Available online at www.sciencedirect.com
ScienceDirect
Communications in Nonlinear Science and Numerical Simulation 14 (2009) 760–769
Communications in Nonlinear Science and Numerical Simulation
www.elsevier.com/locate/cnsns

Numerical solutions for convection–diffusion equation using El-Gendi method

R.S. Temsah

Mathematics Department, Faculty of Science, Beni-Suef University, Beni-Suef, Egypt

Received 24 April 2007; received in revised form 15 July 2007; accepted 14 November 2007
Available online 22 November 2007

Abstract

The method of El-Gendi [El-Gendi SE. Chebyshev solution of differential integral and integro-differential equations. J Comput 1969;12:282–7; Mihaila B, Mihaila I. Numerical approximation using Chebyshev polynomial expansions: El-gendi's method revisited. J Phys A Math Gen 2002;35:731–46] is presented with interface points to deal with linear and non-linear convection–diffusion equations.

The linear problem is reduced to two systems of ordinary differential equations. And, then, each system is solved using three-level time scheme.

The non-linear problem is reduced to three systems of ordinary differential. Each one of these systems is, then, solved using three-level time scheme. Numerical results for Burgers' equation and modified Burgers' equation are shown and compared with other methods. The numerical results are found to be in good agreement with the exact solutions.

PACS: 03.40.Gc; 02.70.Hm; 02.70.Jn; 02.60.−x

Keywords: Numerical solutions; Chebyshev spectral collocation method; Convection–diffusion equations; Burgers' equation; Modified Burgers' equation

1. Introduction

As is well known, the numerical approximation of convection–diffusion equations requires some special treatment in order to obtain good results when problem is convection dominated due to the presence of boundary or interior layers. A lot of work has been done in this direction (see for example [3–11] and their references).

Burgers' equation is a very important fluid dynamical model for both the understanding of a class of physical flows and for testing various numerical algorithms. Many researchers have used various numerical methods to solve Burgers' equation [12–19].

E-mail address: Eltemssah@yahoo.com

1007-5704/$ - see front matter
doi:10.1016/j.cnsns.2007.11.004

The purpose of this paper is to elaborate the Chebyshev spectral collocation methods (known as El-Gendi method [1,2]) to obtain numerical solutions of the following linear and non-linear convection–diffusion equations:

(P1) The linear convection–diffusion problem

$$u_t = \epsilon u_{xx} - u_x, \quad 0 \leqslant x \leqslant 1, \ t \geqslant 0, \ \epsilon > 0 \tag{1}$$

(P2) The non-linear convection–diffusion problem

$$u_t = \epsilon u_{xx} - [f(x,u,t)u]_x, \quad 0 \leqslant x \leqslant 1, \ t \geqslant 0, \ \epsilon > 0 \tag{2}$$

For numerical experiments we consider the following two models of P2:

(i) The Burgers' equation

$$u_t = \epsilon u_{xx} - uu_x, \quad 0 \leqslant x \leqslant 1, \ t \geqslant 0, \ \epsilon > 0 \tag{3}$$

(ii) The modified Burgers' equation

$$u_t = \epsilon u_{xx} - u^2 u_x, \quad 0 \leqslant x \leqslant 1, \ t \geqslant 0, \ \epsilon > 0 \tag{4}$$

In all problems $u(0,t)$, $u(1,t)$ and $u(x,0)$ are prescribed.

It is known that El-Gendi method is not satisfactory for the problems P1 and P2 when ϵ is small, therefore,

(a) For linear problem P1, we introduce an appropriate interface point $d \in R = [0,1]$ and hence apply the El-Gendi method in two intervals $R_1 = [0,d]$ and $R_2 = [d,1]$. The problem is reduced to two systems of ordinary differential equations. And, then, each system is solved using three-level time scheme.
(b) For non-linear problem P2, we introduce two interface points d_1 and d_2 and hence apply the El-Gendi method in the three intervals $R_1 = [0,d_1]$, $R_2 = [d_1,d_2]$ and $R_3 = [d_2,1]$. Using three-level time scheme the method is to be compared with [14,16].

The paper is organized as follows: In Section 2, El-Gendi method with interface points are used to obtain numerical solutions of the problems P1 and P2, the numerical results are presented in Sections 3 and 4 gives the conclusions.

2. Numerical solutions by El-Gendi method with interface point

To illustrate the procedure, two problems related to the linear convection–diffusion problem and non-linear convection–diffusion problem are given in the following.

2.1. The linear convection–diffusion problem

Consider problem P1, we now seek solutions $u_k(x,t)$ defined on R_k, $k = 1,2$ to the problems

$$\dot{u}_k = \epsilon u_k'' - u_k', \quad k = 1,2 \tag{5}$$

$$u_1(0,t) = \alpha, \quad u_2(1,t) = \beta \tag{6}$$

subject to the interface conditions

$$u_1(d,t) = u_2(d,t), \quad u_1'(d,t) = u_2'(d,t) \tag{7}$$

where $\dot{u}_k = \frac{\partial}{\partial t}u_k(x,t)$, $u_k' = \frac{\partial}{\partial x}u_k(x,t)$ and $u_k'' = \frac{\partial^2}{\partial x^2}u_k(x,t)$.

Integrating each equation of (5) with respect to x twice in its region, we obtain

$$\int_{x_0^{(k)}}^{x}\int_{x_0^{(k)}}^{x}\dot{u}_k\,\mathrm{d}x\,\mathrm{d}x = \epsilon\left[u_k(x,t) - u_k\left(x_0^{(k)},t\right) - \left(x - x_0^{(k)}\right)u_k'\left(x_0^{(k)},t\right)\right] + \left(x - x_0^{(k)}\right)u_k\left(x_0^{(k)},t\right) + \int_{x_0^{(k)}}^{x} u_k\,\mathrm{d}x, \quad k = 1,2 \tag{8}$$

By setting $x = x_{N_k}^{(k)}$ in each equation of (8), we get

$$\int_{x_0^{(k)}}^{x_{N_k}^{(k)}}\int_{x_0^{(k)}}^{x}\dot{u}_k\,\mathrm{d}x\,\mathrm{d}x = \epsilon\left[u_k\left(x_{N_k}^{(k)},t\right) - u_k\left(x_0^{(k)},t\right) - \left(x_{N_k}^{(k)} - x_0^{(k)}\right)u_k'\left(x_0^{(k)},t\right)\right] + \left(x_{N_k}^{(k)} - x_0^{(k)}\right)u_k\left(x_0^{(k)},t\right) + \int_{x_0^{(k)}}^{x_{N_k}^{(k)}} u_k\,\mathrm{d}x, \quad k = 1,2 \tag{9}$$

From Eqs. (8) and (9), it is not difficult to show that

$$\left(\int_{x_0^{(k)}}^{x}\int_{x_0^{(k)}}^{x} - \frac{\left(x - x_0^{(k)}\right)}{\left(x_{N_k}^{(k)} - x_0^{(k)}\right)}\int_{x_0^{(k)}}^{x_{N_k}^{(k)}}\int_{x_0^{(k)}}^{x}\right)\dot{u}_k\,\mathrm{d}x\,\mathrm{d}x = \epsilon\left[u_k - \frac{\left(x - x_0^{(k)}\right)}{\left(x_{N_k}^{(k)} - x_0^{(k)}\right)}u_k\left(x_{N_k}^{(k)},t\right) + \frac{\left(x - x_{N_k}^{(k)}\right)}{\left(x_{N_k}^{(k)} - x_0^{(k)}\right)}u_k\left(x_0^{(k)},t\right)\right] - \left(\int_{x_0^{(k)}}^{x} - \frac{\left(x - x_0^{(k)}\right)}{\left(x_{N_k}^{(k)} - x_0^{(k)}\right)}\int_{x_0^{(k)}}^{x_{N_k}^{(k)}}\right)u_k\,\mathrm{d}x, \quad k = 1,2 \tag{10}$$

Differentiating each equation of (10) with respect to x, we obtain

$$\int_0^x \dot{u}_1\,\mathrm{d}x - \frac{1}{d}\int_0^d\int_0^x \dot{u}_1\,\mathrm{d}x\,\mathrm{d}x = \epsilon\left[u_1'(x,t) - \frac{1}{d}u_1(d,t) + \frac{1}{d}u_1(0,t)\right] - \left(u_1(x,t) - \frac{1}{d}\int_0^d u_1\,\mathrm{d}x\right), \quad k = 1,2 \tag{11}$$

$$\int_0^x \dot{u}_2 dx - \frac{1}{1-d}\int_d^1\int_d^x \dot{u}_2\,\mathrm{d}x\,\mathrm{d}x = \epsilon\left[u_2'(x,t) - \frac{1}{1-d}u_2(1,t) + \frac{1}{1-d}u_2(d,t)\right] - \left(u_2(x,t) - \frac{1}{1-d}\int_d^1 u_2\,\mathrm{d}x\right), \quad k = 1,2 \tag{12}$$

We set $x = d$ in Eqs. (11) and (12) and using the boundary and interface conditions (6) and (7), it is not difficult to show that

$$\epsilon\left(\frac{1}{d} + \frac{1}{1-d}\right)u_1(d,t) = \epsilon\left(\frac{\alpha}{d} + \frac{\beta}{1-d}\right) - \left(\int_0^d \dot{u}_1 dx - \frac{1}{d}\int_0^d\int_0^x \dot{u}_1\,\mathrm{d}x\,\mathrm{d}x\right) + \frac{1}{d}\int_0^d u_1 dx - \frac{1}{1-d}\int_d^1\int_d^x \dot{u}_2\,\mathrm{d}x\,\mathrm{d}x - \frac{1}{1-d}\int_d^x u_2\,\mathrm{d}x \tag{13}$$

By using El-Gendi method for discretizing equations (10) and (13) with Chebyshev points

$$x_j^{(1)} = \frac{d}{2} - \frac{d}{2}\cos\left(\frac{\pi j}{N_1}\right) \in R_1, \quad j = 0,1,\ldots,N_1 \tag{14}$$

and

$$x_j^{(2)} = \frac{1+d}{2} - \frac{1-d}{2}\cos\left(\frac{\pi j}{N_2}\right) \in R_2, \quad j = 0,1,\ldots,N_2 \tag{15}$$

where $N_k,\ k=1,2$ are the number of mesh points in R_1 and R_2, respectively, we get,

$$L^{(k)}|\dot{u}_k| = \epsilon|u_k| - \epsilon\left[\frac{\left(x-x_0^{(k)}\right)}{\left(x_{N_k}^{(k)}-x_0^{(k)}\right)}\right]u_k\left(x_{N_k}^{(k)},t\right) + \epsilon\left[\frac{\left(x-x_{N_k}^{(k)}\right)}{\left(x_{N_k}^{(k)}-x_0^{(k)}\right)}\right]u_k\left(x_0^{(k)},t\right) - R^{(k)}|u_k|, \quad k=1,2 \tag{16}$$

$$\begin{aligned}\epsilon\left(\frac{1}{d}+\frac{1}{1-d}\right)u_1(d,t) = {} & \epsilon\left(\frac{\alpha}{d}+\frac{\beta}{1-d}\right) - \sum_{j=0}^{N_1}\left(B_{N_1j}^{(1)}-\frac{1}{d}T_{N_1j}^{(1)}\right)\dot{u}_1\left(x_j^{(1)},t\right) + \frac{1}{d}\sum_{j=0}^{N_1}B_{N_1j}^{(1)}u_1\left(x_j^{(1)},t\right)\\ & - \frac{1}{1-d}\sum_{j=0}^{N_2}T_{N_2j}^{(2)}\dot{u}_2\left(x_j^{(2)},t\right) - \frac{1}{1-d}\sum_{j=0}^{N_2}B_{N_2j}^{(2)}u_2\left(x_j^{(2)},t\right)\end{aligned} \tag{17}$$

where

$$T_{ij}^{(k)} = (x_i^{(k)}-x_j^{(k)})B_{ij}^{(k)}$$

$$L_{ij}^{(k)} = T_{ij}^{(k)} - \left[\frac{(x-x_0^{(k)})}{(x_{N_k}^{(k)}-x_0^{(k)})}\right]T_{N_kj}^{(k)}$$

$$R_{ij}^{(k)} = B_{ij}^{(k)} - \left[\frac{(x-x_0^{(k)})}{(x_{N_k}^{(k)}-x_0^{(k)})}\right]B_{N_kj}^{(k)} \quad i,j=0,1,\ldots,N_k,\ \ k=1,2$$

$B^{(K)}|y| = [\int_{x_0^{(k)}}^{x^{(k)}} y(x)\mathrm{d}x]$ is defined in [1] and [ ] denotes a diagonal matrix.

The systems (16), (17) are solved using the three-level time scheme

$$L^{(k)}|\hat{u}_k| = \epsilon|\tilde{u}_k| - \epsilon\left[\frac{(x-x_0^{(k)})}{(x_{N_k}^{(k)}-x_0^{(k)})}\right]\tilde{u}_k(x_{N_k}^{(k)},t) + \epsilon\left[\frac{(x-x_{N_k}^{(k)})}{(x_{N_k}^{(k)}-x_0^{(k)})}\right]\tilde{u}_k(x_0^{(k)},t) - R^{(k)}|\tilde{u}_k|, \quad k=1,2 \tag{18}$$

$$\begin{aligned}\epsilon\left(\frac{1}{d}+\frac{1}{1-d}\right)u_1(d,t) = {} & \epsilon\left(\frac{\alpha}{d}+\frac{\beta}{1-d}\right) - \sum_{j=0}^{N_1}\left(B_{N_1j}^{(1)}-\frac{1}{d}T_{N_1j}^{(1)}\right)\hat{u}_1(x_j^{(1)},t) + \frac{1}{d}\sum_{j=0}^{N_1}B_{N_1j}^{(1)}\tilde{u}_1(x_j^{(1)},t)\\ & - \frac{1}{1-d}\sum_{j=0}^{N_2}T_{N_2j}^{(2)}\hat{u}_2(x_j^{(2)},t) - \frac{1}{1-d}\sum_{j=0}^{N_2}B_{N_2j}^{(2)}\tilde{u}_2(x_j^{(2)},t)\end{aligned} \tag{19}$$

where

$$\hat{u}_k = ((\theta+1/2)u_k^{n+1} - 2\theta u_k^n + (\theta-1/2)u_k^{n-1})/\Delta t$$
$$\tilde{u}_k = \theta u_k^{n+1} + (1-\theta)u_k^n, \quad k=1,2,\ \ 1/2 \leqslant \theta \leqslant 1$$

The system ((18), $k=1$) is used to eliminate the coefficients $u_1(x_j^{(1)},t_{n+1}),\ j=0,1,\ldots,N_1$ in Eq. (19). The resulting equation in conjunction with system ((18), $k=2$) are used to determine the values $u_2(x_j^{(2)},t_{n+1})$, $j=0,1,\ldots,N_2$ and $u_1(d,t_{n+1})$. Hence on using system ((18), $k=1$), the values of $u_1(x_j^{(1)},t_{n+1})$, $j=0,1,\ldots,N_1$ are determined.

2.2. The non-linear convection–diffusion problem

Consider problem P2, we now seek solutions $u_k(x,t)$ defined on R_k, $k=1,2,3$ to the problems

$$\dot{u}_k = \epsilon u_k'' - f(x^k,t,u_k)u_k', \quad k=1,2,3 \tag{20}$$
$$u_1(0,t)=\alpha, \quad u_3(1,t)=\beta \tag{21}$$

subject to the interface conditions

$$\begin{aligned} u_1(d_1,t) &= u_2(d_1,t), \quad u_2(d_2,t) = u_3(d_2,t)\\ u_1'(d_1,t) &= u_2'(d_1,t), \quad u_2'(d_2,t) = u_3'(d_2,t)\end{aligned} \tag{22}$$

Applying El-Gendi method to the problem (20)–(22), one gets

$$L^{(k)}|\dot{u}_k| = \epsilon|u_k| - \epsilon\left[\frac{(x-x_0^{(k)})}{(x_{N_k}^{(k)}-x_0^{(k)})}\right]u_k(x_{N_k}^{(k)},t) + \epsilon\left[\frac{(x-x_{N_k}^{(k)})}{(x_{N_k}^{(k)}-x_0^{(k)})}\right]u_k(x_0^{(k)},t) - R^{(k)}[f(x^k,t,u_k)]|u_k|, \quad k=1,2,3 \tag{23}$$

$$\begin{aligned}\epsilon\left(\frac{1}{x_{N_\ell}^{(\ell)}-x_0^{(\ell)}}+\frac{1}{x_{N_{\ell+1}}^{(\ell+1)}-x_0^{(\ell+1)}}\right)u_\ell\left(x_{N_\ell}^{(\ell)},t\right) = {} & \left(\frac{\epsilon}{x_{N_\ell}^{(\ell)}-x_0^{(\ell)}}\right)u_\ell\left(x_0^{(\ell)},t\right)+\left(\frac{\epsilon}{x_{N_{\ell+1}}^{(\ell+1)}-x_0^{(\ell+1)}}\right)u_{\ell+1}\left(x_{N_{\ell+1}}^{(\ell+1)},t\right)\\ & -\sum_{j=0}^{N_\ell}\left(B_{N_\ell j}^{(\ell)}-\frac{1}{x_{N_\ell}^{(\ell)}-x_0^{(\ell)}}T_{N_\ell j}^{(\ell)}\right)\dot{u}_\ell\left(x_j^{(\ell)},t\right)\\ & +\frac{1}{x_{N_\ell}^{(\ell)}-x_0^{(\ell)}}\sum_{j=0}^{N_\ell}B_{N_\ell j}^{(\ell)}f_j^{(\ell)}u_\ell\left(x_j^{(\ell)},t\right)\\ & -\frac{1}{x_{N_{\ell+1}}^{(\ell+1)}-x_0^{(\ell+1)}}\left(\sum_{j=0}^{N_{\ell+1}}T_{N_{\ell+1}j}^{(\ell+1)}\dot{u}_{\ell+1}\left(x_j^{(\ell+1)},t\right)\right.\\ & \left.-\sum_{j=0}^{N_{\ell+1}}B_{N_{\ell+1}j}^{(\ell+1)}f_j^{(\ell+1)}u_{\ell+1}\left(x_j^{(\ell+1)},t\right)\right), \quad \ell=1,2\end{aligned} \tag{24}$$

where

$$\begin{aligned}&f_j^{(\ell)} = f(x_j^{(\ell)},t,u_\ell(x^{(\ell)},t))\\ &x_j^{(1)} = \frac{d_1}{2}-\frac{d_1}{2}\cos\left(\frac{\pi j}{N_1}\right)\in R_1, \quad j=0,1,\ldots,N_1\\ &x_j^{(2)} = \frac{d_1+d_2}{2}-\frac{d_2-d_1}{2}\cos\left(\frac{\pi j}{N_2}\right)\in R_2, \quad j=0,1,\ldots,N_2\\ &x_j^{(3)} = \frac{1+d_2}{2}-\frac{1-d_2}{2}\cos\left(\frac{\pi j}{N_3}\right)\in R_3, \quad j=0,1,\ldots,N_3\end{aligned}$$

and $N_k, k=1,2,3$ are the number of mesh points in R_1, R_2 and R_3, respectively.

The systems (23), (24) are solved using the three-level time scheme,

$$L^{(k)}|\hat{u}_k| = \epsilon|\tilde{u}_k| - \epsilon\left[\frac{(x-x_0^{(k)})}{(x_{N_k}^{(k)}-x_0^{(k)})}\right]\tilde{u}_k(x_{N_k}^{(k)},t) + \epsilon\left[\frac{(x-x_{N_k}^{(k)})}{(x_{N_k}^{(k)}-x_0^{(k)})}\right]\tilde{u}_k(x_0^{(k)},t) - R^{(k)}[f(x^k,t,\bar{u}_k)]|\bar{u}_k|, \quad k=1,2,3 \tag{25}$$

$$\begin{aligned}\epsilon\left(\frac{1}{x_{N_\ell}^{(\ell)}-x_0^{(\ell)}}+\frac{1}{x_{N_{\ell+1}}^{(\ell+1)}-x_0^{(\ell+1)}}\right)\tilde{u}_\ell\left(x_{N_\ell}^{(\ell)},t\right) = {} & \left(\frac{\epsilon}{x_{N_\ell}^{(\ell)}-x_0^{(\ell)}}\right)u_\ell\left(x_0^{(\ell)},t\right)+\left(\frac{\epsilon}{x_{N_{\ell+1}}^{(\ell+1)}-x_0^{(\ell+1)}}\right)u_{\ell+1}\left(x_{N_{\ell+1}}^{(\ell+1)},t\right)\\ & -\sum_{j=0}^{N_\ell}\left(B_{N_\ell j}^{(\ell)}-\frac{1}{x_{N_\ell}^{(\ell)}-x_0^{(\ell)}}T_{N_\ell j}^{(\ell)}\right)\hat{u}_\ell\left(x_j^{(\ell)},t\right)\\ & +\frac{1}{x_{N_\ell}^{(\ell)}-x_0^{(\ell)}}\sum_{j=0}^{N_\ell}B_{N_\ell j}^{(\ell)}\bar{f}_j^{(\ell)}\bar{u}_\ell\left(x_j^{(\ell)},t\right)\\ & -\frac{1}{x_{N_{\ell+1}}^{(\ell+1)}-x_0^{(\ell+1)}}\left(\sum_{j=0}^{N_{\ell+1}}T_{N_{\ell+1}j}^{(\ell+1)}\hat{u}_{\ell+1}\left(x_j^{(\ell+1)},t\right)\right.\\ & \left.-\sum_{j=0}^{N_{\ell+1}}B_{N_{\ell+1}j}^{(\ell+1)}\bar{f}_j^{(\ell+1)}\bar{u}_{\ell+1}\left(x_j^{(\ell+1)},t\right)\right), \quad \ell=1,2.\end{aligned} \tag{26}$$

where

$$\bar{f}_j^{(\ell)} = f\left(x_j^{(\ell)}, t, \bar{u}_\ell\left(x^{(\ell)}, t\right)\right)$$
$$\bar{u}_k = (1+\theta)u_k^n + (1-\theta)u_k^{n-1}, \quad k = 1,2,3, \quad 1/2 \leqslant \theta \leqslant 1$$

The system ((25), $k = 1$) is used to eliminate the coefficients $u_1\left(x_j^{(1)}, t_{n+1}\right)$, $j = 0, 1, \ldots, N_1$ in Eq. ((26), $\ell = 1$) and the system ((25), $k = 3$) is used to eliminate the coefficients $u_3\left(x_j^{(3)}, t_{n+1}\right)$, $j = 0, 1, \ldots, N_3$ in Eq. ((26), $\ell = 2$). The resulting equation in conjunction with system ((25), $k = 2$) are used to determine the values $u_2\left(x_j^{(2)}, t_{n+1}\right)$, $j = 0, 1, \ldots, N_2$, $u_1(d_1, t_{n+1})$ and $u_2(d_2, t_{n+1})$. Hence on using systems ((25), $k = 1, 3$), the values of $u_1\left(x_j^{(1)}, t_{n+1}\right)$, $j = 0, 1, \ldots, N_1$ and $u_3\left(x_j^{(3)}, t_{n+1}\right)$, $j = 0, 1, \ldots, N_3$ are determined.

3. Numerical results

In this section, we will present numerical results of the linear and non-linear convection–diffusion equations for some test problems. The accuracy of the numerical methods is measured by computing the difference between the exact and numerical solutions at each mesh point and use these to compute the L_∞ error norm.

Example 1. Consider problem P1 with steady-state solution [5,6,13]

$$u(x) = (e^{(x/\epsilon)} - 1)/(e^{(1/\epsilon)} - 1) \tag{27}$$

In Table 1, we give the maximum absolute error [MAER] between the steady-state solution (27) and the results obtained by the proposed method with three-level time scheme ($\theta = 1$) for different values of ϵ.

Example 2. Consider problem P1 with steady-state solution [7]

$$u(x) = (1 - e^{(x-1)/\epsilon})/(1 - e^{(-1/\epsilon)}) \tag{28}$$

In Table 2, we give the maximum absolute error [MAER] between the steady-state solution (28) and the results obtained by the proposed method with three-level time scheme ($\theta = 1$) for different values of ϵ.

Example 3. Consider the *Burgers*' equation (3) with exact solution [15]

$$u(x,t) = [\mu + \alpha + (\mu - \alpha)\exp(\alpha\xi/\epsilon)]/[1 + \exp(\alpha\xi/\epsilon)] \tag{29}$$

Table 1
$\Delta t = 0.05$

ϵ	N_1	N_2	d	MAER
0.02	10	10	0.85	1.4, −5
10^{-2}	10	10	0.9	1.5, −5
10^{-3}	10	10	0.99	5.4, −5
10^{-5}	10	10	0.9999	2.3, −4

Table 2
$\Delta t = 0.1$

ϵ	N_1	N_2	d	*MAER*
10^{-2}	5	10	0.9	3.68, −5
10^{-3}	10	10	0.99	6.38, −5
10^{-4}	10	10	0.999	9.96, −5
10^{-5}	10	10	0.9999	2.73, −4

Table 3
$\Delta t = 0.02,\ N_1 = N_3 = 5,\ N_2 = 7,\ \alpha = 0.1,\ \mu = 0.9,\ \beta = 0.125$

ϵ	d_1	d_2	t			
			2	3	4	5
10^{-2}	0.02	0.98	4.1, −5	1.7, −5	2.0, −5	2.7, −5
10^{-3}	0.002	0.998	8.1, −4	1.5, −4	5.5, −5	6.5, −5
10^{-5}	0.00002	0.99998	8.4, −4	5.5, −4	4.4, −4	4.3, −4

Table 4
$\Delta t = 0.02,\ N_1 = N_3 = 5,\ N_2 = 7,\ \alpha = 0.4,\ \mu = 0.6,\ \beta = 0.125$

ϵ	d_1	d_2	t		
			3	4	5
10^{-2}	0.02	0.98	3.2, −4	2.8, −5	1.9, −5
10^{-3}	0.002	0.998	2.6, −3	4.7, −4	6.6, −5
10^{-5}	0.00002	0.99998	2.8, −3	8.8, −4	4.4, −4

where $\xi = x - \mu t - \beta$. In Tables 3, 4, we give the maximum absolute error between the exact solution (29) and the results obtained by the proposed method with three-level time scheme ($\theta = 1$) for different values of α, β, μ and ϵ. From Tables 3 and 4 we note that for all the values of ϵ used, the errors are small and acceptable.

Fig. 1, illustrates the behavior of the numerical solution for $N_1 = 5$, $N_2 = 10$, $N_3 = 5$, $\Delta t = 0.01$, $\mu = 0.6$, $\beta = 0.125$, $\alpha = 0.4$, $\epsilon = 0.01$, $d_1 = 0.2$ and $d_2 = 0.8$ at some different times $t = 0.4$, 0.75 and 1.0. Error between the numerical and exact solution is also depicted at time $t = 0.5$ in Fig. 2 for $N_1 = 5$, $N_2 = 7$, $N_3 = 5$, $\Delta t = 0.01$, $\mu = 0.6$, $\beta = 0.125$, $\alpha = 0.4$, $\epsilon = 0.01$, $d_1 = 0.2$ and $d_2 = 0.8$, and maximum error concentrates at around the wave front.

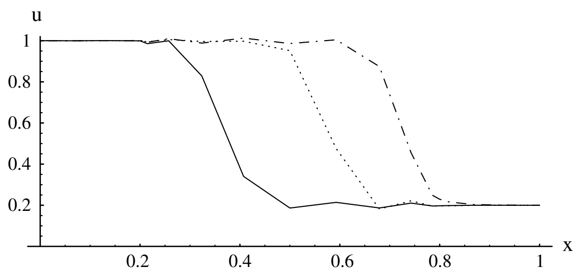

Fig. 1. Numerical solution of Example 3 at different times for $N_1 = 5$, $N_2 = 10$, $N_3 = 5$, $\Delta t = 0.01$, $\mu = 0.6$, $\beta = 0.125$, $\alpha = 0.4$, $\epsilon = 0.01$, $d_1 = 0.2$, $d_2 = 0.8$, (— $t = 0.4$, ····· $t = 0.75$, –·– $t = 1.0$).

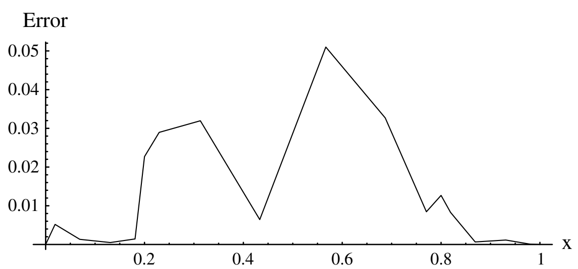

Fig. 2. Error (|Numerical-Exact|) of Example 3 at time $t = 0.5$ for $N_1 = 5$, $N_2 = 7$, $N_3 = 5$, $\Delta t = 0.01$, $\mu = 0.6$, $\beta = 0.125$, $\alpha = 0.4$, $\epsilon = 0.01$, $d_1 = 0.2$, $d_2 = 0.8$.

Example 4. Consider the *Burgers*' equation (4) with the exact solution

$$u(x,t) = (x/t)/[1 + (\sqrt{t/w_0})\exp(x^2/4t\epsilon)] \tag{30}$$

where $w_0 = \exp(1/8\epsilon)$. In Table 5, we give the maximum absolute error between the exact solution (30) and the results obtained by the proposed method with three-level time scheme ($\theta = 1$), compared with the results given by [16]with $\Delta t = 0.01$ and $\Delta x = 0.02$ given between brackets. It is seen that results of the present method are better than results obtained by [16]. Fig. 3, illustrates the behavior of the numerical solution for $N_1 = 5$, $N_2 = 10$, $N_3 = 5$, $\Delta t = 0.01$, $\epsilon = 0.01$, $d_1 = 0.2$ and $d_2 = 0.8$ at some different times $t = 1.3$, 1.5 and 1.7, top curve at $t = 1.3$ and bottom at $t = 1.7$. Fig. 4, illustrates the behavior of the numerical solution for $N_1 = 5$, $N_2 = 10$, $N_3 = 5$, $\Delta t = 0.01$, $\epsilon = 0.005$, $d_1 = 0.2$ and $d_2 = 0.8$ at some different times $t = 1.5$, 1.7 and 2.1, top curve at $t = 1.5$ and bottom at $t = 2.1$. Figs. 4 and 3 show that as the time increases the curve of the numerical solution decays. Error between the numerical and exact solution is also depicted at time $t = 3.1$ in Fig. 5 for $N_1 = 5$, $N_2 = 10$, $N_3 = 5$, $\Delta t = 0.01$, $\epsilon = 0.005$, $d_1 = 0.2$ and $d_2 = 0.8$. We see that maximum error occurs at the right hand boundary.

Example 5. Consider the modified *Burgers*' equation (4) with the exact solution [17]

$$u(x,t) = (x/t)/[1 + (\sqrt{t}/c_0)\exp(x^2/4t\epsilon)] \tag{31}$$

Table 5
$\Delta t = 0.01$, $N_1 = N_3 = 5$, $N_2 = 15$, $d_1 = 0.2$, $d_2 = 0.8$

ϵ	t					
	1.5	1.7	1.9	2.1	2.3	2.6
0.010	1.41, −3 (3.36, −3)	1.01, −3 (3.14, −3)	9.30, −4 (2.89, −3)	8.34, −4 (2.67, −3)	7.17, −4 (2.81, −3)	5.76, −4 (8.07, −3)
0.005	5.57, −3	3.49, −3	3.85, −3	2.32, −3	2.10, −3	2.45, −3

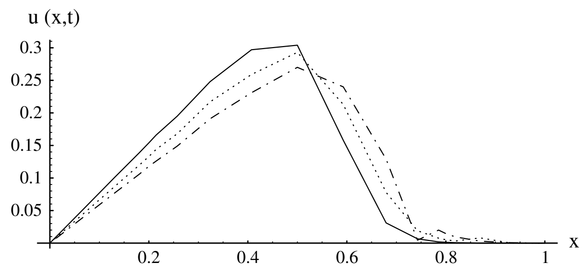

Fig. 3. Numerical solution of Example 4 at different times for $N_1 = 5$, $N_2 = 10$, $N_3 = 5$, $\Delta t = 0.01$, $\epsilon = 0.01$, $d_1 = 0.2$, $d_2 = 0.8$, (— $t = 1.3$, ····· $t = 1.5$, –·–· $t = 1.7$).

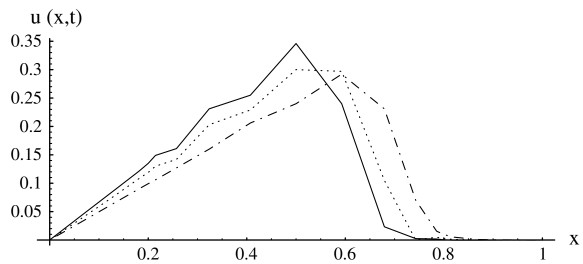

Fig. 4. Numerical solution of Example 4 at different times for $N_1 = 5$, $N_2 = 10$, $N_3 = 5$, $\Delta t = 0.01$, $\epsilon = 0.005$, $d_1 = 0.2$, $d_2 = 0.8$, (—— $t = 1.5$, ····· $t = 1.7$, –·–· $t = 2.1$).

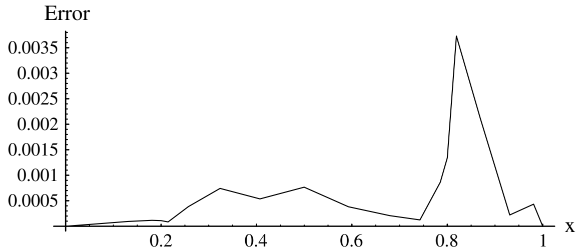

Fig. 5. Error (|Numerical-Exact|) of Example 4 at time $t = 3.1$ for $N_1 = 5$, $N_2 = 10$, $N_3 = 5$, $\Delta t = 0.01$, $\epsilon = 0.005$, $d_1 = 0.2$, $d_2 = 0.8$.

where c_0 is constant greater than zero and we noted that $0 < c_0 < 1$. In Table 6, we give the maximum absolute error between the exact solution (31) and the results obtained by the proposed method with three-level time scheme ($\theta = 1$), compared with the results given by [14] given between brackets and with the results given by [16], which are between square brackets. It is seen that results of the present method are better than results obtained by [14,16].

Table 6
$\Delta t = 0.01$, $N_1 = N_3 = 5$, $N_2 = 10$, $c_0 = 0.5$, $d_1 = 0.1$, $d_2 = 0.9$

ϵ	t					
	2	4	6	8	10	11
10^{-2}	7.58, −4	5.64, −4	4.59, −4	3.66, −4	3.00, −4	2.68, −4
	(1.22, −3)	(9.31, −4)	(7.23, −4)	(9.63, −4)	(1.28, −3)	(1.39, −3)
	[1.70, −3]	[9.97, −4]	[7.61, −4]	[1.36, −3]	[1.80, −3]	
10^{-3}	2.73, −4	1.57, −4	1.39, −4	1.10, −4	9.36, −5	9.26, −5
	(2.80, −4)	(2.19, −4)	(1.72, −4)	(1.42, −4)	(1.21, −4)	(1.13, −4)
	[8.19, −4]	[3.56, −4]	[2.14, −4]	[1.68, −4]	[1.39, −4]	

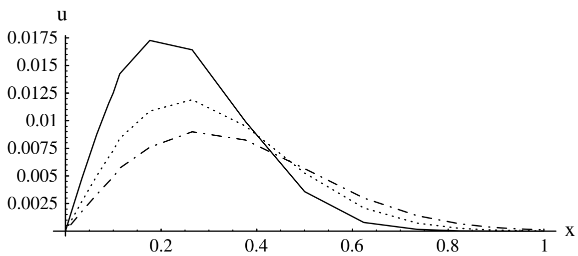

Fig. 6. Numerical solution of Example 5 at different times for $N_1 = 5$, $N_2 = 10$, $N_3 = 5$, $\Delta t = 0.01$, $c_0 = 0.5$, $\epsilon = 0.01$, $d_1 = 0.1$, $d_2 = 0.9$, (— $t = 2$, ····· $t = 3$, –·–· $t = 4$).

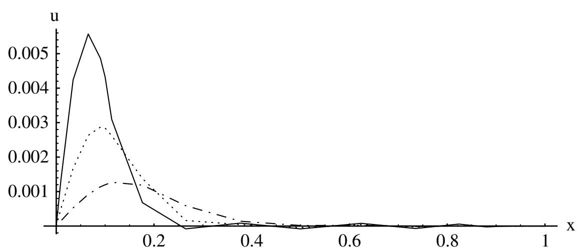

Fig. 7. Numerical solution of Example 5 at different times for $N_1 = 5$, $N_2 = 10$, $N_3 = 5$, $\Delta t = 0.01$, $c_0 = 0.5$, $\epsilon = 0.001$, $d_1 = 0.1$, $d_2 = 0.9$, (— $t = 2$, ····· $t = 4$, –·–· $t = 9$).

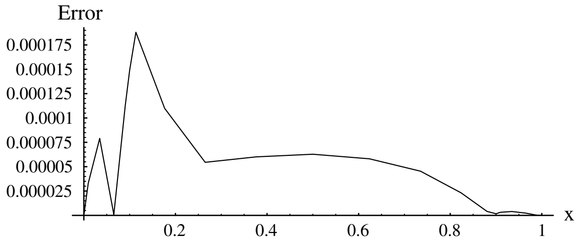

Fig. 8. Error (|Numerical-Exact|) of Example 5 at time $t = 3.25$ for $N_1 = 5$, $N_2 = 10$, $N_3 = 5$, $\Delta t = 0.01$, $c_0 = 0.5$, $\epsilon = 0.001$, $d_1 = 0.1$, $d_2 = 0.9$.

Fig. 6, illustrates the behavior of the numerical solution for $N_1 = 5$, $N_2 = 10$, $N_3 = 5$, $\Delta t = 0.01$, $\epsilon = 0.01$, $d_1 = 0.1$ and $d_2 = 0.9$ at some different times $t = 2$, 3 and 4, top curve at $t = 2$ and bottom at $t = 4$. Fig. 7 illustrates the behavior of the numerical solution for $N_1 = 5$, $N_2 = 10$, $N_3 = 5$, $\Delta t = 0.01$, $\epsilon = 0.001$, $d_1 = 0.1$ and $d_2 = 0.9$ at some different times $t = 2$, 4 and 9, top curve at $t = 2$ and bottom at $t = 9$. The Figs. 6 and 7 show that as the time increases the curve of the numerical solution decays. Error between the numerical and exact solution is also depicted at time $t = 3.25$ in Fig. 8 for $N_1 = 5$, $N_2 = 10$, $N_3 = 5$, $\Delta t = 0.01$, $\epsilon = 0.005$, $d_1 = 0.1$ and $d_2 = 0.9$. We see that maximum error occurs at the left hand boundary.

4. Conclusions

In this paper, a numerical solution for the convection–diffusion equation is proposed using El-Gendi method with interface points. Numerical results for Burgers' equation and modified Burgers' equation are shown. The obtained approximate numerical solution maintains a good accuracy in little computer time compared with exact solution and other methods especially for small values of ϵ. The present method is also useful for solving more general problems in fluid dynamics.

References

[1] El-Gendi SE. Chebyshev solution of differential integral and integro-differential equations. J Comput 1969;12:282–7.
[2] Mihaila B, Mihaila I. Numerical approximation using Chebyshev polynomial expansions: El-gendi's method revisited. J Phys A Math Gen 2002;35:731–46.
[3] Morton KW. Numerical solution of convection–diffusion problems. London: Chapman and Hall; 1996.
[4] Roos RG, Stynes M, Tobiska L. Numerical methods for singularly perturbed differential equations. Berlin: Springer; 1996.
[5] Evans DJ, Abdullah AR. A new explicit method for the diffusion–convection equation. Comp Math Appl 1985;11:145–54.
[6] Varah JM. Stability restrictions on second order, three level finite difference schemes for parabolic equations. Siam J Numer Anal 1980;17:300–9.
[7] Haidvogel DB. Resolution of downstream boundary layers in Chebyshev approximation to viscous flow problems. J Comput Phys 1979;33:313–24.
[8] Miller JJH, editor. The third international conference on boundary and interior layers – computational and asymptotic methods. Dublin: Boole Press; 1984.
[9] Miller JJH, editor. Advances in computational methods for boundary and interior layers. Dublin: Boole Press; 1984.
[10] Ibrahim MAK, Temsah RS. Spectral methods for some singularly perturbed problems with initial and boundary layers. Int J Comput Math 1988;25:33–48.
[11] El-Gendi SE, Shamardan AB. A note on Chebyshev methods for problems with initial and boundary layers, in [3]; 1984.
[12] El-Gendi SE. On numerical methods for convection–diffusion equation, in [3]; 1984.
[13] El-Gendi SE, Ibrahim MAK, Shamardan AB. The numerical treatment of non-linear parabolic partial differential equations governing convection–diffusion processes. Int J Comput Math 1988;25:111–27.
[14] Ramadan MA, El-Danaf TS. Numerical treatment for modified Burgers' equation. Math Comput Simulat 2005;70:90–8.
[15] Dag I, Irk D, Saka B. A numerical solution of the Burgers' equation using cubic spline. Appl Math Comput 2005;163:199–211.
[16] Ramadan MA, El-Danaf TS, Abd Alaal FET. A numerical solution of the Burgers' equation using Septic. B-Splines. Chaos Soliton Fract 2005;26:1249–58.
[17] Harris SI. Sonic shocks governed by the modified Burgers' equation. EJAM 1996;6:75–107.
[18] Aksan EN, Özdes A, Özis T. A numerical solution of Burgers' equation based on least squares approximation. Appl Math Comput 2006;176:270–9.
[19] Gulsu M. A finite difference approach for solution of Burgers' equation. Appl Math Comput 2006;175:1245–55.