A numerical method for solving time-dependent convection-diffusion problems

Mhamed El Merzguioui, Abdelmajid El hajaji, Khalid Hilal, and Lalla Saadia Chadli

ABSTRACT: In this paper, we develop a new numerical method for solving a time-dependent convection-diffusion equation with Dirichlet’s type boundary conditions. We first propose the $\theta$-method, $\theta \in [1/2, 1]$ ($\theta = 1$ corresponds to the back-ward Euler method and $\theta = 1/2$ corresponds to the Crank-Nicolson method) to discretize the temporal variable, resulting in a linear partial differential equation (PDE). To numerically solve this linear PDE, we develop and analyze a new cubic spline collocation method for the spatial discretization. To solve the discretized linear system, we design a collocation method and we prove that the method is second order convergent. The computed results are compared wherever possible with those already available in the literature.

Key Words: Convection-diffusion equation, $\theta$-method, Spline collocation method.

Contents

1 Introduction 217
2 The time semidiscretization and Description of the $\theta$-method 219
3 Spatial discretization and cubic spline collocation method 220
4 Numerical Experiments 223
  4.1 Example 1. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 224
  4.2 Example 2. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 225
5 Conclusion 226

1. Introduction

The time-dependent convection-diffusion equation is a parabolic partial differential equation, which describes physical phenomena where energy is transformed inside a physical system due to two processes: convection and diffusion. The term convection means the movement of molecules within fluids, whereas, diffusion describes the spread of particles through random motion from regions of higher concentration to regions of lower concentration. It is necessary to calculate the transport of fluid properties or trace constituent concentrations within a fluid for applications such as water quality modeling, air pollution, meteorology, oceanography and other physical sciences. When velocity field is complex, changing in time and transport process cannot be analytically calculated, and then numerical

2000 Mathematics Subject Classification: 35B40, 35L70
approximations to the time-dependent convection-diffusion equation are indispensable. Various numerical techniques have been developed and compared for solving the time-dependent convection-diffusion equation. Clavero et al. [1] give a uniform convergent numerical method with respect to the diffusion parameter to solve the one-dimensional time-dependent convection-diffusion problem. The used the implicit Euler method for the time discretization and the simple upwind finite difference scheme on a Shishkin mesh for the spatial discretization in [1]. Ramos presented an exponentially fitted method for singularly perturbed parameter [2]. Surla and Jerkovic considered a singularly perturbed boundary value problem using a spline collocation method in [3]. The main objective of this study is to develop a user friendly, economical method which can work for time-dependent convection-diffusion equation by using a cubic splines collocation method and compared with other numerical methods given in the literature in particular with [7].

Consider the singularly perturbed initial-boundary value problem

\[ \begin{aligned}
\frac{\partial u}{\partial t} - L_\epsilon u &= f, & (x, t) \in (0, 1) \times (0, T], \\
u(x, 0) &= \phi_0(x), & x \in [0, 1], \\
u(0, t) &= \nu(1, t) = 0, & t \in [0, T].
\end{aligned} \]

(1.1)

where \( L_\epsilon : C(\Omega) \cap C^{2,1}(\Omega) \to C(\Omega), \) with \( \Omega = (0, 1) \times (0, T), \) is defined by

\[ L_\epsilon := \epsilon \frac{\partial^2}{\partial x^2} - a(x) \frac{\partial}{\partial x} - b(x, t)I, \quad (x, t) \in \Omega \times (0, T), \]

(1.2)

With \( a(x) \geq \tilde{\alpha} > 0, \) \( b = b(x, t) \geq 0 \) on \( \overline{\Omega}. \) The diffusion coefficient \( \epsilon \) is a small positive parameter. and \( a, b, \phi_0 \) are sufficiently smooth functions.

Here we assume that the problem satisfies sufficient regularity and compatibility conditions which guarantee the problem has a unique solution \( u \in C(\Omega) \cap C^{2,1}(\Omega) \) satisfying (see, [4,5,6]):

\[ \left| \frac{\partial^{i+j} u(x, t)}{\partial x^i \partial t^j} \right| \leq k(1 + \epsilon^{-i}e^{-\alpha(1-x)/\epsilon}) \quad \text{on} \ \overline{\Omega}; \quad 0 \leq j \leq 3 \text{ and } 0 \leq i + j \leq 4, \]

(1.3)

where \( k \) is a constant.

In the present work, we present a numerical method for solving a time-dependent convection-diffusion problem. The method is based on \( \theta \)-method to discretize the temporal variable and a cubic spline collocation method for the spatial discretization. The scheme is second-order convergent with respect to the spatial variable.

The organization of the paper is as follows. In Section 2, we discuss time semi-discretization. Section 3 is devoted to the spline collocation method for time-dependent convection-diffusion equation using a cubic spline collocation method. Next, the error bound of the spline solution is analyzed. In order to validate the theoretical results presented in this paper, we present numerical tests on two known examples in Section 4. The obtained numerical results are compared to the ones given in [7]. Finally, a conclusion is given in Section 5.
2. The time semidiscretization and Description of the $\theta$-method

The observation period has to be specified first and is set from today (time point $t = 0$) to time point $T$. This period will be divided into $M$ equally spaced time intervals with length $\Delta t = \frac{T}{M}$, and we denote by $t_m = m\Delta t$.

We discretize the variable time in (1.1) by means of $\theta$-method, $\theta \in \left[\frac{1}{2}, 1\right]$. Then the semi-discretization yields the following system of equations:

$$\frac{u^{m+1} - u^m}{\Delta t} - \theta L^m u^{m+1} - (1 - \theta)L^m u^m = f^{m+\theta}.\tag{2.1}$$

That is

$$(I - \theta \triangle tL^m)u^{m+1} = (I + (1 - \theta)\triangle tL^m)u^m + \triangle tf^{m+\theta},$$

where, $L^m : C(\overline{\Omega}) \cap C^2(\Omega) \to C(\Omega)$ is the differential operator defined by

$$L^m = \frac{\partial^2}{\partial x^2} + \beta^m + \theta \frac{\partial}{\partial x} + \gamma^m I,$$

with

- $u^m$ approximate the exact solution $u(x, t_m)$ at the time level $t_m = m\Delta t$,
- $\beta^m + \theta = -a(x)$,
- $\gamma^m + \theta = -b(x, t_m + \theta \triangle t)$.

$\theta = 1/2$, corresponds to the Crank-Nicolson method, and $\theta = 1$, corresponds to the back-ward Euler method.

Then, the approximate problem of (1.1) is

$$\begin{cases}
p^{m+\theta}uxx + q^{m+\theta}(x)ux + l^{m+\theta}(x)u = g^{m+\theta}, & x \in \Omega_x, \\
u^0(x) = \phi_0(x), & x \in \Omega_x, \\
u^{m+1}(0) = u^{m+1}(1) = 0, & 0 \leq m < M,\tag{2.2}
\end{cases}$$

where, for any $m \geq 0$ and for any $x \in \Omega_x = (0, 1)$, we have

- $p^{m+\theta} = \theta \triangle t \epsilon$,
- $q^{m+\theta}(x) = \theta \triangle t \beta^{m+\theta}(x)$,
- $l^{m+\theta}(x) = \theta \triangle t \gamma^{m+\theta}(x) - 1$,
- $g^{m+\theta}(x) = -(I + (1 - \theta)\triangle tL^m_x)u^m - \triangle tf^{m+\theta}$.

We have $u^0(x) = \phi_0(x)$. Then, for $0 \leq m \leq M - 1$, $u^m$ being known, we obtain $u^{m+1}$ as a solution of problem (2.2).

The following theorem proves the order of convergence of the solution $u^m$ to $u(x, t_m)$.
Theorem 2.1. (see [8,9]) Problem (2.2) is second order convergent for $\theta = \frac{1}{2}$ and first order convergent for $\theta \in \left[\frac{1}{2}, 1\right]$ i.e.,
\[
\|u(x, t_m) - u^m\|_\infty \leq \text{Cte}(\triangle t)^2 \text{ for } \theta = \frac{1}{2}
\]
\[
\|u(x, t_m) - u^m\|_\infty \leq \text{Cte} \triangle t \text{ for } \theta \in \left[\frac{1}{2}, 1\right].
\]

For any $m \geq 0$, problem (2.2) has a unique solution and can be written on the following form:
\[
\begin{cases}
p u''(x) + q(x)u'(x) + l(x)u(x) = g(x), & x \in \Omega_x \\
u(0) = u(1) = 0
\end{cases}
\] (2.3)

In the sequel of this paper, we will focus on the solution of problem (2.3).

3. Spatial discretization and cubic spline collocation method

In this section we construct a cubic spline which approximates the solution $u$ of problem (2.3), in the interval $\Omega_x = (0, 1) \subset \mathbb{R}$.

We denote by $\| \cdot \|$ the Euclidean norm on $\mathbb{R}^{n+1}$, $\| \cdot \|_\infty$ the uniform norm.

Let $\Theta = \{ 0 = x_{-3} = x_{-2} = x_{-1} = x_0 < x_1 < \cdots < x_{n-1} < x_n = x_{n+1} = x_{n+2} = x_{n+3} = 1 \}$ be a subdivision of the interval $\Omega_x$. Without loss of generality, we put $x_j = jh$, where $0 \leq j \leq n$ and $h = \frac{1}{n}$. Denote by $S_4(\Omega_x, \Theta) = P_3^2(\Omega_x, \Theta)$ the space of piecewise polynomials of degree 3 over the subdivision $\Theta$ and of class $C^2$ everywhere on $\Omega_x$. Let $B_i$, $i = -3, \cdots, n-1$, be the B-splines of degree 3 associated with $\Theta$. These B-splines are positives and form a basis of the space $S_4(\Omega_x, \Theta)$.

Proposition 3.1. Let $u$ be the solution of problem (2.3). Then, there exists a unique cubic spline interpolant $S \in S_4(\Omega_x, \Theta)$ of $u$ which satisfies:
\[
S(\tau_j) = u(\tau_j), \quad j = 0, \cdots, n + 2,
\]
where $\tau_0 = x_0$, $\tau_i = \frac{x_{j-1} + x_j}{2}$, $j = 1, \cdots, n$, $\tau_{n+1} = x_{n-1}$ and $\tau_{n+2} = x_n$.

Proof: Using the Schoenberg-Whitney theorem (see [10]), it is easy to see that there exits a unique cubic spline which interpolates $u$ at the points $\tau_i, i = 0, \cdots, n + 2$.

If we put $S = \sum_{i=-3}^{n-1} c_i B_i$, then by using the boundary conditions of problem (2.3) we obtain $c_{-3} = S(0) = u(0) = 0$ and $c_{n-1} = S(1) = u(1) = 0$. Hence
\[
S = \sum_{i=-2}^{n-2} c_i B_i
\]
Furthermore, since the interpolation with splines of degree $d$ gives uniform norm errors of order $O(h^{d+1})$ for the interpolant, and of order $O(h^{d+1-r})$ for the $r$th derivative of the interpolant (see [2], for instance), then for any $u \in C^4(\Omega_\varepsilon)$ we have

$$p . S^{(2)}(\tau_j) + q(\tau_j) S^{(1)}(\tau_j) + l(\tau_j) S^{(0)}(\tau_j) = g(\tau_j) + O(h^2), \quad j = 1, ..., n + 1,$$

(3.1)

The cubic spline collocation method, that we present in this paper, constructs numerically a cubic spline $S = \sum_{i=3}^{n-1} \tilde{c}_i B_i$ which satisfies the equation of problem (2.3) at the points $\tau_j, \ j = 0, \ldots, n + 2$. It is easy to see that $\tilde{c}_{-3} = 0$ and $\tilde{c}_{n-1} = 0$, and the coefficients $\tilde{c}_i, i = -2, \ldots, n - 1$ satisfy the following:

$$p . \tilde{S}^{(2)}(\tau_j) + q(\tau_j) \tilde{S}^{(1)}(\tau_j) + l(\tau_j) \tilde{S}^{(0)}(\tau_j) = g(\tau_j), \quad j = 1, ..., n + 1.$$  

(3.2)

Taking $C = [c_{-2}, \ldots, c_{n-2}]^T$ and \( \tilde{C} = [\tilde{c}_{-2}, \ldots, \tilde{c}_{n-2}]^T \), and using equations (3.1) and (3.2), we get:

$$PA_h^{(2)} + QA_h^{(1)} + LA_h^{(0)}C = F + E, \quad (3.3)$$

$$PA_h^{(2)} + QA_h^{(1)} + LA_h^{(0)}\tilde{C} = F, \quad (3.4)$$

with:

$$F = [f_1, \ldots, f_{n+1}]^T, \text{ and } f_j = g(\tau_j),$$

$$E = [O(h^2), \ldots, O(h^2)]^T \in \mathbb{R}^{n+1},$$

$$P = (\text{diag}(\Delta t \theta I_j))_{1 \leq j \leq n+1},$$

$$Q = (\text{diag}(\Delta t \theta \beta^{m+\theta} (\tau_j)))_{1 \leq j \leq n+1},$$

$$L = (\text{diag}(\Delta t \theta \gamma^{m+\theta} (\tau_j) - 1))_{1 \leq j \leq n+1},$$

$$A_h^{(k)} = (B_{-3+ k}^{(k)}(\tau_j))_{1 \leq j, l \leq n+1}, \quad k = 0, 1, 2.$$  

It is well known that $A_h^{(k)} = \frac{1}{h^k} A_k$ for $k = 0, 1, 2$ where matrices $A_0, A_1$ and $A_2$ are independent of $h$, with the matrix $A_2$ is invertible [11].

We deduce that (3.3) and (3.4) can be written also in the following form

$$PA_2 \left( I + A_2^{-1} P^{-1} (hQA_1 + h^2 LA_0) \right) C = h^2 F + h^2 E \quad (3.5)$$

$$PA_2 \left( I + A_2^{-1} P^{-1} (hQA_1 + h^2 LA_0) \right) \tilde{C} = h^2 F, \quad (3.6)$$

In order to determine the bounded of $\| C - \tilde{C} \|$, we need the following remark.

**Remark 3.2.** For a small real $h$ such that

$$\| A_2^{-1} P^{-1} \|_{\infty} \left( h \| Q \|_{\infty} \| A_1 \|_{\infty} + h^2 \| L \|_{\infty} \| A_0 \|_{\infty} \right) < 1,$$
the matrix \((I + A_2^{-1}P^{-1}(hQA_1 + h^2LA_0))^{-1}\) exists, and

\[
\| (I + A_2^{-1}P^{-1}(hQA_1 + h^2LA_0))^{-1} \|_\infty < \frac{1}{1 - (h\|Q\|_\infty \|A_1\|_\infty + h^2\|L\|_\infty \|A_0\|_\infty)}.
\]

Hence, in this case, there exists a unique cubic spline that approximates the exact solution \(u\) of problem (2.3).

**Proposition 3.3.** If we choose the real \(h\) such that

\[
\|A_2^{-1}P^{-1}\|_\infty (h\|Q\|_\infty \|A_1\|_\infty + h^2\|L\|_\infty \|A_0\|_\infty) \leq \frac{1}{2},
\]

then there exists a constant \(cte\) that depends only on the functions \(p, q, l\) and \(f\) such that

\[
\|C - \tilde{C}\| \leq cte.h^2.
\]

**Proof:** from relations (3.5) and (3.6) we have

\[
C - \tilde{C} = h^2(PA_2)^{-1} (I + (PA_2)^{-1}(hQA_1 + h^2LA_0))^{-1} E.
\]

Since \(E = O(h^2)\), then there exists a constant \(K_1\) such that \(\|E\| \leq K_1h^2\). This implies that

\[
\|C - \tilde{C}\| \leq K_1 \frac{h^2\|A_2P\|^{-1}\|\|_\infty}{1 - h^2\|A_2P\|^{-1}\|\|_\infty} h^2
\]

Using the inequality \(\|A_2^{-1}P^{-1}\|_\infty (h\|Q\|_\infty \|A_1\|_\infty + h^2\|L\|_\infty \|A_0\|_\infty) \leq \frac{1}{2}\), and \(0 < h \leq 1\), we deduce that

\[
\|C - \tilde{C}\| \leq K_1 \frac{h^2\|A_2P\|^{-1}\|\|_\infty}{\|Q\|_\infty \|A_1\|_\infty + \|L\|_\infty \|A_0\|_\infty} h^2
\]

Finally, we deduce that

\[
\|C - \tilde{C}\| \leq cte.h^2.
\]

\[
\Box
\]

Now, we are in position to prove the main theorem of our work.

**Theorem 3.4.** The spline approximation \(\tilde{S}\) converges quadratically to the exact solution \(u\) of problem (2.3), i.e., \(\|u - \tilde{S}\|_\infty = O(h^2)\).
Proof: It is well known that \( \| u - S \|_{\infty} = O(h^4) \) (see \([10]\)), so \( \| u - S \|_{\infty} \leq Kh^4 \), where \( K \) is a positive constant. On the other hand we have
\[
S(x) - \tilde{S}(x) = \sum_{i=-2}^{n-2} (c_i - \tilde{c}_i)B_i(x).
\]

Therefore, by using (3.8) and \( \sum_{i=-2}^{n-2} B_i(x) \leq 1 \), we get
\[
|S(x) - \tilde{S}(x)| \leq \| C - \tilde{C} \| \sum_{i=-2}^{n-2} B_i(x) \leq \| C - \tilde{C} \| \leq cte.h^2.
\]

Since \( \| u - \tilde{S} \|_{\infty} \leq \| u - S \|_{\infty} + \| S - \tilde{S} \|_{\infty} \), we deduce the stated result. \( \square \)

**Theorem 3.5.** We suppose that \( u(x,t) \) is the solution of (1.1) and \( u_c(x,t) \) is the approximate solution by our presented method, then we have,
\[
\| u(x,t_m) - u_c(x,t_m) \|_{\infty} \leq C_1(\Delta t^2 + h^2) \text{ for } \theta = \frac{1}{2} \\
\| u(x,t_m) - u_c(x,t_m) \|_{\infty} \leq C_2(\Delta t + h^2) \text{ for } \theta \in \left[\frac{1}{2},1\right].
\]

where \( C_1 \) and \( C_2 \) are finite constants. Therefore for sufficiently small \( \Delta t \) and \( h \), the solution of presented scheme (3.3) converges to the solution of initial-boundary value problem (1.1) in the discrete \( L_{\infty} \) - norm and the rates of convergence are \( O(\Delta t^2 + h^2) \) and \( O(\Delta t^2 + h^2) \).

4. Numerical Experiments

In this section we verify experimentally theoretical results obtained in the previous section. If the exact solution is known, then at time \( t \leq T \) the maximum error \( E_{\epsilon}^{\text{max}} \) can be calculated as:
\[
E_{\epsilon}^{\text{max}} = \max_{x \in [0,1], t \in [0,T]} | S^{M,N}(x,t) - u(x,t) |.
\]

Otherwise it can be estimated by the following double mesh principle:
\[
E_{\epsilon, M,N}^{\text{max}} = \max_{x \in [0,1], t \in [0,T]} | S^{M,N}(x,t) - S^{2M,2N}(x,t) |,
\]

where \( S^{M,N}(x,t) \) is the numerical solution on the \( M + 1 \) grids in space and \( N + 1 \) grids in time, and \( S^{2M,2N}(x,t) \) is the numerical solution on the \( 2M + 1 \) grids in space and \( 2N + 1 \) grids in time.

In this section, we present the numerical results of present method on two problems presented in the paper of the authors C.Clavero et al. \([7]\). We tested the accuracy of this method for different values of \( N, M, \epsilon \) and compared the obtained results to the ones given in \([7]\).
4.1. Example 1.

Consider the time-dependent convection-diffusion problem (1.1) with the following data:

\[ \Omega = (0,1) \times (0,1), \text{ and } \phi_0 = 0, \]
\[ a(x) = 1 + x^2 + \frac{1}{2} \sin(\pi x), \]
\[ b(x,t) = 1 + x^2 + \sin\left(\frac{\pi t}{2}\right), \]
\[ f(x,t) = x^3(1-x)^3 + t(1-t) \sin(\pi t). \]

Table 1 shows values of the maximum error \(\text{max \_ error}\) obtained in our numerical experiments for different values of \(N\), \(M\) and \(\epsilon\), we note the convergence of the solution \(S\) to the function \(u\) depends on the discretization parameters \(h\), \(\Delta t\) and the parameter \(\epsilon\). Theorem 6 shown the convergence of the method provided that the parameters \(\epsilon\), \(h\) and \(\Delta t\) satisfy the relation (3.7).

Table 2 shows values of the maximum error \(\text{max \_ error}\) obtained in our numerical experiments and the one obtained in [7]. Table 2 illustrates the comparative performance of our method over another existing method [7]. The obtained approximate numerical solutions show that our proposed method maintains better accuracy compared with a recent other existing method [7] based on the hybrid numerical method.

From these tables, we can say that the results of our schemes are acceptable, and conclude that the proposed schemes are feasible and valid. We observe that present method is nearly of second order of convergence with respect to these error norms.

Table 1: Numerical results for \(\theta = \frac{1}{2}\).

<table>
<thead>
<tr>
<th>(N)</th>
<th>32</th>
<th>64</th>
<th>128</th>
<th>256</th>
<th>512</th>
</tr>
</thead>
<tbody>
<tr>
<td>(M)</td>
<td>16</td>
<td>32</td>
<td>64</td>
<td>128</td>
<td>256</td>
</tr>
<tr>
<td>(\epsilon=2^{-2})</td>
<td>(\text{max _ error})</td>
<td>0.56075 \times 10^{-3}</td>
<td>0.23896 \times 10^{-3}</td>
<td>0.11801 \times 10^{-3}</td>
<td>0.61301 \times 10^{-4}</td>
</tr>
<tr>
<td>(\epsilon=2^{-3})</td>
<td>(\text{max _ error})</td>
<td>0.59979 \times 10^{-3}</td>
<td>0.25169 \times 10^{-3}</td>
<td>0.12173 \times 10^{-3}</td>
<td>0.61512 \times 10^{-4}</td>
</tr>
<tr>
<td>(\epsilon=2^{-4})</td>
<td>(\text{max _ error})</td>
<td>0.62299 \times 10^{-3}</td>
<td>0.25910 \times 10^{-3}</td>
<td>0.12374 \times 10^{-3}</td>
<td>0.61633 \times 10^{-4}</td>
</tr>
<tr>
<td>(\epsilon=2^{-5})</td>
<td>(\text{max _ error})</td>
<td>0.63569 \times 10^{-3}</td>
<td>0.26310 \times 10^{-3}</td>
<td>0.12479 \times 10^{-3}</td>
<td>0.61708 \times 10^{-4}</td>
</tr>
</tbody>
</table>
4.2. Example 2.

Consider the time-dependent convection-diffusion problem (1.1) with the following data:
\[ \Omega = (0, 1) \times (0, 1), \quad \phi_0 = 0, \]
\[ a(x) = 4 + 4x - x^2 - e^x, \]
\[ b(x, t) = 5xt, \]
\[ f(x, t) = 3\sin(\pi x) + 5(e^t - 1). \]

Table 3 shows values of the maximum error (max_error) obtained in our numerical experiments for different values of \( N, \) \( M \) and \( \epsilon, \) we note the convergence of the solution \( S \) to the function \( u \) depends on the discretization parameters \( h, \Delta t \) and the parameter \( \epsilon. \) Theorem 6 shown the convergence of the method provided that the parameters \( \epsilon, h \) and \( \Delta t \) satisfy the relation (3.7).

Table 4 shows values of the maximum error (max_error) obtained in our numerical experiments and the one obtained in [7]. The results show that the errors in our methods are smaller than errors of the methods in [7].

<table>
<thead>
<tr>
<th>( N )</th>
<th>32</th>
<th>64</th>
<th>128</th>
<th>256</th>
<th>512</th>
</tr>
</thead>
<tbody>
<tr>
<td>( M )</td>
<td>16</td>
<td>32</td>
<td>64</td>
<td>128</td>
<td>256</td>
</tr>
<tr>
<td>For ( \epsilon = 2^{-10} )</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>our max_error</td>
<td>0.6425x10^{-3}</td>
<td>0.2651x10^{-3}</td>
<td>* * *</td>
<td>* * *</td>
<td>* * *</td>
</tr>
<tr>
<td>max_error in [7]</td>
<td>0.249x10^{-3}</td>
<td>0.142x10^{-3}</td>
<td>* * *</td>
<td>* * *</td>
<td>* * *</td>
</tr>
<tr>
<td>For ( \epsilon = 2^{-9} )</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>our max_error</td>
<td>0.6482x10^{-3}</td>
<td>0.2670x10^{-3}</td>
<td>0.1247x10^{-3}</td>
<td>0.6170x10^{-4}</td>
<td>0.2466x10^{-4}</td>
</tr>
<tr>
<td>max_error in [7]</td>
<td>0.262x10^{-3}</td>
<td>0.153x10^{-3}</td>
<td>0.829x10^{-4}</td>
<td>0.434x10^{-4}</td>
<td>0.222x10^{-4}</td>
</tr>
<tr>
<td>For ( \epsilon = 2^{-8} )</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>our max_error</td>
<td>0.6490x10^{-3}</td>
<td>0.2672x10^{-3}</td>
<td>0.1247x10^{-3}</td>
<td>0.6172x10^{-4}</td>
<td>0.2467x10^{-4}</td>
</tr>
<tr>
<td>max_error in [7]</td>
<td>0.265x10^{-3}</td>
<td>0.154x10^{-3}</td>
<td>0.840x10^{-4}</td>
<td>0.439x10^{-4}</td>
<td>0.225x10^{-4}</td>
</tr>
<tr>
<td>For ( \epsilon = 2^{-7} )</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>our max_error</td>
<td>0.6490x10^{-3}</td>
<td>0.2673x10^{-3}</td>
<td>0.1247x10^{-3}</td>
<td>0.6173x10^{-4}</td>
<td>0.2468x10^{-4}</td>
</tr>
<tr>
<td>max_error in [7]</td>
<td>0.266x10^{-3}</td>
<td>0.154x10^{-3}</td>
<td>0.841x10^{-4}</td>
<td>0.440x10^{-4}</td>
<td>0.225x10^{-4}</td>
</tr>
</tbody>
</table>
Table 3: Numerical results for $\theta = \frac{1}{2}$.

<table>
<thead>
<tr>
<th>$N$</th>
<th>84</th>
<th>168</th>
<th>336</th>
<th>672</th>
<th>1344</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M$</td>
<td>5</td>
<td>20</td>
<td>80</td>
<td>320</td>
<td>1280</td>
</tr>
<tr>
<td>$\epsilon = 2^{-2}$</td>
<td>max error</td>
<td>0.60231x10^{-3}</td>
<td>0.25895x10^{-3}</td>
<td>0.10874x10^{-3}</td>
<td>0.32201x10^{-4}</td>
</tr>
<tr>
<td>$\epsilon = 2^{-4}$</td>
<td>max error</td>
<td>0.65814x10^{-3}</td>
<td>0.27101x10^{-3}</td>
<td>0.11365x10^{-3}</td>
<td>0.32107x10^{-4}</td>
</tr>
<tr>
<td>$\epsilon = 2^{-6}$</td>
<td>max error</td>
<td>0.67545x10^{-3}</td>
<td>0.27412x10^{-3}</td>
<td>0.11541x10^{-3}</td>
<td>0.32213x10^{-4}</td>
</tr>
<tr>
<td>$\epsilon = 2^{-8}$</td>
<td>max error</td>
<td>0.68000x10^{-3}</td>
<td>0.28061x10^{-3}</td>
<td>0.11627x10^{-3}</td>
<td>0.32358x10^{-4}</td>
</tr>
</tbody>
</table>

Table 4: Numerical results for $\theta = \frac{1}{2}$.

<table>
<thead>
<tr>
<th>$N$</th>
<th>84</th>
<th>168</th>
<th>336</th>
<th>672</th>
<th>1344</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M$</td>
<td>5</td>
<td>20</td>
<td>80</td>
<td>320</td>
<td>1280</td>
</tr>
<tr>
<td>$\epsilon = 2^{-9}$</td>
<td>our max error</td>
<td>0.68077x10^{-3}</td>
<td>0.2812x10^{-3}</td>
<td>0.11622x10^{-3}</td>
<td>* * *</td>
</tr>
<tr>
<td>$\epsilon = 2^{-11}$</td>
<td>max error in [7]</td>
<td>0.616x10^{-3}</td>
<td>0.230x10^{-3}</td>
<td>0.681x10^{-3}</td>
<td>* * *</td>
</tr>
<tr>
<td>$\epsilon = 2^{-13}$</td>
<td>our max error</td>
<td>0.6814x10^{-3}</td>
<td>0.2818x10^{-3}</td>
<td>0.11640x10^{-3}</td>
<td>0.3235x10^{-4}</td>
</tr>
<tr>
<td>$\epsilon = 2^{-15}$</td>
<td>max error in [7]</td>
<td>0.618x10^{-3}</td>
<td>0.232x10^{-3}</td>
<td>0.689x10^{-3}</td>
<td>0.181x10^{-3}</td>
</tr>
<tr>
<td>$\epsilon = 2^{-17}$</td>
<td>our max error</td>
<td>0.6815x10^{-3}</td>
<td>0.2819x10^{-3}</td>
<td>0.11655x10^{-3}</td>
<td>0.3236x10^{-4}</td>
</tr>
<tr>
<td>$\epsilon = 2^{-19}$</td>
<td>max error in [7]</td>
<td>0.619x10^{-3}</td>
<td>0.232x10^{-3}</td>
<td>0.690x10^{-3}</td>
<td>0.181x10^{-3}</td>
</tr>
</tbody>
</table>

5. Conclusion

A numerical method is developed to solve a time-dependent convection-diffusion problems. This method is based on $\theta$-method for the temporal discretization and the cubic spline collocation method in the spatial direction. We have shown the convergence of the method provided that the parameters $\epsilon, h$ and $\Delta t$ satisfy the relation (3.7). Moreover we have provided an error estimate of order $O(h^2 + \Delta t^2)$ with respect to the maximum norm: $\| \cdot \|_{\infty}$. Numerical experiments were performed on two known models to validate the convergence and efficiency of the method. Comparisons of the computed results with exact solutions showed that the scheme is capable of solving the time-dependent convection-diffusion equation and
is also capable of producing highly accurate solutions with minimal computational
effort for both time and space. The computational results show that the proposed
numerical method is an efficient alternative method to the one proposed in [7].

References

1. C. Clavero, J.C. Jorge, F. Lisbona, Uniformly convergent scheme on a nonuniform mesh for

2. J.I. Ramos, An exponentially fitted method for singularly perturbed, one-dimensional,

3. K. Surla, V. Jerkovic, Some possibilities of applying spline collocations to singular perturbation
19-25.

4. A. Friedman, Partial Differential Equation of Parabolic Type, Robert E. Krieger Publishing

5. O.A. Ladyzenskaja, V.A. Solonnikov, N.N.Ural’ceva, Linear and Quasilinear Equations of

6. H.-G. Roos, M. Stynes, L. Tobiska, Robust numerical methods for singularly perturbed differen-

7. C. Clavero, J.L. Gracia, M. Stynes, A simpler analysis of a hybrid numerical method for time-
dependent convection-diffusion problems. Journal of Computational and Applied Mathemat-
ics 235 (2011) 5240-5248

8. M. K. Kadalbajoo, L. P. Tripathi, A. Kumar, A cubic B-spline collocation for a numerical
solution of generalized Black-Scholes equation, Mathematical and Computer Modelling, 55
(2012), 1483-1505.

doi:10.5269/bspm.v32i2.21354


M. El Merzguioui,  
Department of Mathematics,  
FST, University Abdelmalek Essaadi, Tanger,  
Morocco  
E-mail address: mh.elmerzguioui@gmail.com  

and  

A. El hajaji  
LERSEM Laboratory,  
ENCGJ, University Chouaïb Doukkali, El Jadida,  
Morocco  
E-mail address: a_elhajaji@yahoo.fr  

and  

K. Hilal  
LMC Laboratory,  
FST, University Sultan Moulay Slimane, Beni-Mellal,  
Morocco  
E-mail address: hilal_khalid@yahoo.fr  

and  

L. S. Chadli  
LMC Laboratory,  
FST, University Sultan Moulay Slimane, Beni-Mellal,  
Morocco  
E-mail address: chadli@fstbm.ac.ma