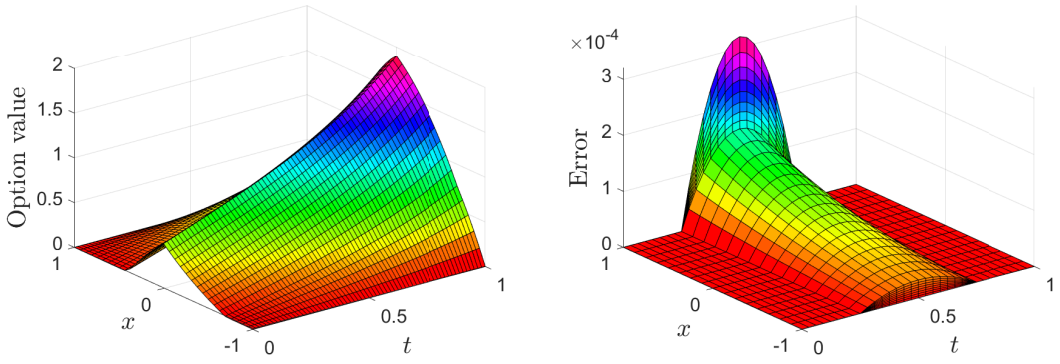
Table 1: Error and convergence order for CN-SDFEM for Example 1

M	N	Shishkin mesh		Bakhvalov-type mesh		Exponentially graded	
		Error	Order	Error	Order	Error	Order
32	8	2.6549e – 02		2.6549e – 02		2.6573e – 02	
64	16	1.1531e – 02	1.2031	1.1531e – 02	1.2032	1.1543e – 02	1.2030
128	32	5.3497e – 03	1.1080	5.3494e – 03	1.1080	5.3525e – 03	1.1087
256	64	2.5736e – 03	1.0557	2.5734e – 03	1.0557	2.5742e – 03	1.0561
512	128	1.2618e – 03	1.0283	1.2617e – 03	1.0283	1.2619e – 03	1.0285
1024	256	6.2469e – 04	1.0143	6.2464e – 04	1.0143	6.2469e – 04	1.0144

Table 2: Error and convergence order for BE-SDFEM for Example 1

Table 1 presents the error and the corresponding order of convergence for Example 1 with various layer-adapted meshes when using Crank-Nicolson for temporal variable discretization. The computational error and the order of convergence for BE-SDFEM are presented in Table 2.

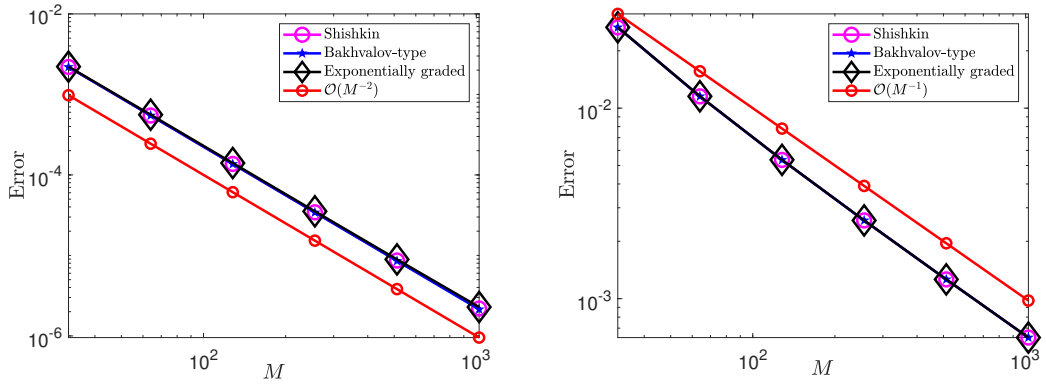
In Figure 1(a), we have illustrated the numerical solution obtained for Example 1 using CN-SDFEM with a Bakhvalov-type mesh. Additionally, Figure 1(b) presents the corresponding error plot.



(a) Numerical solution (b) Error plot

Figure 1: Numerical solution and error plot for Example 1 using CN-SDFEM with the Bakhvalov-type mesh and $M = N = 64$

Also, for Example 1, we have plotted M and the error in a log-log scale for CN-SDFEM in Figure 2(a) and for BE-SDFEM in Figure 2(b).



(a) CN-SDFEM (b) BE-SDFEM

Figure 2: Log-log plots for Example 1

The slope of the lines indicates that the order of the CN-SDFEM method is the same as the line with a slope of two in Figure 2(a). Similarly, in Figure 2(b), it can be seen that the order of the BE-SDFEM method is the same as the line with a slope of one, and the error values of all three meshes are overlapping.

Remark 3 (Finite difference method). The fully discrete scheme for solving the model problem described in (3) using the finite difference method, particularly implicit Euler in the time derivative and the central difference method in the space derivative, is as follows:

$$\begin{cases} V_i^0 = w_0(x_i), \\ \left(-\dfrac{\sigma^2}{2h^2} - \dfrac{1}{2h}\left(r - \delta - \dfrac{\sigma^2}{2}\right)\right) V_{i+1}^{n+1} + \left(\dfrac{1}{\Delta t} + \dfrac{\sigma^2}{h^2} + r\right) V_i^{n+1} \\ + \left(-\dfrac{\sigma^2}{2h^2} + \dfrac{1}{2h}\left(r - \delta - \dfrac{\sigma^2}{2}\right)\right) V_{i-1}^{n+1} = \dfrac{1}{\Delta t} V_i^n + f(x_i, t_n), \quad i = 1, 2, \ldots, M-1, \\ V_0^n = 0 \text{ and } V_M^n = 0, \text{ for } n = 1, 2, \ldots, N. \end{cases} \tag{14}$$

In order to effectively compare our method with other existing techniques, we will analyze the finite difference method given in (14) and the cubic spline method proposed by Huang and Cen [6]. Their work solved the generalized Black-Scholes PDE using implicit Euler for time discretization and the cubic spline method for spatial variable discretization. We plan to use the double mesh principle to calculate the error in the maximum norm for all of these methods, including our proposed SDFEM technique using a Bakhvalov-type mesh, the finite difference method, and the cubic spline method for Example 1. Here, M and N represent the discretization intervals in the spatial and temporal variables, respectively. Table 3 shows the error and the convergence order for these cases.

M	N	Method by [6]		Finite difference (14)		CN-SDFEM	
		Error	Order	Error	Order	Error	Order
32	8	5.5060e − 02		7.9738e − 03		5.0979e − 03	
64	16	2.6344e − 02	1.0635	4.4902e − 03	0.8285	1.2778e − 03	1.9962
128	32	1.2872e − 02	1.0332	2.3896e − 03	0.9100	3.1967e − 04	1.9990
256	64	6.3609e − 03	1.0170	1.2336e − 03	0.9540	7.9936e − 05	1.9997
512	128	3.1616e − 03	1.0086	6.2682e − 04	0.9767	1.9995e − 05	1.9992
1024	256	1.5761e − 03	1.0043	3.1596e − 04	0.9883	5.0201e − 06	1.9939

Table 3: Error and convergence order for the proposed scheme, finite difference (14) and the method by [6]

The suggested method has a lower error than the one presented in [6] and the finite difference method (14), as shown in Table 3. The superior performance of SDFEM is evident from the results in Table 3. Additionally, the semi-log plot comparing grid points and error in Figure 3 clearly illustrates that the suggested method outperforms the approach presented in [6] and the finite difference method (14).

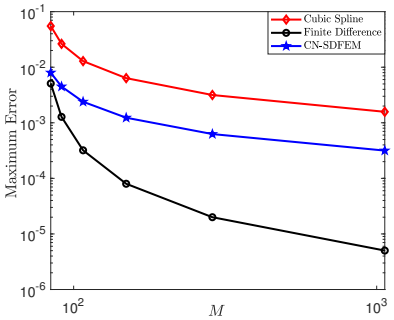


Figure 3: The semi-log plot of M vs. maximum error for the method by [6], finite difference (14) and the proposed technique

Example 2. Consider the model problem given in (3) for the long put butterfly spread option with $f(x,t) = (1-x^2)\exp(t^2)$, where we take $\Omega = (-1,1)$, $K_1 = 0.45$, $K_2 = 0.90$, $K_3 = 1.35$, $r = 0.06$, $T = 1$, $\sigma = 10^{-5}$ and $r - \delta = 10^{-10}$.

For Example 2, the computational error and the corresponding order of convergence for CN-SDFEM are presented in Table 4. Here one can see that the order of convergence for the CN-SDFEM method is two for all three types of nonuniform meshes.

M	N	Shishkin mesh		Bakhvalov-type mesh		Exponentially graded	
		Error	Order	Error	Order	Error	Order
32	8	2.1914e – 03		2.1905e – 03		2.2310e – 03	
64	16	5.4734e – 04	2.0014	5.4656e – 04	2.0028	5.6545e – 04	1.9802
128	32	1.3704e – 04	1.9978	1.3657e – 04	2.0007	1.5953e – 04	1.8256
256	64	3.4228e – 05	2.0014	3.4142e – 05	2.0001	3.7134e – 05	2.1030
512	128	8.6813e – 06	1.9792	8.5607e – 06	1.9957	8.9016e – 06	2.0606
1024	256	2.1862e – 06	1.9895	2.1363e – 06	2.0026	2.2769e – 06	1.9670

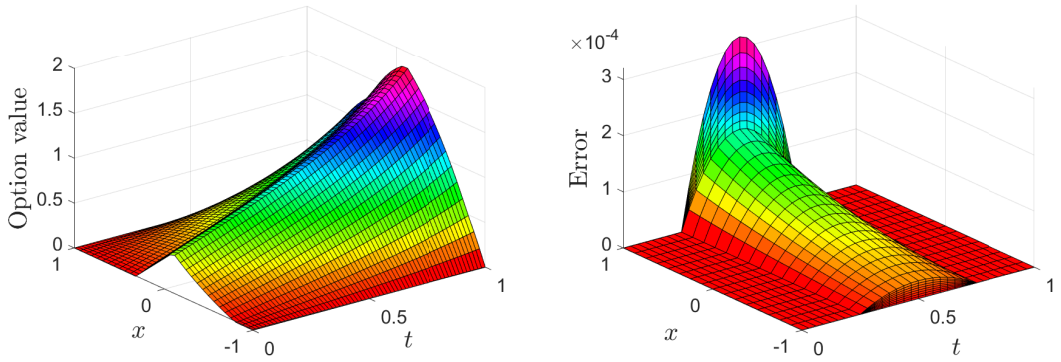
Table 4: Error and convergence order for CN-SDFEM for Example 2

Table 5 presents the error and the order of convergence of BE-SDFEM for Example 2, and it can be seen that the order of convergence is one for all three types of meshes.

M	N	Shishkin mesh		Bakhvalov-type mesh		Exponentially graded	
		Error	Order	Error	Order	Error	Order
32	8	2.6548e – 02		2.6547e – 02		2.6592e – 02	
64	16	1.1531e – 02	1.2031	1.1530e – 02	1.2032	1.1546e – 02	1.2035
128	32	5.3495e – 03	1.1080	5.3490e – 03	1.1080	5.3458e – 03	1.1109
256	64	2.5731e – 03	1.0559	2.5732e – 03	1.0557	2.5733e – 03	1.0548
512	128	1.2616e – 03	1.0282	1.2616e – 03	1.0283	1.2618e – 03	1.0282
1024	256	6.2464e – 04	1.0142	6.2459e – 04	1.0143	6.2464e – 04	1.0144

Table 5: Error and convergence order for BE-SDFEM for Example 2

The numerical solution for Example 2 achieved using CN-SDFEM with a Bakhvalov-type mesh is visually presented in Figure 4(a). Furthermore, the corresponding error plot is depicted in Figure 4(b).



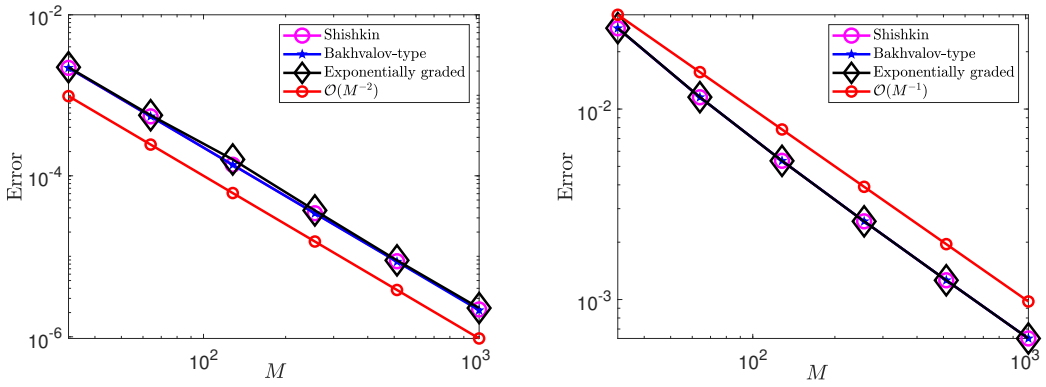
(a) Numerical solution (b) Error plot

Figure 4: Numerical solution and error plot for Example 2 using CN-SDFEM with the Bakhvalov-type mesh and $M = N = 64$

In Figure 5(a), we have plotted M and the error in a log-log scale of CN-SDFEM for Example 2. The

rate of decrease in the error as we increase the grid points is shown by the log-log plot. The log-log plot offers an enhanced visualisation of results when there is a very tiny or huge quantity is involved. Mainly, a logarithmic scale is used for the comparison.

For the CN-SDFEM method, the slope of lines of all types of layer-adapted meshes is the same as the line with slope two which assures its second-order accuracy, and again error values of all types of meshes are overlapping.



(a) CN-SDFEM (b) BE-SDFEM

Figure 5: Log-log plots for Example 2

We have plotted M and the error in a log-log scale for BE-SDFEM in Figure 5(b) for Example 2. Here all three types of meshes are overlapping and have a slope which is parallel to the line with slope one which indicates first-order accuracy of BE-SDFEM.

Example 3. Consider the model problem given in (3) for a portfolio of options using the butterfly spread delta function with $f(x,t) = (1-x^2)\exp(t^2)$, where we take $\Omega = (-1,1)$, $S_1 = 0$, $S_2 = 0.25$, $S_3 = 0.5$, $r = 0.06$, $T = 1$, $\sigma = 10^{-8}$ and $r - \delta = 10^{-16}$.

The error and the associated order of convergence for Example 3 are displayed in Table 6 for different layer-adapted meshes when using Crank-Nicolson for temporal variable discretization.

M	N	Shishkin mesh		Bakhvalov-type mesh		Exponentially graded	
		Error	Order	Error	Order	Error	Order
16	4	8.8228e − 03		8.8228e − 03		1.0821e − 02	
32	8	2.1906e − 03	2.0099	2.1906e − 03	2.0099	2.7571e − 03	1.9725
64	16	5.4656e − 04	2.0028	5.4656e − 04	2.0028	6.9761e − 04	1.9827
128	32	1.3657e − 04	2.0007	1.3657e − 04	2.0007	1.7557e − 04	1.9904
256	64	3.4142e − 05	2.0001	3.4139e − 05	2.0002	4.4047e − 05	1.9949
512	128	8.5447e − 06	1.9984	8.5364e − 06	1.9997	1.1032e − 05	1.9973

Table 6: Error and convergence order for CN-SDFEM for Example 3

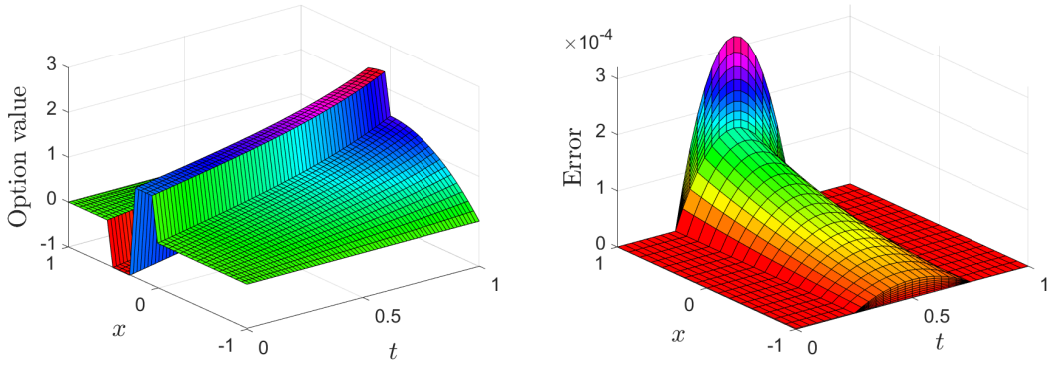
Table 7 displays the BE-SDFEM error and the order of convergence for Example 3. All three types of meshes exhibit the same order of convergence.

The numerical solution for Example 3 using CN-SDFEM with a Bakhvalov-type mesh is depicted in Figure 6(a). Furthermore, the corresponding error plot can be found in Figure 6(b).

Additionally, Figure 7(a) shows the plot of M and the error in a log-log scale for CN-SDFEM. The CN-SDFEM method guarantees second-order accuracy because the slope of the lines in all types of layer-adapted meshes is the same as a line with a slope of two.

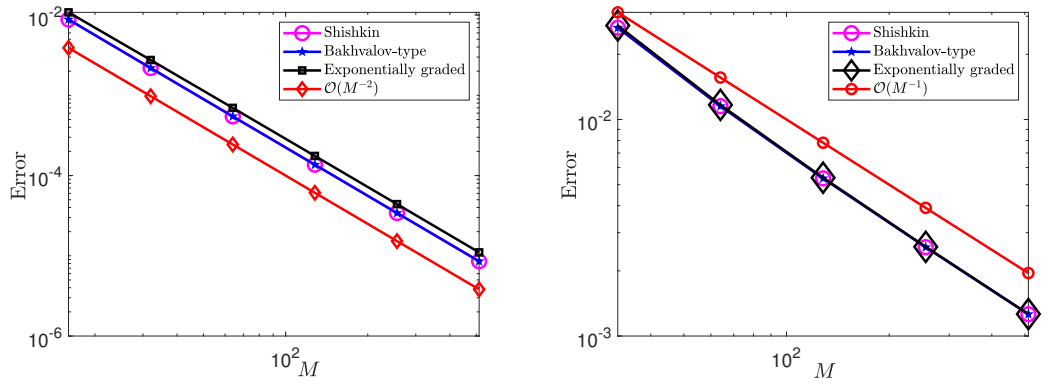
M	N	Shishkin mesh		Bakhvalov-type mesh		Exponentially graded	
		Error	Order	Error	Order	Error	Order
32	8	2.6536e – 02		2.6536e – 02		2.7159e – 02	
64	16	1.1525e – 02	1.2033	1.1525e – 02	1.2033	1.1678e – 02	1.2176
128	32	5.3464e – 03	1.1081	5.3464e – 03	1.1081	5.3845e – 03	1.1169
256	64	2.5719e – 03	1.0557	2.5719e – 03	1.0557	2.5813e – 03	1.0607
512	128	1.2609e – 03	1.0283	1.2609e – 03	1.0283	1.2633e – 03	1.0309

Table 7: Error and convergence order for BE-SDFEM for Example 3



(a) Numerical solution (b) Error plot

Figure 6: Numerical solution and error plot for Example 3 using CN-SDFEM with the Bakhvalov-type mesh and $M = N = 64$



(a) CN-SDFEM (b) BE-SDFEM

Figure 7: Log-log plots for Example 3

In Figure 7(b), we plotted M and the error in a log-log scale for BE-SDFEM in Example 3. In this case, error values of all three mesh types overlap and have a slope parallel to the line with a slope of one, indicating BE-SDFEM's first-order accuracy.

7. Conclusion

In this article, we rigorously examine the numerical solution of the Black-Scholes equation through the application of the Crank-Nicolson/backward-Euler and streamline-diffusion finite element method (CN-SDFEM/BE-SDFEM). We begin by transforming the Black-Scholes PDE to enable the application of

numerical methods. The time derivative is discretized using the Crank-Nicolson/backward-Euler scheme on uniform meshes, while the spatial variable is discretized using SDFEM on nonuniform meshes. The CN-SDFEM method delivers second-order convergent solutions, whereas the BE-SDFEM method provides first-order accuracy. We derive stability and convergence results and validate the proposed theoretical error bounds through numerous numerical experiments. We have utilized various types of layer-adapted meshes, each with its own significance and properties. While it's challenging to determine the best option, the Bakhvalov-type mesh is worth considering as supported by theoretical and numerical evidence. Our proposed approach has been proven to be more accurate and efficient than the method described in [6].

Acknowledgements

The authors express their sincere thanks to the referee whose valuable comments helped to improve the presentation.

The Indian Institute of Technology in Guwahati is acknowledged for the funding it provides to the first author.

References

[1] D. AVIJIT and S. NATESAN, Convergence analysis of a fully-discrete FEM for singularly perturbed two-parameter parabolic PDE, *Math. Comput. Simul.* **197** (2022), 185–206.

[2] S. BANSAL and S. NATESAN, Richardson extrapolation technique for generalized Black–Scholes PDEs for European options, *Appl. Comput. Math.* **42** (2023), no. 5, 238.

[3] S. BANSAL and S. NATESAN, A novel higher-order efficient computational method for pricing European and Asian options, *Numer. Algorithms* (2024), 1–33.

[4] F. BLACK and M. SCHOLES, The pricing of options and corporate liabilities, *J. Political Econ.* **81** (1973), no. 3, 637–654.

[5] A. CHACUR, M. ALI, and J. SALAZAR, Real options pricing by the finite element method, *Comput. Math. Appl.* **61** (2011), no. 9, 2863–2873.

[6] J. HUANG and Z. CEN, Cubic spline method for a generalized Black-Scholes equation, *Math. Probl. Eng.* **2014** (2014).

[7] M. KADALBAJOO, L. TRIPATHI, and A. KUMAR, A cubic B-spline collocation method for a numerical solution of the generalized Black-Scholes equation, *Math. Comput. Model.* **55** (2012), no. 3-4, 1483–1505.

[8] R. KANGRO and R. NICOLAIDES, Far field boundary conditions for Black-Scholes equations, *SIAM J. Numer. Anal.* **38** (2000), no. 4, 1357–1368.

[9] O. LADYZHENSKAIA, V. SOLONNIKOV, and N. URAL'TSEVA, *Linear and Quasi-linear Equations of Parabolic Type*, American Mathematical Society, Providence, R.I., 1968.

[10] R. MOHAMMADI, Quintic B-spline collocation approach for solving generalized Black-Scholes equation governing option pricing, *Comput. Math. Appl.* **69** (2015), no. 8, 777–797.

[11] E. O'RIORDAN, M. PICKETT, and G. SHISHKIN, Parameter-uniform finite difference schemes for singularly perturbed parabolic diffusion-convection-reaction problems, *Math. Comp.* **75** (2006), no. 255, 1135–1154.

[12] S. RAO and M. SRIVASTAVA, Numerical solution of generalized Black-Scholes model, *Appl. Math. Comput.* **321** (2018), 401– 421.

[13] R. SEYDEL, *Tools for Computational Finance*, Springer, London, 2017.

[14] P. WILMOTT, J. DEWYNNE, and S. HOWISON, *Option Pricing: Mathematical Models and Computation*, Oxford Financial Press, Oxford, 1993.