Applied Mathematical Modelling 36 (2012) 5555–5573

Contents lists available at SciVerse ScienceDirect

Applied Mathematical Modelling

journal homepage: www.elsevier.com/locate/apm

Redefined cubic B-splines collocation method for solving convection–diffusion equations

R.C. Mittal, R.K. Jain *

Department of Mathematics, I.I.T. Roorkee, Roorkee 247667, Uttarakhand, India

ARTICLE INFO

Article history:
Received 30 June 2011
Received in revised form 24 December 2011
Accepted 4 January 2012
Available online 13 January 2012

Keywords:
Convection–diffusion partial differential equation
Redefined cubic B-splines basis functions
Thomas algorithm

ABSTRACT

In this work, we discuss collocation method based on redefined cubic B-splines basis functions for solving convection–diffusion equation with Dirichlet's type boundary conditions. Stability of this method has been discussed and shown that it is unconditionally stable. The developed method is tested on various problems and the numerical results are reported in tabular form. The computed results are compared wherever possible with those already available in literature. The method is shown to work for Péclet number $\leqslant 10$. Easy and economical implementation process is the strength of it. This method can be easily extended to handle non-linear convection–diffusion partial differential equations.

1. Introduction

The 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 approximations to the convection equation are indispensable. Various numerical techniques have been developed and compared for solving the one dimensional convection–diffusion equation with constant coefficient [1–13]. Most of these techniques are based on the two-level finite difference approximations. In [14] several different numerical techniques will be developed and compared for solving the three-dimensional advection–diffusion equation with constant coefficient. These techniques are based on the two-level fully explicit and fully implicit finite difference approximations. In [15] new classes of high-order accurate methods have developed for solving the two-dimensional unsteady convection–diffusion equation based on the method of lines approach. In [16] a new practical scheme designing approach has presented whose application is based on the modified equivalent partial differential equation (MEPDE). These methods are second-order accurate and techniques that are third order or fourth order accurate. In [17] a variety of explicit and implicit algorithms has been studied dealing with the solution of the one-dimensional advection equation. These schemes are based on the weighted finite difference approximations. In [18] several finite difference schemes are discussed for solving the two-dimensional Schrodinger equation with Dirichlet's boundary conditions. In [19] several different computational LOD procedures were developed and discussed for solving the two-dimensional

* Corresponding author.
E-mail addresses: rcmmmfma@iitr.ernet.in (R.C. Mittal), rkjain_2000@rediffmail.com (R.K. Jain).

0307-904X/$ - see front matter © 2012 Elsevier Inc. All rights reserved.
doi:10.1016/j.apm.2012.01.009

transport equation. These schemes are based on the time-splitting finite difference approximations. In [20] the solution of Cauchy reaction–diffusion problem is presented by means of variational iteration method. The main objective of this study is to develop a user friendly, economical and stable method which can work for higher values of Péclet number for convection–diffusion equation by using redefined cubic B-splines collocation method.

Consider the one dimensional convection–diffusion equation

$$\frac{\partial u}{\partial t}+\varepsilon\frac{\partial u}{\partial x}=\gamma\frac{\partial^2 u}{\partial x^2},\quad 0<x<L,\quad 0<t\leqslant T, \tag{1}$$

with initial condition

$$u(x,0)=\varphi(x) \tag{2}$$

And boundary conditions are as follows:

$$u(0,t)=g_0(t),\quad u(1,t)=g_1(t),\quad t\in[0,T], \tag{3}$$

where the parameter γ is the viscosity coefficient and ε is the phase speed and both are assumed to be positive. φ, g_0 and g_1 are known functions with sufficient smoothness.

In the present work, we propose the collocation method based on redefined cubic B-spline basis functions for solving Eqs. (1)–(3), in which we first discretize the time derivative in the usual forward difference way and apply Crank Nicolson scheme to (1) to obtain a tri diagonal system of linear equations. The stability of the present method has been discussed by using von-Neumann stability scheme. It is shown that the proposed method is unconditionally stable.

The organization of the paper is as follows. In Section 2, description of the method is explained. In Section 3, procedure for redefined cubic B-splines collocation method for Dirichlet's type of boundary conditions is implemented to the equation. In Section 4, the stability analysis of the method is discussed. In Section 5, the procedure for evaluating the initial vector is explained which we require to start our present method. In Section 6, numerical experiments and discussion on six problems are given with graphical illustration. Conclusion of the presented method is given at the end of the paper in Section 7.

2. Description of the method

In cubic B-splines collocation method the approximate solution can be written as a linear combination of cubic B-spline basis functions for the approximation space under consideration.

We consider a mesh $0=x_0<x_1,\ldots,x_{N-1}<x_N=L$ as a uniform partition of the solution domain $0\leqslant x\leqslant L$ by the knots x_j with $h=x_{j+1}-x_j=\frac{L}{N},\ j=0,\ldots,N-1$.

Our numerical treatment for solving equation (1) using the collocation method with cubic B-spline is to find an approximate solution $U^N(x,t)$ to the exact solution $u(x,t)$ in the form [22]:

$$U^N(x,t)=\sum_{j=-1}^{N+1}\alpha_j(t)B_j(x), \tag{4}$$

where $\alpha_j(t)$ are unknown time dependent quantities to be determined from the boundary conditions and collocation from the differential equation.

The cubic B-splines $B_j(x)$ at the knots are given by

$$B_j(x)=\frac{1}{h^3}\begin{cases}(x-x_{j-2})^3 & x\in[x_{j-2},x_{j-1})\\ (x-x_{j-2})^3-4(x-x_{j-1})^3 & x\in[x_{j-1},x_j)\\ (x_{j+2}-x)^3-4(x_{j+1}-x)^3 & x\in[x_j,x_{j+1})\\ (x_{j+2}-x)^3 & x\in[x_{j+1},x_{j+2})\\ 0 & \text{otherwise}\end{cases}, \tag{5}$$

where $\{B_{-1},B_0,B_1,\ldots,B_{N-1},B_N,B_{N+1}\}$ forms a basis over the region $0\leqslant x\leqslant L$.

Each cubic B-spline covers four elements so that each element is covered by four cubic B-splines.

The values of $B_j(x)$ and its derivatives are tabulated in Table 1.

Table 1
Coefficient of cubic B-splines and its derivatives at knots x_j.

x	x_{j-z}	x_{j-1}	x_j	x_{j-1}	x_{j-z}
$B_j(x)$	0	1	4	1	0
$B_j(x)$	0	3/h	0	−3/h	0
$B_j''(x)$	0	$6/h^2$	$-12/h^2$	$6/h^2$	0

Using approximate function (4) and cubic B-spline functions (5), the approximate values of $U^N(x)$ and its two derivatives at the knots/nodes are determined in terms of the time parameters α_j as follows:

$$\left.\begin{aligned} U_j &= \alpha_{j-1} + 4\alpha_j + \alpha_{j+1} \\ hU_j' &= 3(\alpha_{j+1} - \alpha_{j-1}) \\ h^2U_j'' &= 6(\alpha_{j-1} - 2\alpha_j + \alpha_{j+1}) \end{aligned}\right\}. \tag{6}$$

Using (4) and the boundary conditions (2), we get the approximate solutions at the boundary points as

$$\left.\begin{aligned} U_0(x_0,t) &= \sum_{j=-1}^{1} \alpha_j B_j(x_0) = g_0(t) \\ U_N(x_N,t) &= \sum_{j=N-1}^{N+1} \alpha_j B_j(x_N) = g_1(t) \end{aligned}\right\}. \tag{7}$$

Using Table 1, in (7) we get

$$\left.\begin{aligned} \alpha_{-1} + 4\alpha_0 + \alpha_1 &= g_0(t) \\ \alpha_{N-1} + 4\alpha_N + \alpha_{N+1} &= g_1(t) \end{aligned}\right\}. \tag{8}$$

3. Redefined cubic B-splines collocation method

In collocation method, the basis functions should vanish on the boundary where Dirichlet's type of boundary conditions are specified, but in the set of cubic B-splines $\{B_{-1}, B_0, B_1, \ldots, B_{N-1}, B_N, B_{N+1}\}$ the basis functions $B_{-1}, B_0, B_1, \ldots, B_{N-1}, B_N, B_{N+1}$ are not vanishing at one of the boundary points. So, there is a necessity of redefining the basis functions into a new set of basis functions which vanish on the boundary where the Dirichlet's type of boundary conditions is specified.

The procedure for redefining the basis functions is as follows:

Our numerical treatment for solving equation (1) with (2) and (3) using the Cubic B-splines collocation method with redefined basis functions is to find an approximate solution $U^N(x,t)$ to the exact solution $u(x,t)$ by eliminating α_{-1}, α_{N+1} from (5) and (9); we get the approximate solution in the form [21,23]:

$$U^N(x,t) = w(x,t) + \sum_{j=0}^{N} \alpha_j \widetilde{B}_j(x), \tag{9}$$

where

$$w(x,t) = \frac{B_{-1}(x)}{B_{-1}(x_0)} g_0(t) + \frac{B_{N+1}(x)}{B_{N+1}(x_N)} g_1(t), \tag{10}$$

$$\left.\begin{aligned} \widetilde{B}_0(x) &= B_0(x) - \tfrac{B_0(x_0)}{B_{-1}(x_0)} B_{-1}(x) \\ \widetilde{B}_1(x) &= B_1(x) - \tfrac{B_1(x_0)}{B_{-1}(x_0)} B_{-1}(x) \\ \widetilde{B}_j(x) &= B_j(x), j = 2, \ldots, N-2 \\ \widetilde{B}_{N-1}(x) &= B_{N-1}(x) - \tfrac{B_{N-1}(x_N)}{B_{N+1}(x_N)} B_{N+1}(x) \\ \widetilde{B}_N(x) &= B_N(x) - \tfrac{B_N(x_N)}{B_{N+1}(x_N)} B_{N+1}(x) \end{aligned}\right\}. \tag{11}$$

Here, the new set of cubic B-splines basis functions $\widetilde{B}_j(x),\ j = 0, \ldots, N$ are redefined in such a way, so that they vanish on the boundary when Dirichlet's type of boundary conditions is specified. The function $w(x, t)$ takes care of the given boundary conditions.

To apply the proposed method with the redefined set of cubic B-splines basis functions $\widetilde{B}_j(x),\ j = 0, \ldots, N$ to the equation (1)–(3) we proceed as follows:

First, discretizing the time derivative in the usual finite difference way and applying Crank–Nicolson scheme to space derivative in (1), we get

$$\frac{u^{n+1} - u^n}{t} = -\varepsilon\left(\frac{u_x^{n+1} + u_x^n}{2}\right) + \gamma\left(\frac{u_{xx}^{n+1} + u_{xx}^n}{2}\right), \tag{12}$$

$$\Rightarrow u^{n+1} + \frac{\varepsilon\Delta t}{2} u_x^{n+1} - \frac{\gamma\Delta t}{2} u_{xx}^{n+1} = u^n - \frac{\varepsilon\Delta t}{2} u_x^n + \frac{\gamma\Delta t}{2} u_{xx}^n. \tag{13}$$

Now, using (9) in (13), we get

$$\sum_{j=0}^{N}\alpha_j^{n+1}\widetilde{B}_j(x)+\frac{\varepsilon\Delta t}{2}\sum_{j=0}^{N}\alpha_j^{n+1}\widetilde{B}_j'(x)-\frac{\gamma\Delta t}{2}\sum_{j=0}^{N}\alpha_j^{n+1}\widetilde{B}_j''(x)=\sum_{j=0}^{N}\alpha_j^{n}\widetilde{B}_j(x)-\frac{\varepsilon\Delta t}{2}\sum_{j=0}^{N}\alpha_j^{n}\widetilde{B}_j'(x)+\frac{\gamma\Delta t}{2}\sum_{j=0}^{N}\alpha_j^{n}\widetilde{B}_j''(x)$$
$$+\left\{w(x,t^n)-\frac{\varepsilon\Delta t}{2}w_x(x,t^n)+\frac{\gamma\Delta t}{2}w_{xx}(x,t^n)\right\}$$
$$-\left\{w(x,t^{n+1})+\frac{\varepsilon\Delta t}{2}w_x(x,t^{n+1})-\frac{\gamma\Delta t}{2}w_{xx}(x,t^{n+1})\right\}. \tag{14}$$

Using Table 1 with (10) and (11) in (14) we get

$$A\alpha^{n+1}=B\alpha^n+b, \tag{15}$$

where

$$A=\begin{bmatrix} y_0 & x_1 & & & \\ x & y & x & & \\ & \cdots & \cdots & \cdots & \\ & & \cdots & \cdots & \cdots \\ & & x & y & x \\ & & & x_{N-1} & y_N \end{bmatrix},\quad B=\begin{bmatrix} q_0 & p_1 & & & \\ p & q & p & & \\ & \cdots & \cdots & \cdots & \\ & & \cdots & \cdots & \cdots \\ & & p & q & p \\ & & & p_{N-1} & q_N \end{bmatrix},\quad b=\begin{bmatrix} b_0 \\ 0 \\ \cdots \\ \cdots \\ 0 \\ b_N \end{bmatrix},$$

$$\alpha^{n+1}=\begin{bmatrix} \alpha_0^{n+1} \\ \alpha_1^{n+1} \\ \cdots \\ \cdots\alpha_{N-1}^{n+1} \\ \alpha_N^{n+1} \end{bmatrix},\quad \alpha^n=\begin{bmatrix} \alpha_0^{n} \\ \alpha_1^{n} \\ \cdots \\ \cdots \\ \alpha_{N-1}^{n} \\ \alpha_N^{n} \end{bmatrix},$$

$$y_0=\left(\frac{\varepsilon\Delta t}{2}\right)\left(\frac{12}{h}\right)-\left(\frac{\gamma\Delta t}{2}\right)\left(\frac{-36}{h^2}\right),\quad x_1=\left(\frac{\varepsilon\Delta t}{2}\right)\left(\frac{6}{h}\right),$$
$$x=1+\left(\frac{\varepsilon\Delta t}{2}\right)\left(-\frac{3}{h}\right)-\left(\frac{\gamma\Delta t}{2}\right)\left(\frac{6}{h^2}\right),\quad y=4-\left(\frac{\gamma\Delta t}{2}\right)\left(\frac{-12}{h^2}\right),$$
$$z=1+\left(\frac{\varepsilon\Delta t}{2}\right)\left(\frac{3}{h}\right)-\left(\frac{\gamma\Delta t}{2}\right)\left(\frac{6}{h^2}\right),$$
$$q_0=-\left(\frac{\varepsilon\Delta t}{2}\right)\left(\frac{12}{h}\right)+\left(\frac{\gamma\Delta t}{2}\right)\left(\frac{-36}{h^2}\right),\quad p_1=-\left(\frac{\varepsilon\Delta t}{2}\right)\left(\frac{6}{h}\right),$$
$$p=1-\left(\frac{\varepsilon\Delta t}{2}\right)\left(-\frac{3}{h}\right)+\left(\frac{\gamma\Delta t}{2}\right)\left(\frac{6}{h^2}\right),\quad q=4+\left(\frac{\gamma\Delta t}{2}\right)\left(\frac{-12}{h^2}\right),$$
$$r=1-\left(\frac{\varepsilon\Delta t}{2}\right)\left(\frac{3}{h}\right)+\left(\frac{\gamma\Delta t}{2}\right)\left(\frac{6}{h^2}\right)$$
$$b_0=g_0(t^n)\left\{1-\left(\frac{\varepsilon\Delta t}{2}\right)\left(-\frac{3}{h}\right)+\left(\frac{\gamma\Delta t}{2}\right)\left(\frac{6}{h^2}\right)\right\}-g_0(t^{n+1})\left\{1+\left(\frac{\varepsilon\Delta t}{2}\right)\left(-\frac{3}{h}\right)-\left(\frac{\gamma\Delta t}{2}\right)\left(\frac{6}{h^2}\right)\right\},$$
$$b_N=g_1(t^n)\left\{1-\left(\frac{\varepsilon\Delta t}{2}\right)\left(\frac{3}{h}\right)+\left(\frac{\gamma\Delta t}{2}\right)\left(\frac{6}{h^2}\right)\right\}-g_1(t^{n+1})\left\{1+\left(\frac{\gamma\Delta t}{2}\right)\left(\frac{3}{h}\right)-\left(\frac{\gamma\Delta t}{2}\right)\left(\frac{6}{h^2}\right)\right\}.$$

Here A and B are $(N+1)\times(N+1)$ tri-diagonal matrices and b is an $(N+1)$ order column vector, which depends on the boundary conditions.

The time evolution of the approximate solution $U^N(x,t)$ is determined by the time evolution of the vector α^n. This is found by repeatedly solving the recurrence relationship once the initial vector α^0 has been computed from the initial conditions. The recurrence relationship (15) is tri-diagonal and so can be solved using Thomas algorithm.

4. Stability of the method

We have investigated stability of the proposed method by applying von-Neumann stability method.

For testing stability, we consider the equation

$$x\alpha_{j-1}^{n+1}+y\alpha_j^{n+1}+z\alpha_{j+1}^{n+1}=p\alpha_{j-1}^{n}+q\alpha_j^{n}+r\alpha_{j+1}^{n}, \tag{16}$$

where x,y,z,p,q and r are given in equation (15).

Now, we consider the trial solution (one Fourier mode out of the full solution) at a given point x_j

$$\alpha_j^n = \xi^n \exp(ij\beta h), \tag{17}$$

where $i = \sqrt{-1}$, β is the mode number and h is the element size.

Now, by substituting $\alpha_j^n = \xi^n \exp(ij\beta h)$ in (16), and simplifying the equation, we get

$$\xi = \frac{p\exp(-i\beta h) + q + r\exp(i\beta h)}{x\exp(-i\beta h) + y + z\exp(i\beta h)}. \tag{18}$$

Now, substituting the values of x,y,z,p,q and r from (16) in (19) and simplifying, we get

$$\xi = \frac{\left[2(\cos\beta h + 2) - \left(\frac{6\gamma\Delta t}{h^2}\right)(1-\cos\beta h)\right] - i\left[\left(\frac{3\varepsilon\Delta t}{h}\right)\sin\beta h\right]}{\left[2(\cos\beta h + 2) + \left(\frac{6\gamma\Delta t}{h^2}\right)(1-\cos\beta h)\right] + i\left[\left(\frac{3\varepsilon\Delta t}{h}\right)\sin\beta h\right]}, \tag{19}$$

$$\Rightarrow \xi = \frac{X_1 - iY}{X_2 + iY}, \tag{20}$$

where

$$\left.\begin{aligned} X_1 &= \left[2(\cos\beta h + 2) - \left(\tfrac{6\gamma\Delta t}{h^2}\right)(1-\cos\beta h)\right] \\ X_2 &= \left[2(\cos\beta h + 2) + \left(\tfrac{6\gamma\Delta t}{h^2}\right)(1-\cos\beta h)\right], \\ Y &= \left[\left(\tfrac{3\varepsilon\Delta t}{h}\right)\sin\beta h\right] \end{aligned}\right\}.$$

Now substitute $\lambda = \frac{\Delta t}{h^2}$, $\rho = \gamma\lambda$, $P_e = \frac{\varepsilon h}{\gamma}$ and $\mu = \cos\beta h$, where P_e is called Péclet number in (19), we get

$$\Rightarrow \xi = \frac{[2(\mu+2) - 6\rho(1-\mu)] - i[3P_e\rho\sqrt{(1-\mu^2)}]}{[2(\mu+2) + 6\rho(1-\mu)] + i[3P_e\rho\sqrt{(1-\mu^2)}]},$$

$$\Rightarrow |\xi|^2 = \frac{[2(\mu+2) - 6\rho(1-\mu)]^2 + 9P_e^2\rho^2(1-\mu^2)]}{[2(\mu+2) + 6\rho(1-\mu)]^2 + 9P_e^2\rho^2(1-\mu^2)]}. \tag{21}$$

Since numerator in (21) is less than denominator, therefore $|\xi| \leqslant 1$, hence the method is unconditionally stable. It means that there is no restriction on the grid size, i.e. on h and Δt, but we should choose them in such a way that the accuracy of the scheme is not degraded.

5. The initial vector α^0

The initial vector α^0 can be obtained from the initial condition and boundary values of the derivatives of the initial condition as the following expressions:

$$\begin{aligned} &U_x(x_j,0) = \varphi'(x_j), \quad j = 0, \\ &U(x_j,0) = \varphi(x_j), \quad j = 0,\ldots,N, \\ &U_x(x_j,0) = \varphi'(x_j), \quad i = N. \end{aligned}$$

This yields a $(N+1)\times(N+1)$ system of equations, of the form

$$A\alpha^0 = b, \tag{22}$$

where

$$A = \begin{bmatrix} 4 & 2 & & & \\ 1 & 4 & 1 & & \\ & \cdots & \cdots & \cdots & \\ & & \cdots & \cdots & \cdots \\ & & 1 & 4 & 1 \\ & & & 2 & 4 \end{bmatrix}, \quad \alpha^0 = \begin{bmatrix} \alpha_0^0 \\ \alpha_1^0 \\ \cdots \\ \cdots \\ \alpha_{N-1}^0 \\ \alpha_N^0 \end{bmatrix}, \quad \text{and } b = \begin{bmatrix} \varphi(x_0) + \left(\frac{h}{3}\right)\varphi'(x_0) \\ \varphi(x_1) \\ \cdots \\ \cdots \\ \varphi(x_{N-1}) \\ \varphi(x_N) - \left(\frac{h}{3}\right)\varphi'(x_N) \end{bmatrix}.$$

The solution of (22) can be found by Thomas algorithm.

5.1. Tri-diagonal matrix and Thomas algorithm

In (22) we obtained a very important special case of system of linear equations, in which the elements distributed on the diagonal and above and below that diagonal.

In this case instead of storing the above system in matrix form with a very large number of zero elements (especially in case of large N), we store the system in vector format: the diagonal elements in vector **b** of order $(N+1)$, the elements below

the diagonal a vector **a** also of order $(N+1)$ and the elements above the diagonal in a vector **c** of order $(N+1)$ with $a(0) = c(N) = 0$, and we apply the elimination process of one row below the diagonal.

The benefits for solving such systems using this technique (Thomas algorithm) instead of using the general elimination (Gauss elimination on a full matrix) is to reduce the number of arithmetic operations from $O[(N+1)^3]$ to $O[N+1]$. The Thomas algorithm is very efficient in computer time and storage for solving tri-diagonal system of algebraic equations.

6. Numerical experiments and discussion

In this section, we present the numerical results of present method on several problems. We tested the accuracy and stability of this method for different values of h, Δt, ε and γ.

Some important non-dimensional parameters in numerical analysis are defined as follows:

Courant number: The Courant number is defined as $C_r = \varepsilon \frac{\Delta t}{h}$.
Diffusion number: The diffusion number is defined as $S = \gamma \frac{\Delta t}{h^2}$.
Grid Péclet number: The Péclet number is defined as $P_e = \frac{C_r}{s} = \frac{\varepsilon}{\gamma} h$.

When the Péclet number is high, the convection term dominates and when the Péclet number is low the diffusion term dominates.

Problem 1. We consider the following equation [5]

$$\frac{\partial u}{\partial t} + \varepsilon \frac{\partial u}{\partial x} = \gamma \frac{\partial^2 u}{\partial x^2}, \quad 0 \leqslant x \leqslant 1, \quad 0 \leqslant t \leqslant T,$$

Table 2
Absolute errors for Problem-1.

x	Present method			Douglas Method			Ismail et al. [5]		
	$t = 1$	$t = 2$	$t = 5$	$t = 1$	$t = 2$	$t = 5$	$t = 1$	$t = 2$	$t = 5$
0.10	1.73E−07	2.29E−07	2.58E−07	1.33E−04	1.77E−04	2.00E−04	2.22E−16	2.22E−16	3.33E−16
0.50	5.24E−07	9.13E−07	1.36E−06	4.04E−04	7.02E−04	1.05E−03	8.88E−16	1.33E−15	2.44E−15
0.90	5.37E−07	8.09E−07	1.12E−06	4.15E−04	6.30E−04	8.83E−04	0.00E+00	4.44E-16	8.88E-16

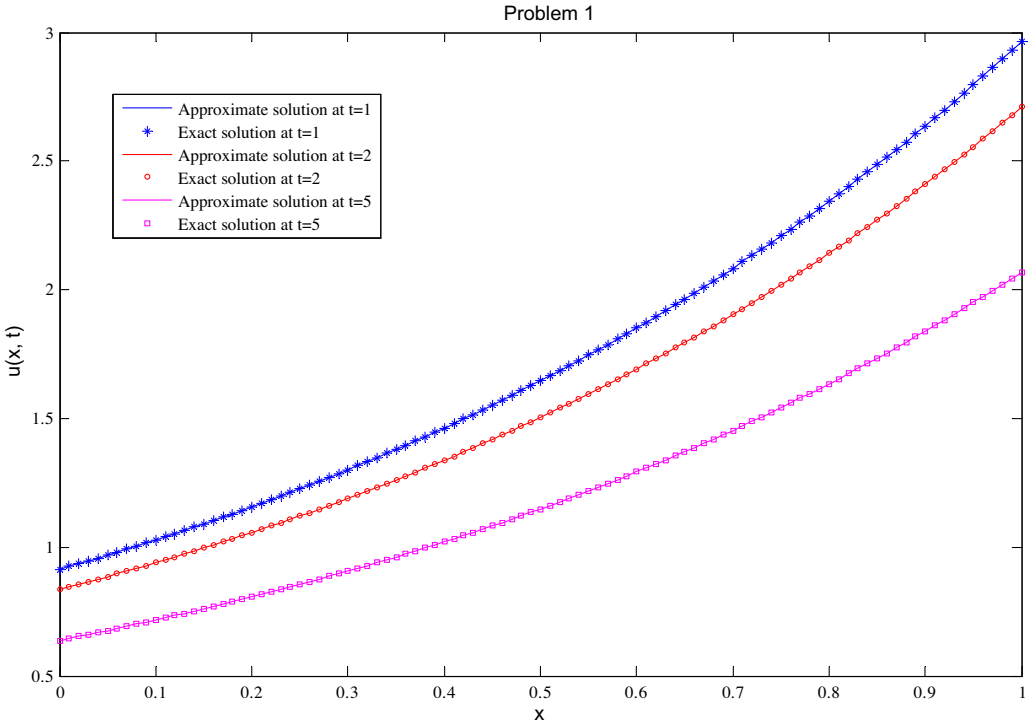

Fig. 1. Approximate and exact solution at $t = 1.0$, 2.0 and 5.0 with $P_e = 0.05$.

with $\varepsilon = 0.1$, $\gamma = 0.02$ and the following initial condition

$$\varphi(x) = \exp(\alpha x).$$

The exact solution is given by

$$u(x,t) = \exp(\alpha x + \beta t).$$

The boundary conditions can be obtained from the exact solution.

In our numerical computation, we take

$$\varepsilon = 0.1, \quad \gamma = 0.02, \quad \alpha = 1.17712434446770, \quad \beta = -0.09, \quad h = 0.01, \quad k = 0.001,$$

so that $C_r = 0.01$, $s = 0.2$, $P_e = 0.05$. The results are computed for different time levels. The absolute errors are reported in Table 2. In Fig. 1, we show the graphs between exact and numerical solutions at $t = 1$, $t = 2$ and $t = 5$. In this problem, our numerical results are more accurate in comparison to those by Douglas Method but less accurate to Ismail et al. [5] method. However, we observe the fact that our method is unconditionally stable while method in [5] is conditionally stable.

Problem 2. We consider the following equation

$$\frac{\partial u}{\partial t} + \varepsilon \frac{\partial u}{\partial x} = \gamma \frac{\partial^2 u}{\partial x^2}, \quad 0 \leqslant x \leqslant 1, \quad 0 \leqslant t \leqslant T,$$

with $\varepsilon = 3.5$, $\gamma = 0.022$ and the following initial condition

$$\varphi(x) = \exp(\alpha x).$$

Table 3
Absolute errors for Problem-2.

x	Present method			Douglas Method			Ismail et al. [5]		
	$t = 1$	$t = 2$	$t = 5$	$t = 1$	$t = 2$	$t = 5$	$t = 1$	$t = 2$	$t = 5$
0.10	2.16E−12	1.95E−12	1.45E−12	2.56E−07	2.37E−07	5.63E−07	2.56E−10	2.38E−10	5.65E−10
0.50	1.09E−11	9.88E−12	7.32E−12	8.37E−07	1.38E−06	1.90E−06	8.39E−10	1.38E−409	1.91E−09
0.90	1.99E−11	1.80E−11	1.33E−11	1.33E−06	2.82E−06	3.95E−06	1.33E−09	2.83E-09	3.97E-09

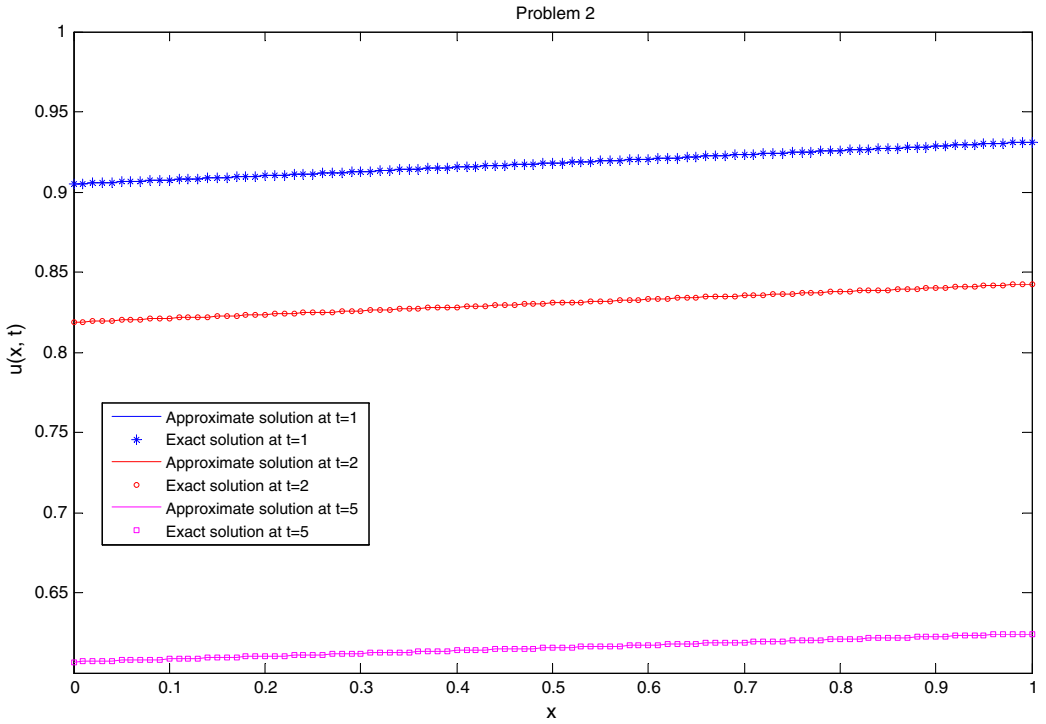

Fig. 2. Approximate and exact solution at $t = 1.0$, 2.0 and 5.0 with $P_e = 1.5909$.

The exact solution is given by

$$u(x,t) = \exp(\alpha x + \beta t).$$

The boundary conditions can be obtained from the exact solution.

In our numerical computation, we take

$$\varepsilon = 3.5, \quad \gamma = 0.022, \quad \alpha = 0.02854797991928, \quad \beta = -0.0999, \quad h = 0.01, \quad k = 0.001,$$

so that $C_r = 0.35$, $s = 0.22$, $P_e = 1.5909$. The results are computed for different time levels. The absolute errors are reported in Table 3. The exact and numerical solutions at $t = 1$, $t = 2$ and $t = 5$ are depicted in Fig. 2. In this problem, our numerical results are more accurate in comparison to shown methods. Here, we observe that for higher value of P_e our method produces more accurate results in comparison to other shown method.

Problem 3. We consider the following equation [9]

$$\frac{\partial u}{\partial t} + \varepsilon \frac{\partial u}{\partial x} = \gamma \frac{\partial^2 u}{\partial x^2}, \quad 0 \leqslant x \leqslant 1, \quad 0 \leqslant t \leqslant T,$$

with the following initial condition $\varphi(x) = \exp\left(-\frac{(x-2)^2}{80\gamma}\right)$.

Table 4
Absolute errors for Problem-3.

x	Present method		Dehghan [9]	
	$t = 1$	$t = 2$	$t = 5$	$t = 1$
0.10	9.96E−09	8.45E−09	9.06E−10	3.40E−05
0.20	1.91E−08	1.76E−08	1.54E−09	3.20E−05
0.30	2.70E−08	2.71E−08	1.84E−09	3.10E−05
0.40	3.33E−08	3.67E−08	1.77E−09	2.90E−05
0.50	3.78E−08	4.59E−08	1.33E−09	2.70E−05
0.60	4.02E−08	5.38E−08	5.59E−10	2.70E−05
0.70	3.99E−08	5.89E−08	3.83E−10	2.50E−05
0.80	3.60E−08	5.79E−08	1.18E−09	2.20E−05
0.90	2.55E−08	4.36E−08	1.31E−09	2.00E−05

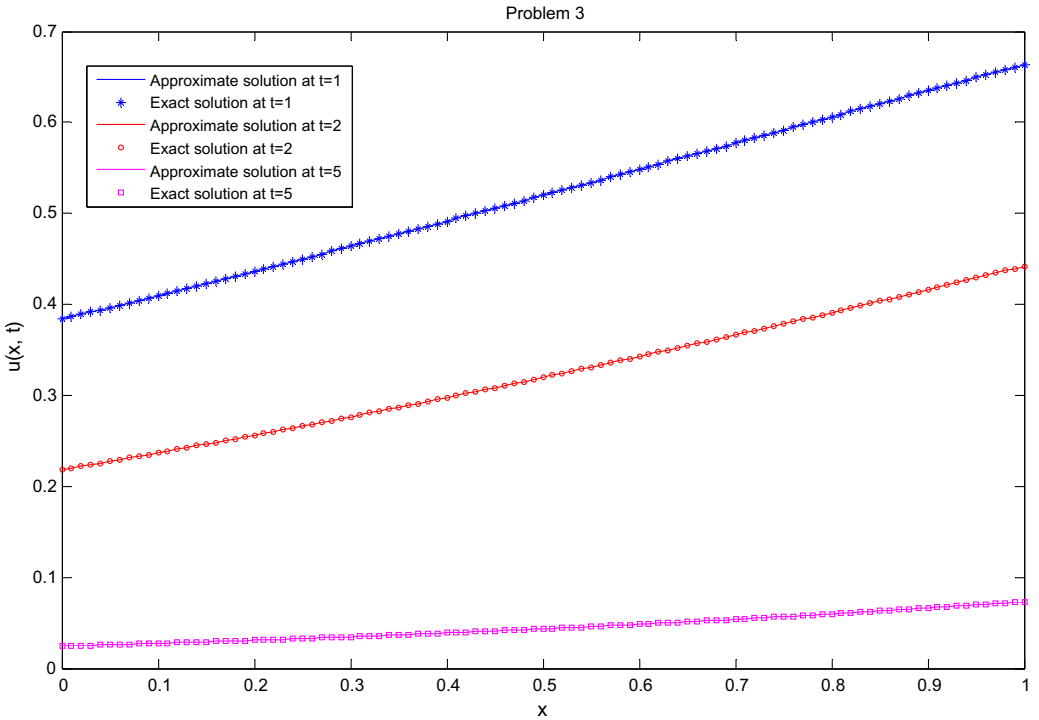

Fig. 3. Approximate and exact solution at $t = 1.0$, 2.0 and 5.0 with $P_e = 0.08$.

The exact solution is given by

$$u(x,t) = \left(\sqrt{\left(\frac{20}{20+t}\right)}\right) exp\left(-\frac{(x-2-\varepsilon t)^2}{4\gamma(t+20)}\right).$$

The boundary conditions can be obtained from the exact solution.

In our numerical computation, we take $\varepsilon = 0.8$, $\gamma = 0.1$, $h = 0.01$, $k = 0.001$.

So that $C_r = 0.08$, $s = 1.0$, $P_e = 0.08$.

The results are computed for different time levels. The absolute errors are reported in Table 4. In Fig. 3, we show the graphs between exact and numerical solutions at t = 1, t = 2 and t = 5. In this problem, our numerical results are more accurate in comparison to Dehghan [9] at t = 1. Approximate and exact solutions are also drawn for different values Pe = 0.4, 1.0, 2.0 and 4.0 in Fig. 4. We observe that as we increase the value of Pe, the results tend to zero.

Problem 4. We consider the following equation [10–12]

$$\frac{\partial u}{\partial t} + \varepsilon \frac{\partial u}{\partial x} = \gamma \frac{\partial^2 u}{\partial x^2}, \quad 0 \leqslant x \leqslant L, \quad 0 \leqslant t \leqslant T.$$

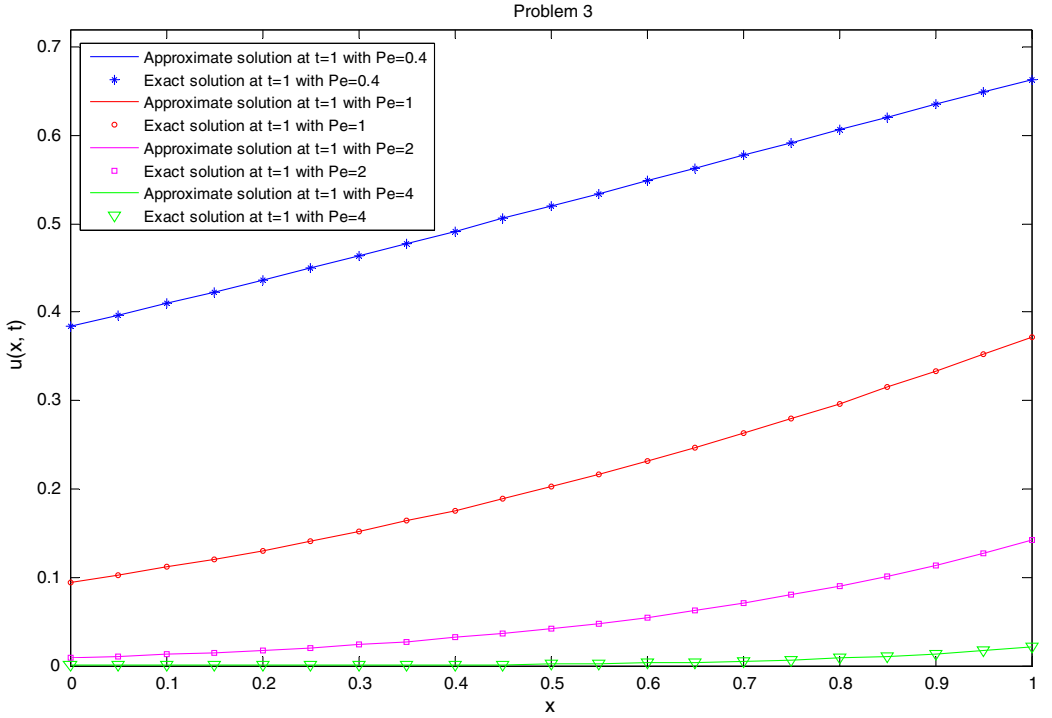

Fig. 4. Approximate and exact solution with $P_e = 0.4$, 1.0, 2.0 and 4.0 at $t = 1.0$ ($\varepsilon = 0.8$, $h = 0.05$ and $k = 0.01$).

Table 5
Absolute errors for Problem-4 (with $h = 0.01$, $k = 0.001$).

Present method		
x	$t = 1$	$t = 2$
0.10	1.06E−06	1.01E−13
0.20	5.02E−06	3.65E−12
0.30	1.82E−05	7.56E−11
0.40	1.08E−05	1.06E−09
0.50	4.63E−05	1.04E−08
0.60	4.17E−06	7.27E−08
0.70	3.78E−05	3.53E−07
0.80	7.10E−06	1.14E−06
0.90	8.98E−06	2.12E−06

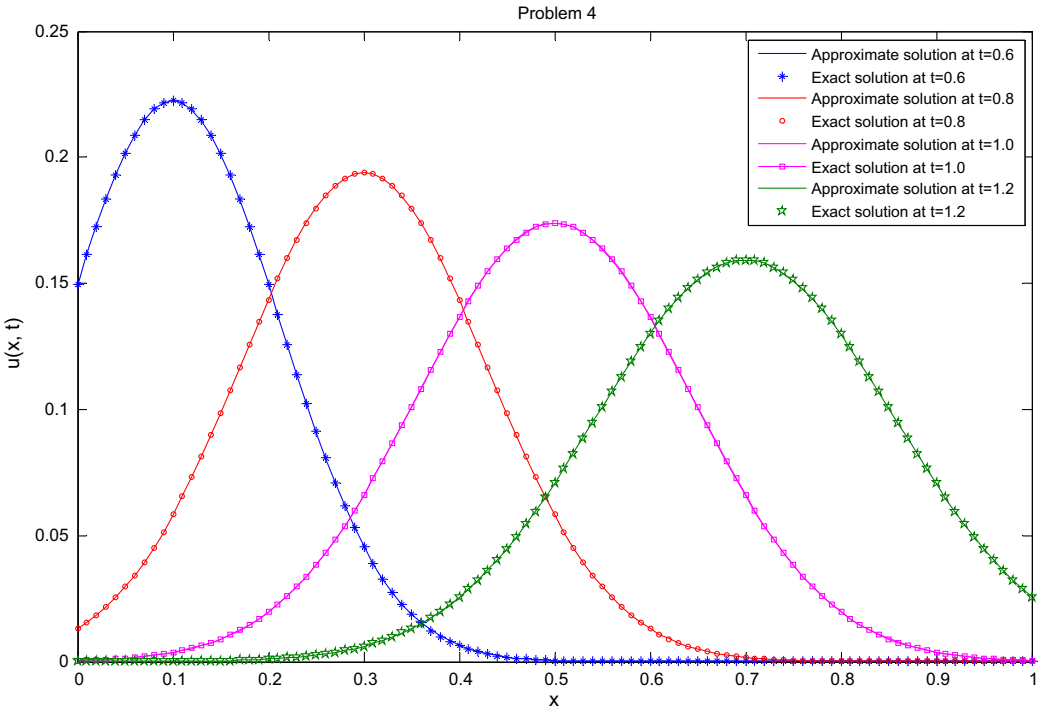

Fig. 5. Approximate and exact solution at $t = 0.6$, 0.8, 1.0 and 1.2 with $P_e = 1.0$ ($\varepsilon = 1.0$, $\gamma = 0.01$, $h = 0.01$ and $k = 0.01$).

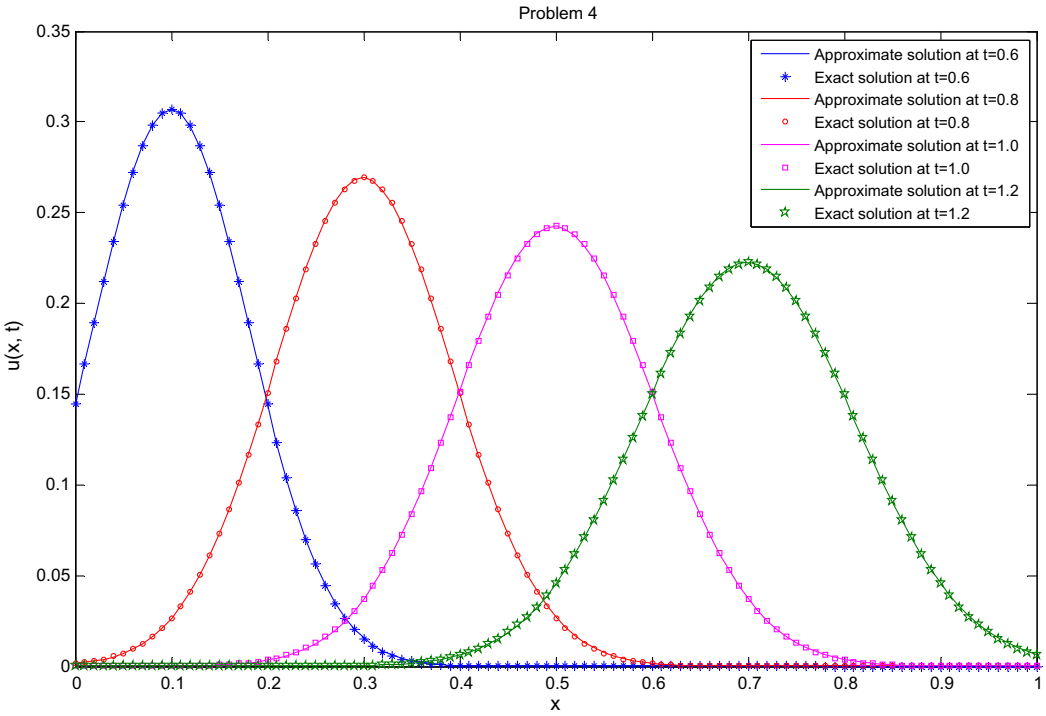

Fig. 6. Approximate and exact solution at $t = 0.6$, 0.8, 1.0 and 1.2 with $P_e = 2.0$ ($\varepsilon = 1.0$, $\gamma = 0.005$, $h = 0.01$ and $k = 0.01$).

The exact solution is given by

$$u(x,t) = \left(\frac{\sigma_0}{\sigma}\right) exp\left(-\frac{(x - x_0 - \varepsilon t)^2}{2\sigma^2}\right),$$

where $\sigma^2 = \sigma_0^2 + 2\gamma t$ and the initial condition $\varphi(x) = \exp\left(-\frac{(x-x_0)^2}{2\sigma_0^2}\right)$.

The boundary conditions can be obtained from the exact solution.

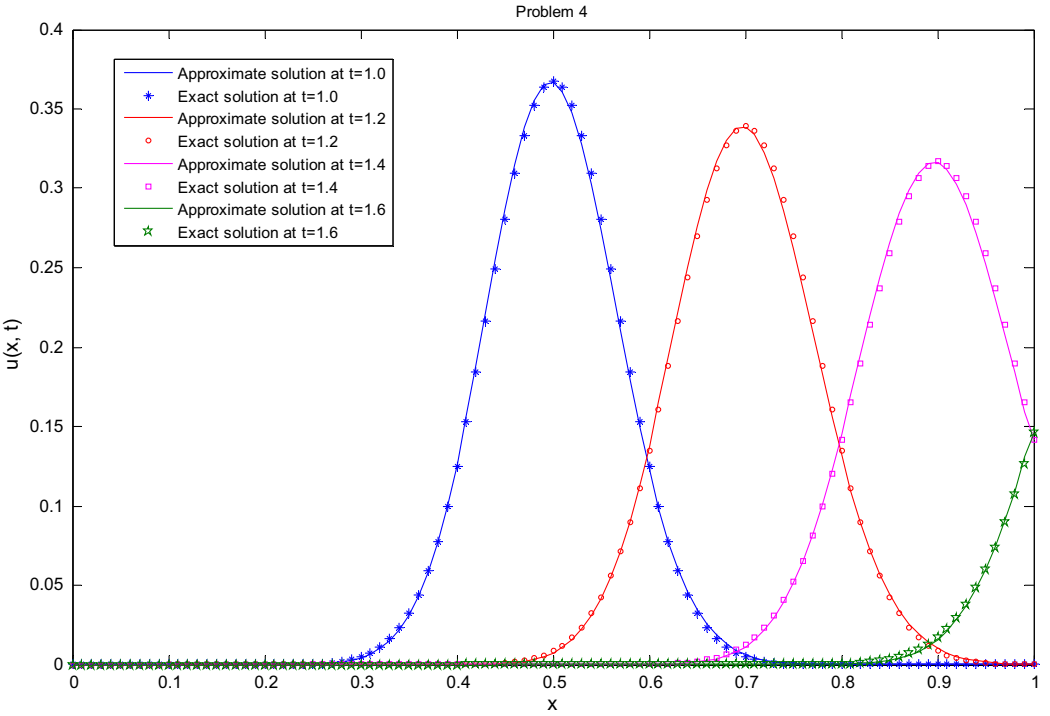

Fig. 7. Approximate and exact solution at $t = 1.0$, 1.2, 1.4 and 1.6 with $P_e = 5.0$ ($\varepsilon = 1.0$, $\gamma = 0.002$, $h = 0.01$ and $k = 0.01$).

Table 6
Approximate and exact Solutions for Problem-4 (with $h = 0.01$, $k = 0.01$ and $t = 1.0$).

x	Present Method	Kadalbajoo and Arora [10]	Exact
0.1	0.0035861	0.0035860	0.0035992
0.2	0.0196139	0.0196139	0.0196423
0.3	0.0660911	0.0660910	0.0660099
0.4	0.1368574	0.1368574	0.1366028
0.5	0.1740807	0.1740807	0.1740777
0.6	0.1362195	0.1362195	0.1366028
0.7	0.0658154	0.0658153	0.0660099
0.8	0.0197572	0.0197571	0.0196423
0.9	0.0037190	0.0037190	0.0035992

Table 7
Absolute errors for Problem-4.

x	Present method					
	$t = 1$	$t = 2$	$t = 3$	$t = 4$	$t = 5$	$t = 6$
0.5	9.02E−06	2.71E−07	2.36E−09	1.29E−11	5.30E−14	0.00E+00
1.0	4.62E−05	2.01E−05	1.18E−06	1.99E−08	1.79E−10	1.10E−12
1.5	1.04E−04	8.12E−05	2.61E−05	2.93E−06	8.31E−08	1.17E−09
2.0	4.53E−05	1.21E−04	9.53E−05	2.68E−05	5.30E−06	2.34E−07
2.5	3.28E−06	5.42E−05	1.14E−04	9.95E−05	2.35E−05	7.92E−06
3.0	1.70E−08	4.08E−06	4.38E−05	1.04E−04	9.75E−05	1.79E−05
3.5	1.55E−12	1.58E−07	2.25E−07	2.88E−05	9.35E−05	9.26E−05

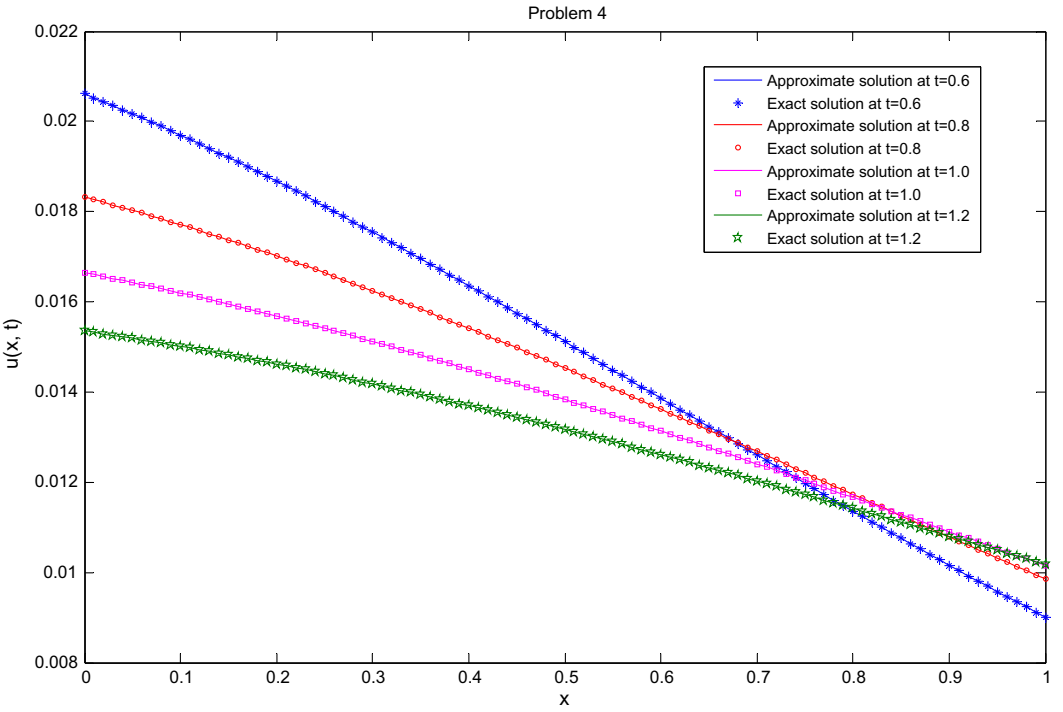

Fig. 8. Approximate and exact solution at $t = 0.6, 0.8, 1.0$ and 1.2 with $P_e = 0.0001$ ($\varepsilon = 0.01$, $\gamma = 1.0$, $h = 0.01$ and $k = 0.01$).

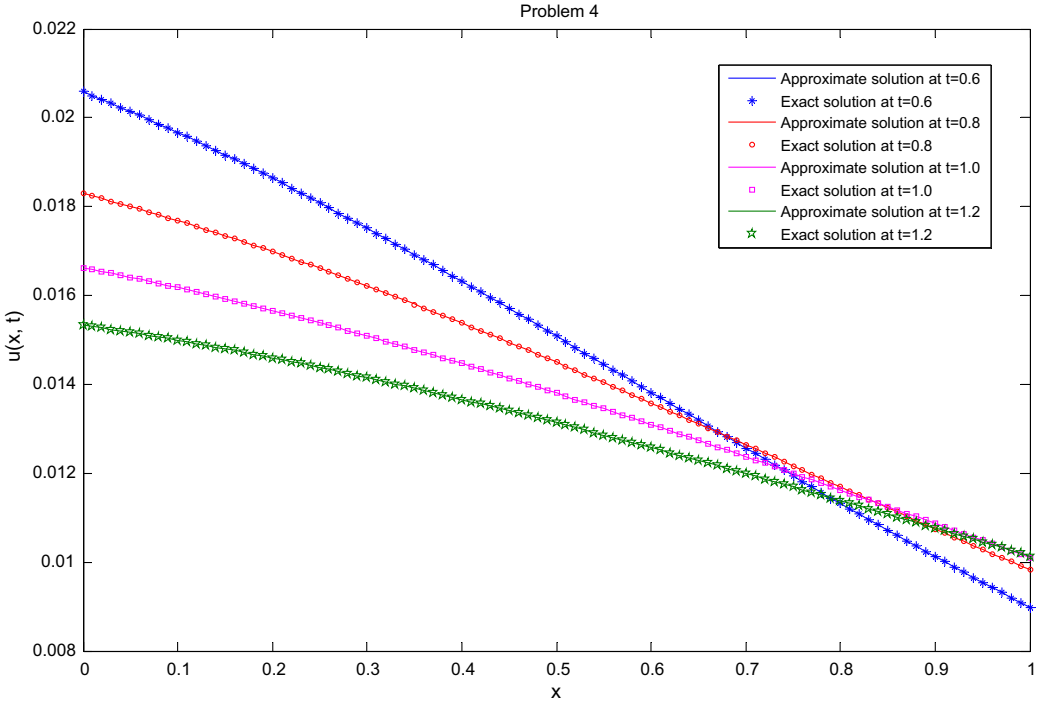

Fig. 9. Approximate and exact solution at $t = 0.6, 0.8, 1.0$ and 1.2 with $P_e = 0.00005$ ($\varepsilon = 0.005$, $\gamma = 1.0$, $h = 0.01$ and $k = 0.01$).

First, we take $L = 1$, $\sigma_0 = 0.025$, $x_0 = -0.5$, $\varepsilon = 1.0$, $\gamma = 0.01$ as given in [11], and then the exact solution is

$$u(x,t) = \left(\frac{0.025}{\sqrt{(0.000625 + 0.02t)}}\right)\exp\left(-\frac{(x+0.5-t)^2}{(0.00125+0.04t)}\right)$$

In our computation, we take $h = 0.01$, $k = 0.001$, so that $C_r = 0.1$, $s = 0.1$, $P_e = 1.0$. The results are computed for different time levels. The absolute errors for $t = 1$ and $t = 2$ are reported in Table 5. The exact and numerical solutions are also depicted at $t = 0.6$, 0.8, 1.0 and 1.2 with $h = 0.01$, $k = 0.01$, $\varepsilon = 1.0$, $\gamma = 0.01$ and $P_e = 1.0$ in Fig. 5 and with $h = 0.01$, $k = 0.01$, $\varepsilon = 1.0$, $\gamma = 0.005$ and $P_e = 2.0$ in Fig. 6. We also show the graphs at $t = 1.0$, $t = 1.2$, $t = 1.4$ and $t = 1.6$ with $h = 0.01$, $k = 0.01$, $\varepsilon = 1.0$, $\gamma = 0.002$ and $P_e = 5.0$ in Fig. 7. It is observed that our graphs are similar to the graphs shown in Sari et at. [11]. Results are also compared in Table 6 with those given in Kadalbajoo and Arora [10] where $h = 0.01$, $k = 0.01$ and $t = 1$, so that $C_r = 1.0$, $s = 1.0$, $P_e = 1.0$ and found that our results are compatible with [10]. We also computed numerical solution at $t = 0.4$, 0.6, 08, 1.0 and 1.2 with $\varepsilon = 0.01$, $\gamma = 1.0$, $h = 0.01$ and $k = 0.01$ so that $P_e = 0.0001$ and recorded in Table 7. We also depicted numerical and exact solutions at $P_e = 0.0001$ and 0.00005 in Figs. 8 and 9 respectively.

Table 8
Absolute errors for Problem-4.

x	Present method				
	$t = 0.4$	$t = 0.6$	$t = 0.8$	$t = 1.0$	$t = 1.2$
0.1	1.35E−07	8.49E−08	3.55E−08	1.11E−08	6.49E−10
0.2	4.38E−07	1.49E−07	3.78E−08	6.89E−09	7.66E−10
0.3	6.17E−07	2.14E−07	5.57E−08	1.17E−08	3.45E−10
0.4	7.30E−07	2.62E−07	7.03E−08	1.62E−08	1.78E−09
0.5	7.71E−07	2.86E−07	7.90E−08	1.95E−08	3.22E−09
0.6	7.38E−07	2.83E−07	8.01E−08	2.10E−08	4.31E−09
0.7	6.34E−07	2.51E−07	7.27E−08	1.99E−08	4.74E−09
0.8	4.68E−07	1.90E−07	5.64E−08	1.61E−08	4.26E−09
0.9	2.52E−07	1.05E−07	3.18E−08	9.45E−09	2.71E−09

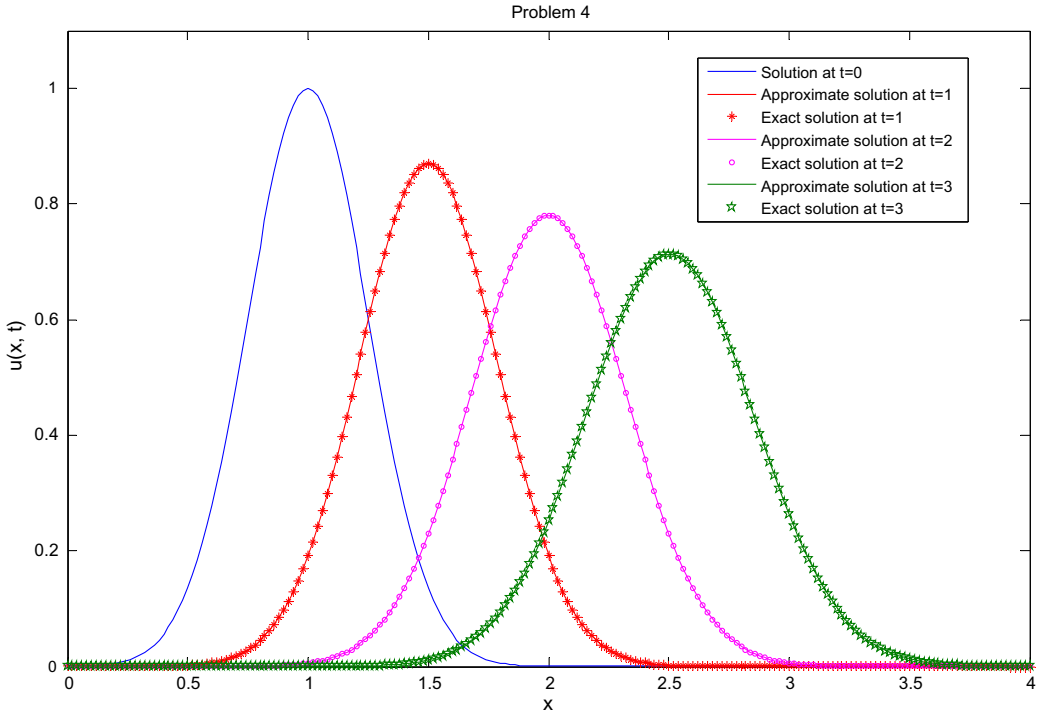

Fig. 10. Approximate and exact solution at $t = 0.0$, 1.0, 2.0 and 3.0 with $P_e = 1.0$ ($\varepsilon = 0.5$, $\gamma = 0.01$, $h = 0.02$ and $k = 0.01$).

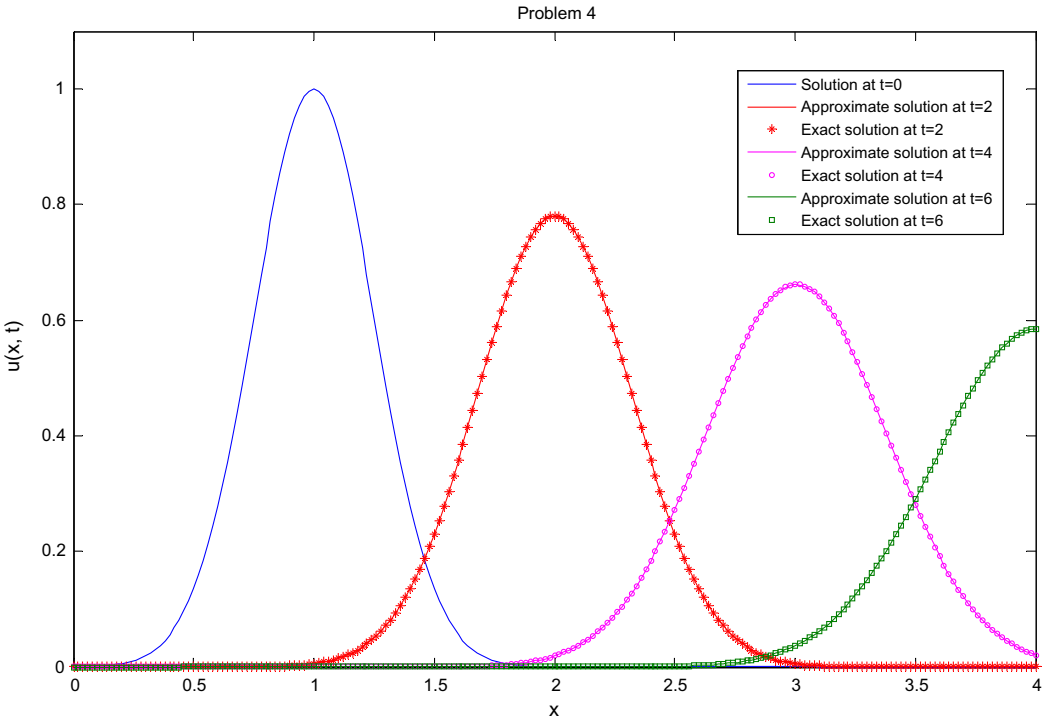

Fig. 11. Approximate and exact solution at $t = 0.0, 2.0, 4.0$ and 6.0 with $P_e = 1.0$ ($\varepsilon = 0.5$, $\gamma = 0.01$, $h = 0.02$ and $k = 0.01$).

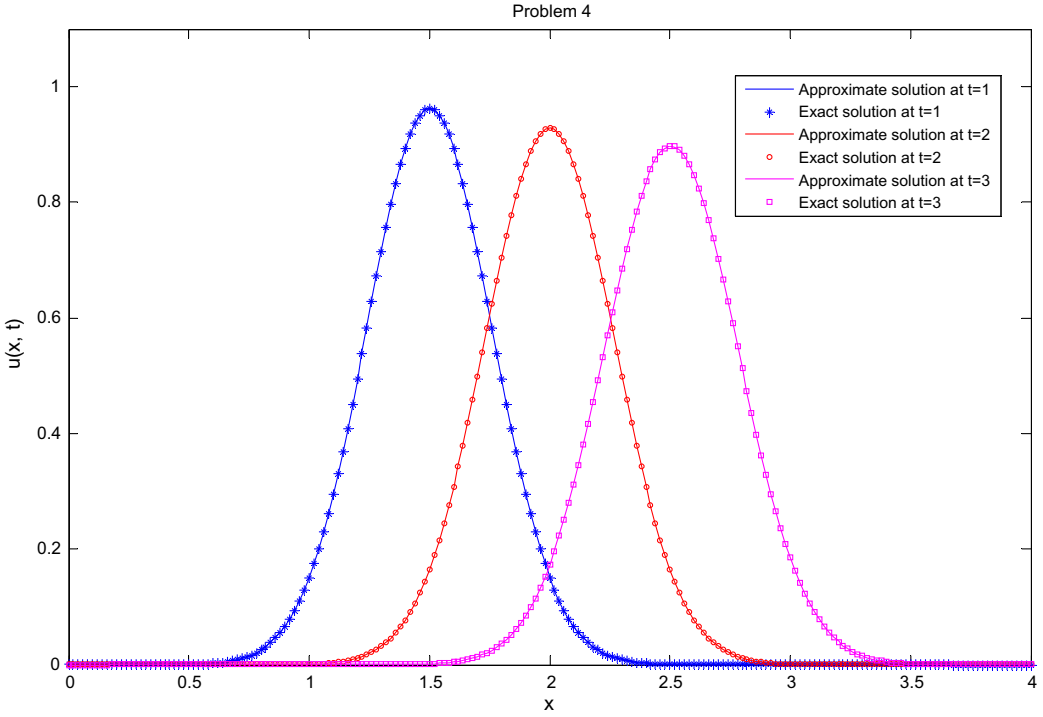

Fig. 12. Approximate and exact solution at $t = 1.0, 2.0$ and 3.0 with $P_e = 4.0$ ($\varepsilon = 0.5$, $\gamma = 0.0025$, $h = 0.02$ and $k = 0.01$).

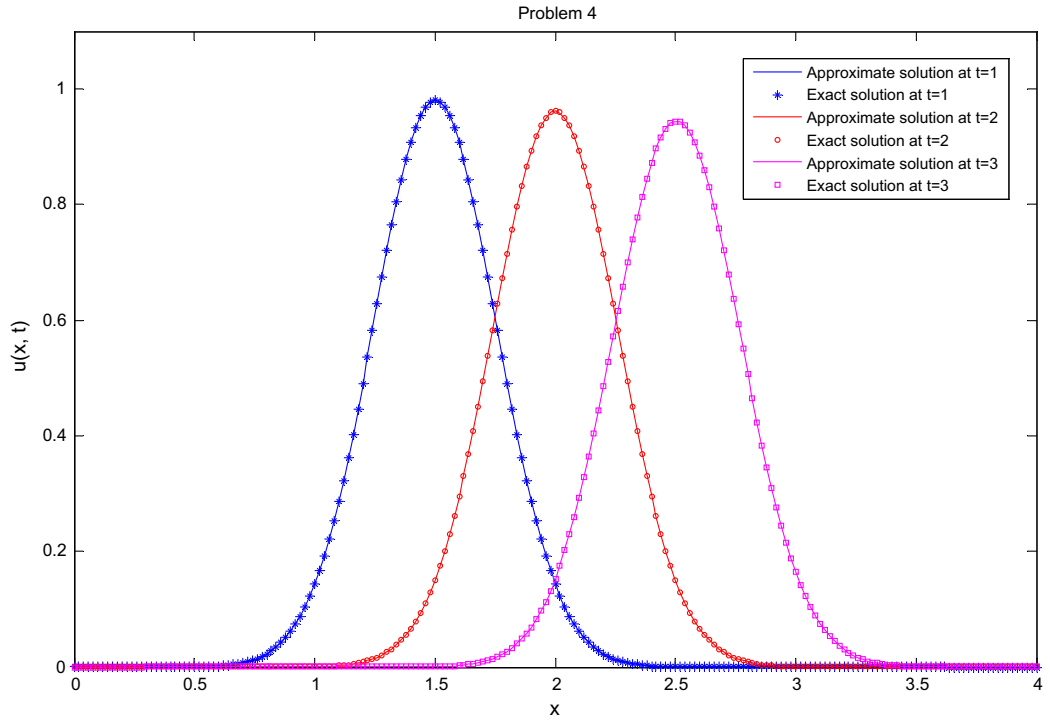

Fig. 13. Approximate and exact solution at $t = 1.0$, 2.0 and 3.0 with $P_e = 8.0$ ($\varepsilon = 0.5$, $\gamma = 0.00125$, $h = 0.02$ and $k = 0.01$).

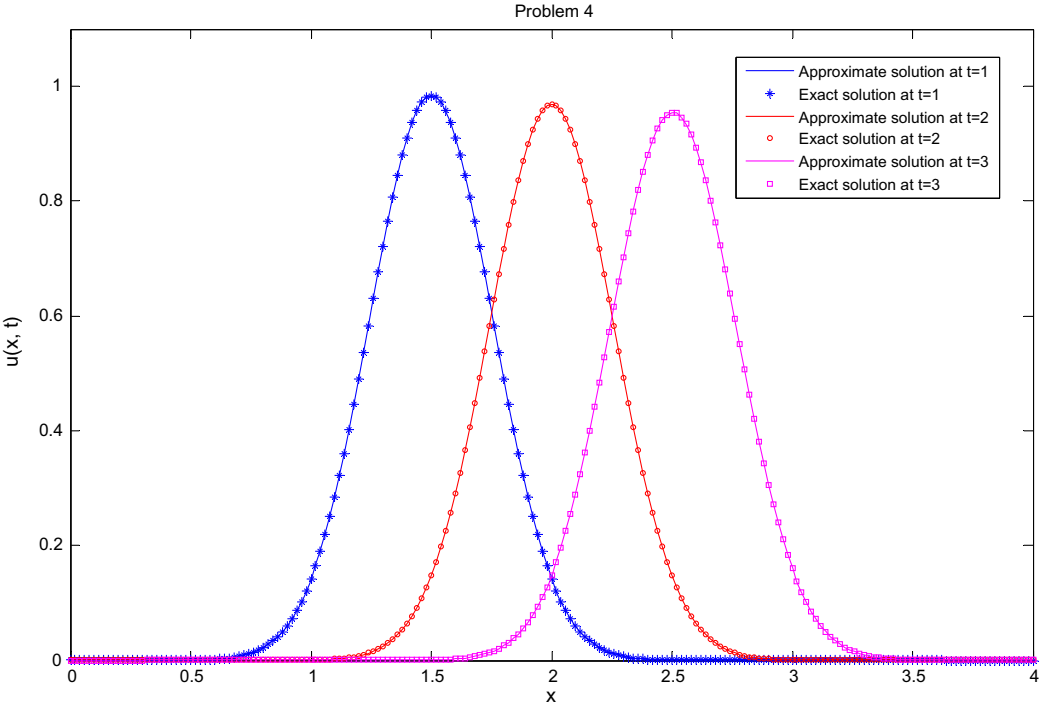

Fig. 14. Approximate and exact solution at $t = 1.0$, 2.0 and 3.0 with $P_e = 10.0$ ($\varepsilon = 0.5$, $\gamma = 0.001$, $h = 0.02$ and $k = 0.01$).

We also take $L = 4$, $\sigma_0 = 0.25$, $x_0 = 1.0$, $\varepsilon = 0.5$, $\gamma = 0.01$ so that exact solution becomes

$$u(x,t) = \left(\frac{0.25}{\sqrt{(0.0625 + 0.02t)}}\right)\exp\left(-\frac{(x - 1.0 - 0.5t)^2}{(0.125 + 0.04t)}\right).$$

In our computation, we take $h = 0.02$, $k = 0.01$, so that $C_r = 0.25$, $s = 0.25$, $P_e = 1.0$. The results are computed for different time levels. The absolute errors are reported in Table 8. We show the graphs between exact and numerical solutions at $t = 0.0$, 1.0, 2.0 and 3.0 with $h = 0.02$, $P_e = 1.0$ in Fig. 10 and at $t = 0.0$, 2.0, 4.0 and 6.0 with $h = 0.02$, $P_e = 1.0$ in Fig. 11. We also show the graphs for $h = 0.02$, $P_e = 4.0$, 8.0 and 10.0 in Figs. 12–14. It is observed that our Fig. 10 is exactly similar to the graphs shown in Rizwan-Uddin [12].

Problem 5. We consider the following equation

$$\frac{\partial u}{\partial t} + \varepsilon\frac{\partial u}{\partial x} = \gamma\frac{\partial^2 u}{\partial x^2}, \quad 0 \leqslant x \leqslant 1, \quad 0 \leqslant t \leqslant T,$$

with $\varepsilon = 1.0$, $\gamma = 1.0$ and the following initial condition

$$\varphi(x) = \frac{1}{\sqrt{s}}exp\left(-50\frac{x^2}{s}\right), \quad s(\text{diffusion number}) = 1.0.$$

The exact solution is given by

$$u(x,t) = \left(\frac{1}{\sqrt{s}}\right)exp\left(-50\frac{(x-t)^2}{s}\right), \quad s = (1 + 200\gamma t).$$

Table 9
Maximum absolute errors for Problem-5.

Present method			Chawla et al. [8]
$t = 1$	$t = 2$	$t = 5$	$t = 1$
1.27E−06	3.88E−08	1.45E−08	1.80E−05

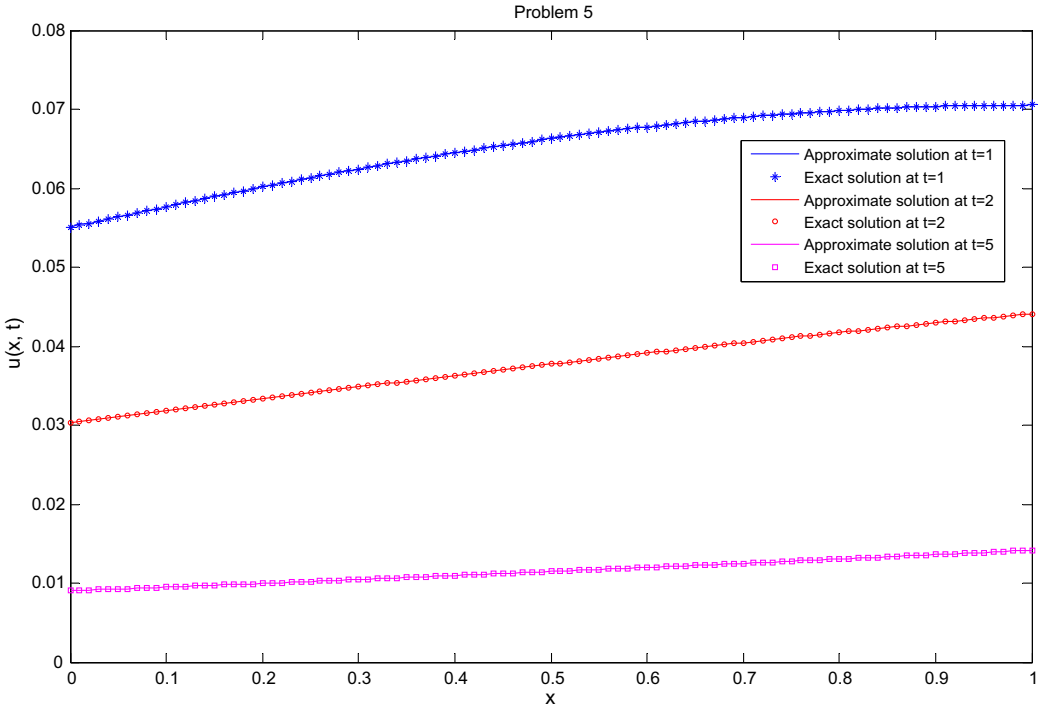

Fig. 15. Approximate and exact solution at $t = 1.0$, 2.0 and 5.0 with $P_e = 0.01$ ($\varepsilon = 1.0$, $\gamma = 1.0$, $h = 0.01$ and $k = 0.01$).

The boundary conditions can be obtained from the exact solution.

In our computation, we take $\varepsilon = 1.0$, $\gamma = 1.0$, $h = 0.05$, $k = 0.01$, so that $C_r = 0.2$, $s = 4.0$, $P_e = 0.05$. The results are computed for different time levels. The maximum absolute errors are reported in Table 9. In Fig. 15, we show the graphs between exact and numerical solutions at $t = 1.0$, 2.0 and 5.0 with $h = 0.01$. In this problem, our numerical results are found to be more accurate in comparison to those of Chawla et al. [8].

Problem 6. We consider the following equation

$$\frac{\partial u}{\partial t} + \varepsilon \frac{\partial u}{\partial x} = \gamma \frac{\partial^2 u}{\partial x^2}, \quad 0 \leqslant x \leqslant 1, \quad 0 \leqslant t \leqslant T,$$

with $\varepsilon = 1.0$, $\gamma = 0.1$ and the following initial condition

$$\varphi(x) = \exp(5x)\left[cos\left(\frac{\pi}{2}x\right) + 0.25sin\left(\frac{\pi}{2}x\right)\right].$$

The exact solution is given by

$$u(x,t) = \exp\left(5\left(x - \frac{t}{2}\right)\right) exp\left(-\frac{\pi^2}{40}t\right) \exp(5x)\left[cos\left(\frac{\pi}{2}x\right) + 0.25sin\left(\frac{\pi}{2}x\right)\right].$$

Table 10
Absolute errors for Problem-6.

x	Present method				Mohebbi and Dehghan [1]
	$h = 0.01$, $k = h$			$h = 0.01$, $k = 2$ h	
	$t = 1$	$t = 2$	$t = 5$	$t = 2$	
0.10	8.16E−06	5.60E−07	1.49E−10	2.06E−06	1.80E−06
0.20	1.89E−05	1.32E−06	3.57E−10	5.64E−06	2.77E−06
0.30	2.95E−05	2.13E−06	5.87E−10	1.13E−05	4.17E−06
0.40	3.39E−05	2.59E−06	7.43E−10	1.94E−05	6.17E−06
0.50	2.05E−05	1.97E−06	6.40E−10	3.00E−05	9.00E−06
0.60	2.71E−05	8.31E−07	1.10E−12	4.20E−05	1.30E−05
0.70	1.24E−04	6.83E−06	1.46E−09	5.25E−05	1.84E−05
0.80	2.60E−04	1.55E−05	3.61E−09	5.56E−05	2.55E−05
0.90	3.35E−04	2.05E−05	4.95E−09	4.20E−05	3.41E−05

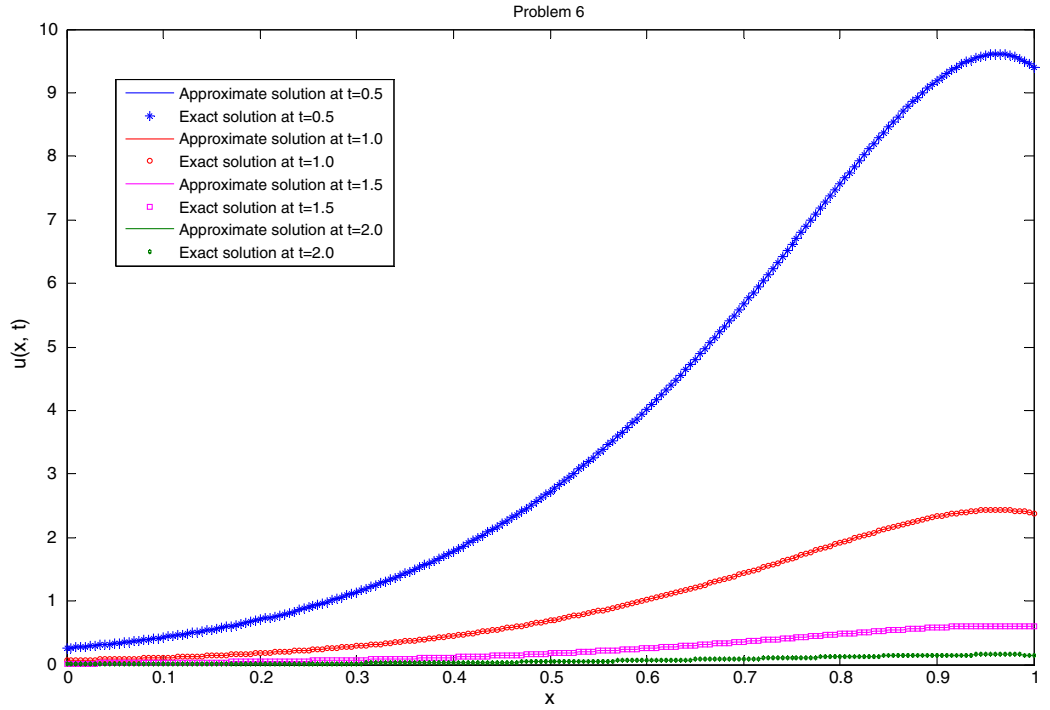

Fig. 16. Approximate and exact solution at $t = 0.5$, 1.0, 1.5 and 2.0 with $P_e = 0.05$ ($\varepsilon = 1.0$, $\gamma = 0.1$, $h = 0.005$ and $k = 0.01$).

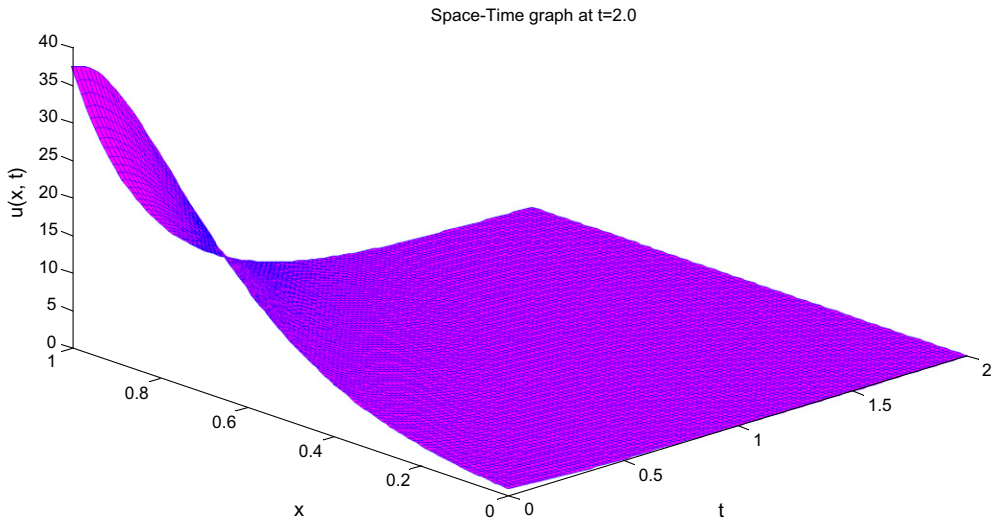

Fig. 17. Space-time graph of approximate solution with $h = 0.01$.

The boundary conditions can be obtained from the exact solution.

In our computation, we take $\varepsilon = 1.0$, $\gamma = 0.1$, $h = 0.01$, $k = h$, so that $C_r = 1.0$, $s = 10.0$, $P_e = 0.1$. The results are computed for different time levels. The absolute errors are reported in Table 10. In Fig. 16, the graphs between exact and numerical solutions for $t = 0.5$, 1.0, 1.5 and 2.0 are depicted. We also compute results with $k = 2$ h, so that $C_r = 2.0$, $s = 20.0$, $P_e = 0.1$. In this problem, for $h = 0.01$, $k = h$ and $t = 2$, our results are found to be more accurate in comparison to those given by Mohebbi and Dehghan [1]. However, for $k = 2$ h, our results are not that accurate. We also show the graph in Fig. 17, between space and time at $t = 2.0$ and it is similar to graph of Mohebbi and Dehghan [1].

7. Conclusions

In this work, the convection–diffusion equation was dealt by using collocation method with redefined cubic B-splines basis functions. The method successfully worked to give very accurate solutions for values of P_e upto 10. The performance of the method for the considered problems was measured by calculating the absolute errors for different time levels which are reported in tabular form. Moreover, approximate and exact solutions for different values of P_e at different time levels are shown graphically. The method gives convergent approximations and handles the given equations very well. Comparisons of the results with exact solutions showed that the present method is capable of solving the given equations very well. Easy and economical implementation process is the strength of the presented method. The presented method is unconditionally stable and efficient; hence it can be used as an alternative method for modeling the behavior of the similar problems.

Acknowledgment

One of the authors R.K. Jain thankfully acknowledges the sponsorship under QIP, provided by Technical Education and Training Department, Bhopal (M.P.), India. The authors are very thankful to the reviewers for their valuable suggestions to improve the quality of the paper.

References

[1] A. Mohebbi, M. Dehghan, High-order compact solution of the one-dimensional heat and advection–diffusion equations, Appl. Math. Model. 34 (2010) 3071–3084.
[2] D.K. Salkuyeh, On the finite difference approximation to the convection–diffusion equation, Appl. Math. Comput. 179 (2006) 79–86.
[3] H. Karahan, Unconditional stable explicit finite difference technique for the advection–diffusion equation using spreadsheets, Adv. Eng. Software 38 (2007) 80–86.
[4] H. Karahan, Implicit finite difference techniques for the advection–diffusion equation using spreadsheets, Adv. Eng. Software 37 (2006) 601–608.
[5] H.N.A. Ismail, E.M.E. Elbarbary, G.S.E. Salem, Restrictive Taylor's approximation for solving convection–diffusion equation, Appl. Math. Comput. 147 (2004) 355–363.
[6] H.F. Ding, Y.X. Zhang, A new difference scheme with high accuracy and absolute stability for solving convection–diffusion equations, J. Comput. Appl. Math. 230 (2009) 600–606.
[7] M.M. Chawla, M.A. Al-Zanaidi, D.J. Evans, Generalized trapezoidal formulas for convection–diffusion equations, Int. J. Comput. Math. 72 (1999) 141–154.
[8] M.M. Chawla, M.A. Al-Zanaidi, M.G. Al-Aslab, Extended one step time-integration schemes for convection–diffusion equations, Comput. Math. Appl. 39 (2000) 71–84.
[9] M. Dehghan, Weighted finite difference techniques for the one-dimensional advection–diffusion equation, Appl. Math. Comput. 147 (2004) 307–319.

[10] M.K. Kadalbajoo, P. Arora, Taylor-Galerkin B-spline finite element method for the one dimensional advection–diffusion equation, Numer. Methods Partial Diff. Eqs. 26 (5) (2009). 2006-1223.
[11] M. Sari, G. Güraslan, A. Zeytinoğlu, High-Order finite difference schemes for solving the advection–diffusion equation, Math. Comput. Appl. 15 (3) (2010) 449–460.
[12] Rizwan-Uddin, A second-order space and time nodal method for the one-dimensional convection–diffusion equation, Comput. Fluids 26 (3) (1997) 233–247.
[13] S. Karaa, J. Zhang, High order ADI method for solving unsteady convection–diffusion problems, J. Comput. Phys. 198 (2004) 1–9.
[14] M. Dehghan, Numerical solution of the three-dimensional advection–diffusion equation, Appl. Math. Comput. 150 (2004) 5–19.
[15] M. Dehghan, A. Mohebbi, High-order compact boundary value method for the solution of unsteady convection–diffusion problems, Math. Comput. Simul. 79 (2008) 683–699.
[16] M. Dehghan, On the numerical solution of the one-dimensional convection–diffusion equation, Math. Problems Eng. 2005 (1) (2005) 61–74.
[17] M. Dehghan, Quasi-implicit and two-level explicit finite-difference procedures for solving the one-dimensional advection equation, Appl. Math. Comput. 167 (2005) 46–67.
[18] M. Dehghan, Finite difference procedures for solving a problem arising in modeling and design of certain optoelectronic devices, Math. Comput. Simul. 71 (2006) 16–30.
[19] M. Dehghan, Time-splitting procedures for the solution of the two-dimensional transport equation, Kybernetes 36 (5/6) (2007) 791–805.
[20] M. Dehghan, F. Shakeri, Application of He's variational iteration method for solving the Cauchy reaction–diffusion problem, J. Comput. Appl. Math. 214 (2008) 435–446.
[21] J.H. Ahlberg, T. Ito, A collocation method for two point Boundary value Problems, Math. Comput. 29 (131) (1975) 761–776.
[22] R.C. Mittal, Geeta Arora, Numerical solution of the coupled viscous Burgers' equation, Commun. Nonlinear Sci. Numer. Simul. 16 (2011) 1304–1313.
[23] K.N.S. Vishwanadham Kasi, S.R. Koneru, Finite element method for one dimensional and two-dimensional time dependent problems with B-splines, Comput. Methods Appl. Mech. Eng. 108 (1993) 201–222.