Local discontinuous Galerkin method for solving Burgers and coupled Burgers equations*

Zhang Rong-Pei(张荣培)[a)†], Yu Xi-Jun(蔚喜军)[b)], and Zhao Guo-Zhong(赵国忠)[b)]

[a)] *School of Sciences, Liaoning ShiHua University, Fushun* 113001, *China*

[b)] *Laboratory of Computational Physics, Institute of Applied Physics and Computational Mathematics, Beijing* 100088, *China*

(Received 25 March 2011; revised manuscript received 9 June 2011)

In the current work, we extend the local discontinuous Galerkin method to a more general application system. The Burgers and coupled Burgers equations are solved by the local discontinuous Galerkin method. Numerical experiments are given to verify the efficiency and accuracy of our method. Moreover the numerical results show that the method can approximate sharp fronts accurately with minimal oscillation.

Keywords: local discontinuous Galerkin method, Burgers equation, coupled Burgers equation

PACS: 02.70.Dh, 05.45.Pq **DOI:** 10.1088/1674-1056/20/11/110205

1. Introduction

In this paper, the purpose is to apply the local discontinuous Galerkin (LDG) method to the Burgers equation

$$u_t + uu_x - \nu u_{xx} = 0, \tag{1}$$

and coupled Burgers equations:

$$\begin{aligned} u_t - u_{xx} + \gamma u u_x + \alpha(uv)_x = 0,\\ v_t - v_{xx} + \gamma v v_x + \beta(uv)_x = 0, \end{aligned} \tag{2}$$

$x \in [a, b]$, $t \in [0, T]$, where ν and γ are constants, α and β are arbitrary constants depending on the Peclet number, the Stokes velocity of particles due to gravity and Brownian diffusivity. The Burgers equation was first introduced by Bateman[1] when he mentioned it as worthy of study and gave its steady solutions. It was later referred to as the Burgers equation after Burger[2] introduced this equation as a mathematical model for fluid flow. Equation (1) has found applications in field as diverse as gas dynamics, heat conduction, elasticity, continuous stochastic processes, etc. The second coupled system (2), derived by Esipov,[3] is a simple model of sedimentation or evolution of scaled volume concentrations of two kinds of particles in fluid suspensions or colloids, under the effect of gravity.[4] Various powerful mathematical methods such as Cole-Hopf transformation,[5] tanh-function method,[6] variational iteration method,[7] and other methods[8−10] have been used to solve Eqs. (1) and (2) analytically.

Simulation of the Burgers equation is a natural first step towards developing methods for the computation of complex flows. Many researchers have developed various numerical schemes for solving the Burgers equation to test their algorithm. These numerical methods include implicit methods,[11] the boundary element method,[12] the finite element methods,[13] local Crank–Nicolson method[14] and the lattice Boltzmann method.[15] In addition, Khater *et al.*[16] used the Chebyshev spectral collocation method and M. Dehghan *et al.*[17] introduced the Adomian–Pade technique for solving the coupled Burgers equations. More recently Mittal[18] presented a cubic B-spline collocation scheme in order to solve Eqs. (2).

Cockburn and Shu introduced the local discontinuous Galerkin (LDG) method[19] for solving nonlinear convection diffusion equation in 1998. The idea of the LDG method is to rewrite the equation with second order derivative into systems of first-order equations. The systems are then discretized by using the DG space discretization technique to obtain ordinary differential equations (ODEs) system. We then use total variation diminishing (TVD) high order Runge–Kutta methods[[20] to solve the ODEs. Some of the more recent applications of LDG method for higher order nonlinear partial differential equations can be found in Refs. [21]–[23]. Zhao have transformed the Burgers

*Project supported by the National Natural Science Foundation of China (Grant No. 11171038).

†Corresponding author. E-mail: rongpeizhang@163.com

© 2011 Chinese Physical Society and IOP Publishing Ltd

equation into linear diffusion equation by Hopf–Cole transformation and then solved it by LDG method.[24] In this paper the LDG method is directly applied to Burgers and coupled Burgers equations.

The rest of the paper is organized as follows: In Section 2, we describe the proposed LDG method. The numerical results are presented and discussed in Section 3. Finally the conclusion is given in Section 4.

2. The LDG method

In this section,we will illustrate the algorithm formulation of the LDG method for Eqs. (1) and (2) in detail.

2.1. Burgers equation

To define LDG method, we introduce an auxiliary variable, $p = \sqrt{\nu} u_x$, and rewrite Eq. (1) as the system

$$u_t + \left(\frac{1}{2}u^2\right)_x - \sqrt{\nu} p_x = 0, \tag{3}$$

$$p - \sqrt{\nu} u_x = 0. \tag{4}$$

The computational domain $[a, b]$ is divided into N cells as follows:

$$a = x_{\frac{1}{2}} < x_{\frac{3}{2}} < \cdots < x_{N+\frac{1}{2}} = b. \tag{5}$$

We denote the cell by $I_j = [x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$. Again denote the size of each cell as $h_j = x_{j+\frac{1}{2}} - x_{j-\frac{1}{2}}$ and $h = \max\limits_j h_j$. The finite element space consisting of piecewise polynomials is defined as

$$V_h^k = \{v : v|_{I_j} \in P^k(I_j), j = 1, \ldots, N\}. \tag{6}$$

where $P^k(I_j)$ denotes the set of all polynomials of degree up to k on I_j. The numerical solution in the space V_h^k can be written as follows: $u_h(x,t) = \sum\limits_{l=0}^{k} u_j^l(t) v_1^j(x) = \boldsymbol{V}^{\mathrm{T}}(x)\boldsymbol{U}_j(t)$, $\boldsymbol{U}_j(t) = (u_j^0, u_j^1, \ldots, u_j^k)^{\mathrm{T}}$ denotes the degree of freedom in cell I_j. The basis functions in $P^k(I_j)$ are defined here by scaled Legendre polynomials. The $u^+_{j+\frac{1}{2}}$ and $u^-_{j+\frac{1}{2}}$ are values of u at $x_{j+\frac{1}{2}}$ from the right cell, I_{j+1}, and the left cell, I_j, respectively,

$$u^+_{j+\frac{1}{2}} = \lim_{\epsilon \to 0+} u(x_{j+\frac{1}{2}} + \epsilon), \quad u^-_{j+\frac{1}{2}} = \lim_{\epsilon \to 0+} u(x_{j+\frac{1}{2}} - \epsilon).$$

The semi-discrete LDG method for solving the system (3) and (4) is defined as follows: find $u_h, p_h \in V_h^k$ such that, for all test functions $v, q \in V_h^k$ and all $1 \le j \le N$, we have

$$\begin{aligned} \int_{I_j} (u_h)_t v \,\mathrm{d}x \; - & \int_{I_j} \left(\frac{1}{2}u_h^2 - \sqrt{\nu} p_h\right) v_x \,\mathrm{d}x \\ & + (\hat{f} - \sqrt{\nu}\hat{p}_h)_{j+\frac{1}{2}} v^-_{j+\frac{1}{2}} \\ & - (\hat{f} - \sqrt{\nu}\hat{p}_h)_{j-\frac{1}{2}} v^+_{j-\frac{1}{2}} = 0, \end{aligned} \tag{7}$$

$$\begin{aligned} \int_{I_j} p_h q \,\mathrm{d}x \; + & \int_{I_j} \sqrt{\nu} u_h q_x \,\mathrm{d}x - (\sqrt{\nu}\hat{u}_h q^-)_{j+\frac{1}{2}} \\ & + (\sqrt{\nu}\hat{u}_h q^+)_{j-\frac{1}{2}} = 0 \end{aligned} \tag{8}$$

where $f = \frac{1}{2}u_h^2$. The "hat" terms are numerical fluxes, which are single-valued functions defined at the cell interfaces and in general depends on the values from both sides of the interface. There are two types of numerical fluxes to be defined. One type is the diffusion numerical fluxes, $\hat{u}_h$, $\hat{p}_h$, and the other one is convection numerical flux $\hat{f}$. The convection flux $\hat{f}$ is chosen to be local Lax–Friedrichs flux[25]

$$\begin{aligned} &\hat{f} = \frac{1}{2}(f(u^-) + f(u^+) - \alpha(u^+ - u^-)), \\ &\alpha = \max_u |f'(u)|. \end{aligned} \tag{9}$$

where the maximum is taken over a relevant range of u. The diffusion fluxes $\hat{u}$, $\hat{p}$ can be chosen as alternating fluxes[25]

$$\hat{u}_h = u_h^+, \quad \hat{p}_h = p_h^-. \tag{10}$$

The choices

$$\hat{u}_h = u_h^-, \quad \hat{p}_h = p_h^+ \tag{11}$$

are also fine. Note that we can solve p_h explicitly and locally in terms of u_h from Eq. (8) with numerical fluxes (9) and (10). This is why the method is referred to as "local" discontinuous Galerkin method.

2.2. Coupled Burgers equation

We introduce the auxiliary variables, $p = u_x$, $q = v_x$ and rewrite the coupled system (2) into a first order system as follows:

$$u_t + \phi(u,v)_x - p_x = 0, \tag{12}$$

$$p - u_x = 0, \tag{13}$$

$$v_t + \varphi(u,v)_x - q_x = 0, \tag{14}$$

$$q - v_x = 0, \tag{15}$$

where $\phi(u,v) = \frac{\gamma}{2}u^2 + \alpha uv$, $\varphi(u,v) = \frac{\gamma}{2}v^2 + \beta uv$. The mesh and finite element space are still given by Eqs. (5) and (6). Apply discontinuous Galerkin

method to the system: find $u, p, v, q \in V_h^k$ such that for all test functions $w, r, z, s \in V_h^k$, we have

$$\int_{I_j} u_t w \mathrm{d}x - \int_{I_j} (\phi - p) w_x \mathrm{d}x + (\hat{\phi} - \hat{p})_{j+\frac{1}{2}} w^-_{j+\frac{1}{2}} - (\hat{\phi} - \hat{p})_{j-\frac{1}{2}} w^+_{j-\frac{1}{2}} = 0, \quad (16)$$

$$\int_{I_j} p r \mathrm{d}x + \int_{I_j} u r_x \mathrm{d}x - \hat{u}_{j+\frac{1}{2}} r^-_{j+\frac{1}{2}} + \hat{u}_{j-\frac{1}{2}} r^+_{j-\frac{1}{2}} = 0, \quad (17)$$

$$\int_{I_j} v_t z \mathrm{d}x - \int_{I_j} (\varphi - q) z_x \mathrm{d}x + (\hat{\varphi} - \hat{q})_{j+\frac{1}{2}} z^-_{j+\frac{1}{2}} - (\hat{\varphi} - \hat{q})_{j-\frac{1}{2}} z^+_{j-\frac{1}{2}} = 0, \quad (18)$$

$$\int_{I_j} q s \mathrm{d}x + \int_{I_j} v s_x \mathrm{d}x - \hat{v}_{j+\frac{1}{2}} s^-_{j+\frac{1}{2}} + \hat{v}_{j-\frac{1}{2}} s^+_{j-\frac{1}{2}} = 0, \quad (19)$$

where $1 \leq j \leq N$. The "hat" terms in Eqs. (16)–(19) are the numerical fluxes. The convection numerical fluxes are defined as standard Lax–Friedrichs fluxes in the system case

$$\hat{\phi} = \frac{1}{2}(\phi^- + \phi^+ - \alpha(u^+ - u^-)),$$
$$\hat{\varphi} = \frac{1}{2}(\varphi^- + \varphi^+ - \alpha(v^+ - v^-)), \quad (20)$$

where α is the maximum eigenvalue of the Jacobian of the flux $(\phi, \varphi)^{\mathrm{T}}$. The diffusion numerical fluxes are chosen as "alternating fluxes":

$$\hat{u} = u^-,\ \hat{p}_h = p^+,\ \hat{v} = v^-,\ \hat{q}_h = q^+. \quad (21)$$

At boundary points, for Dirichlet boundary conditions, we choose

$$\hat{u}_{\frac{1}{2}} = u(a), \quad \hat{p}_{\frac{1}{2}} = p^+_{\frac{1}{2}},$$
$$\hat{v}_{\frac{1}{2}} = v(a), \quad \hat{q}_{\frac{1}{2}} = q^+_{\frac{1}{2}},$$
$$\hat{u}_{N+\frac{1}{2}} = u(b), \quad \hat{p}_{N+\frac{1}{2}} = p^-_{N+\frac{1}{2}},$$
$$\hat{v}_{N+\frac{1}{2}} = v(b), \quad \hat{q}_{N+\frac{1}{2}} = q^-_{N+\frac{1}{2}}. \quad (22)$$

The algorithm is now well defined.

2.3. Time integration

The application of LDG method generates a large coupled system of ordinary differential equations (ODEs) system

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{U}_h = L(\boldsymbol{U}_h, t). \quad (23)$$

Here $\boldsymbol{U}_h = (\boldsymbol{U}_1, \boldsymbol{U}_2, \ldots, \boldsymbol{U}_N)^{\mathrm{T}}$ are the degrees of freedom on all cells. We obtain the right-hand side of Eq. (23) by eliminating the auxiliary variable p_h in Eqs. (8). In the current work, the ODE system (23) are integrated with respect of time with consideration of low-storage TVD-RK3 scheme.[20] The TVD-RK3 scheme integrates from t to $t + \Delta t$ as follows:

$$\boldsymbol{U}_h^{(1)} = \boldsymbol{U}_h^n + \Delta t L(\boldsymbol{U}_h^n, t^n), \quad (24)$$
$$\boldsymbol{U}_h^{(2)} = \frac{3}{4}\boldsymbol{U}_h^n + \frac{1}{4}\boldsymbol{U}_h^{(1)} + \frac{1}{4}\Delta t L(\boldsymbol{U}_h^{(1)}, t^n + \Delta t), \quad (25)$$
$$\boldsymbol{U}_h^{(3)} = \frac{1}{3}\boldsymbol{U}_h^n + \frac{2}{3}\boldsymbol{U}_h^{(2)} + \frac{2}{3}\Delta t L\left(\boldsymbol{U}_h^{(2)}, t^n + \frac{1}{2}\Delta t\right). \quad (26)$$

3. Results and discussion

In this section, the solutions of the Burgers equation and coupled Burgers system will be investigated by using the LDG method. To show the efficiency of the LDG method for our problem as compared with the exact solution, we report maximum error which is defined as

$$L_\infty = \max_{1 \leq j \leq N} \{|u(x_j, t) - u_h(x_j, t)|\},$$

where $u(x_j, t)$ is the exact solution and $u_h(x_j, t)$ is the solution obtained by LDG method at the centre of cell I_j. The ratio of the errors and the order of convergence of the scheme are defined by using the formulas,

$$\text{Ratio} = \frac{\text{error}(N_1)}{\text{error}(N_2)},$$
$$\text{Order} = \frac{\log(\text{error}(N_1)/\text{error}(N_2))}{\log(N_2/N_1)},$$

where error(N_1) and error(N_2) are the L_∞ errors for the numbers of cells N and $2N$, respectively. We consider four different problems to test our methods. The first and the second problems are of the viscous Burgers equation (1). The first problem has large gradients moving with small diffusion constant. The performance of our LDG method shows its ability to resolve the large gradient regions. The third problem is of the coupled Burgers equation (2). We obtain the optimal convergence order and compare it with that of other methods. The fourth one is also of coupled Burgers equation with its solution being shock wave. We give this example to test the capability of LDG method in capturing the shock wave. All the test problems in this paper are computed by P^2 elements.

Example 1 Consider the viscous Burgers equation[26]

$$u_t + u u_x - \nu u_{xx} = 0, \quad x \in [-1, 1], \quad t > 0. \quad (27)$$

The exact solution of this problem is a shock wave,

$$u(x,t) = 1 - \tanh\left(\frac{1}{2\nu}(x-t)\right). \tag{28}$$

The initial and boundary conditions are determined by Eq. (28). The numerical result is obtained by LDG scheme and the number of partitions being 100. We plot the LDG solution and exact solution in Fig. 1 at different time levels for the values $\nu = 0.1$ and $\nu = 0.01$, respectively. Figure 1(b) shows that the steep variation is spread over only a few grids, moreover, there are none unphysical oscillations before and after the front.

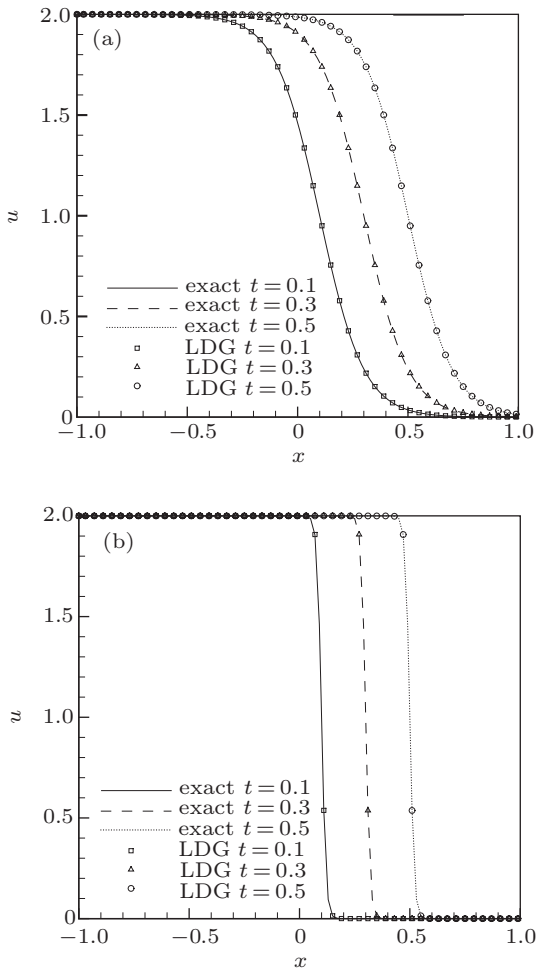


Fig. 1. Comparison between LDG solution and analytical results for Example 1 for diffusion coefficients $\nu = 0.1$ (a) and $\nu = 0.01$ (b)

Example 2 The test problem of the one-dimensional viscous Burgers equation is

$$u_t + uu_x - \nu u_{xx} = 0, \quad 0 < x < 1, t > 0 \tag{29}$$

with initial and boundary conditions

$$\begin{aligned} &u(x,0) = \sin(\pi x), \quad 0 \le x \le 1, \\ &u(0,t) = u(1,t) = 0, t > 0. \end{aligned} \tag{30}$$

The exact solution of this equation is in the form of the infinite series[14]

$$u(x,t) = 4\pi\nu\frac{C}{D+E}, \tag{31}$$

where

$$C = \sum_{n=1}^{\infty} n\,\mathrm{I}_n\left(\frac{1}{2\pi\nu}\right)\sin(n\pi x)\exp(-n^2\pi^2\nu t),$$

$$D = 2\sum_{n=1}^{\infty} \mathrm{I}_n\left(\frac{1}{2\pi\nu}\right)\cos(n\pi x)\exp(-n^2\pi^2\nu t),$$

$E = \mathrm{I}_0(1/2\pi\nu)$, and $\mathrm{I}_n(x)$ is the first type of the n-th modified Bessel function. When $n = 50$, it is used as an approximation to the infinite sum (31). The LDG scheme is chosen as P^2 element and the interval is divided into 25 cells equally. Comparisons are made with exact solution and several existing numerical schemes, which are fully implicit finite difference method (IFDM),[11] a mixed finite difference and boundary element method (BEM).[12] The numerical results are presented in Table 1 for different times and different viscosity coefficients. It is clear that the LDG method gives more accurate solution.

Table 1. Comparison of the numerical results in Example 2 for $\nu = 0.1$ and $\nu = 0.01$.

$\nu = 0.1$						
Method	t	$x = 0.1$	$x = 0.3$	$x = 0.5$	$x = 0.7$	$x = 0.9$
IFDM[11]	2.0	0.02909	0.08044	0.10939	0.09838	0.04037
BEM[12]		0.02913	0.07951	0.10770	0.09663	0.03976
LDG		0.02876	0.07945	0.10789	0.09685	0.03969
Exact		0.02876	0.07946	0.10789	0.09685	0.03969
$\nu = 0.01$						
Method	t	$x = 0.1$	$x = 0.3$	$x = 0.5$	$x = 0.7$	$x = 0.9$
IFDM[11]	2.0	0.04367	0.13095	0.21800	0.30466	0.38024
BEM[12]		0.04300	0.12877	0.21468	0.30075	0.37452
LDG		0.04297	0.12885	0.21457	0.30002	0.37355
Exact		0.04296	0.12884	0.21456	0.30000	0.37328

Example 3 We consider the coupled Burgers equations with $\alpha = 1$, $\beta = 1$, $\eta = -2$,

$$\begin{aligned} &u_t - u_{xx} - 2uu_x + (uv)_x = 0, \\ &v_t - v_{xx} - 2vv_x + (uv)_x = 0. \end{aligned} \tag{32}$$

where $x \in [-\pi, \pi]$, in the initial condition $u(x,0) = v(x,0) = \sin(x)$. The exact solutions of Eqs. (32) are given in Ref. [18] as $u(x,t) = v(x,t) = \mathrm{e}^{-t}\sin(x)$. The LDG scheme is chosen as P^2 element. We computed the maximum error for different numbers of cells and obtain the order of convergence shown in Table 2. The results are compared with other numerical results in Ref. [18]. We can see that the LDG scheme provides much more accurate solutions.

Table 2. Maximum error and order of convergence of the method for $u(x,t)$ in Example 3 at $t = 0.5$.

	Ref. [18]			LDG		
N	L_∞	Ratio	Order	L_∞	Ratio	Order
32	9.7487×10^{-4}	–	–	1.1058×10^{-4}	–	–
64	2.4361×10^{-4}	4.0014	2.005	1.3789×10^{-5}	8.0194	3.003
128	6.0896×10^{-5}	4.0004	2.001	1.7224×10^{-6}	8.0057	3.001
256	1.5223×10^{-5}	4.0003	2.001	2.1527×10^{-7}	8.0011	3.000
512	3.8052×10^{-6}	4.0006	2.002	2.6907×10^{-8}	8.0005	3.000

Example 4 We finally give the LDG solutions of the coupled Burgers equation[26]

$$u_t - u_{xx} - 2uu_x + \frac{5}{2}(uv)_x = 0,$$
$$v_t - v_{xx} - 2vv_x + \frac{5}{2}(uv)_x = 0 \qquad (33)$$

with the initial conditions $u(x,0) = v(x,0) = \lambda(1 - \tanh(\frac{3}{2}\lambda x))$, where λ is an arbitrary constant. The exact solutions of this coupled system is

$$u(x,t) = v(x,t) = \lambda\left[1 - \tanh\left(\frac{3}{2}\lambda(x - 3\lambda t)\right)\right].$$

Results are calculated by considering the domain as $x \in [-20, 20]$, and the boundary conditions are determined by exact solution. We compute the system (33) for different λ with 100 cells. The curves of numerical solution and analytical solution of $u(x,t)$ are plotted in Fig. 2 for $\lambda = 1$ at $t = 1,\ 3,\ 5$ and for $\lambda = 10$ at $t = 0.1,\ 0.3,\ 0.5$, respectively. It can be shown that the numerical solution by using LDG method is in good agreement with the exact profile for steep solution profiles.

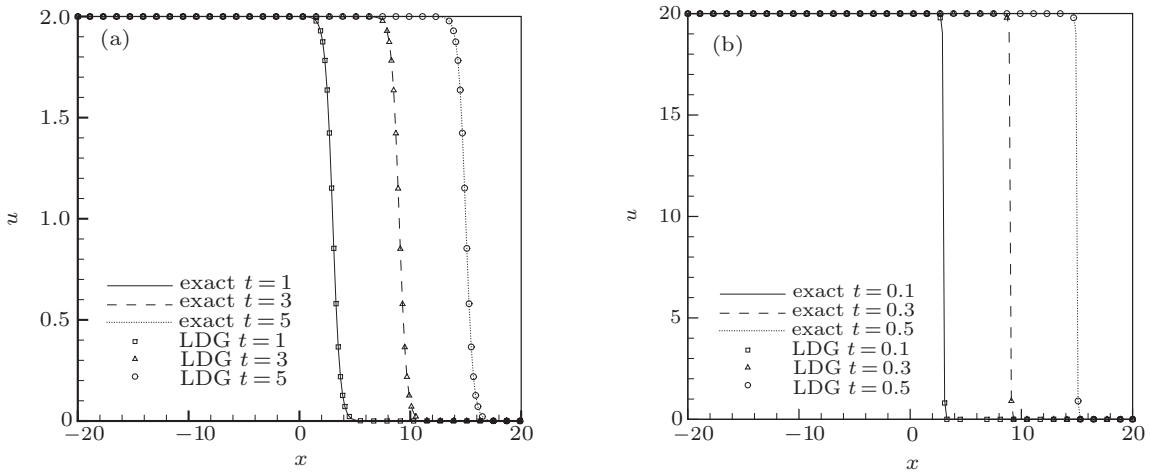


Fig. 2. Comparison between LDG solution and analytical results for Example 4 for $\lambda = 1$ (a) and $\lambda = 10$ (b).

4. Conclusion

In this paper the LDG method was proposed to solve the Burgers equation and coupled Burgers equation. We compute Eqs. (1) and (2) for various parameters by LDG method. The test examples show that the LDG scheme can give more accurate results than the other numerical schemes. Moreover, the LDG scheme is proved to be successful to approximate shock wave solutions.

References

[1] Bateman H 1915 *Mon. Weather. Rev.* **43** 163
[2] Burgers J M 1948 *Adv. Appl. Mech.* **1** 171
[3] Esipov S E 1995 *Phys. Rev.* E **52** 3711
[4] Nee J and Duan J 1998 *Appl. Math. Lett.* **11** 57
[5] Taku O 2009 *Appl. Comput. Math.* **8** 107
[6] Soliman A A 2006 *Physica* A **361** 394
[7] Abdou M A and Soliman A A 2005 *J. Comput. Appl. Math.* **181** 245
[8] Lian Z J, Chen L L and Lou S Y 2005 *Chin. Phys.* **14** 1486

[9] Chen J B, Geng X G and Qiao Z J 2010 *Chin. Phys.* B **19** 090403
[10] Zuo J M and Zhang Y M 2011 *Chin. Phys.* B **20** 010205
[11] Bahadir A R 1999 *Int. J. Appl. Math.* **8** 897
[12] Bahadir A R and Saglam M 2005 *Appl. Math. Comput.* **160** 663
[13] Caldwell J, Wanless P and Cook A E 1981 *Appl. Math. Model* **5** 189
[14] Huang P Z and Abduwali A 2010 *Comput. Math. Appl.* **59** 2452
[15] Yu X M and Shi B C 2006 *Chin. Phys.* **15** 1009
[16] Khater A H, Temsah R S and Hassan M M 2008 *J. Comput. Appl. Math.* **222** 333
[17] Dehghan M, Hamidi A and Shakourifar M 2007 *Appl. Math. Comput.* **189** 1034
[18] Mittal R C and Arora G 2011 *Commun. Nonlinear Sci. Numer. Simul.* **16** 1304
[19] Cockburn B and Shu C W 1998 *SIAM J. Numer. Anal.* **35** 2440
[20] Shu C W and Osher S 1989 *J. Comput. Phys.* **83** 32
[21] Xu Y and Shu C W 2005 *J. Comput. Phys.* **205** 72
[22] Xu Y and Shu C W 2006 *Comput. Method Appl. Mech. Eng.* **195** 3430
[23] Yan J and Shu C W 2002 *J. Sci. Comput.* **17** 27
[24] Zhao G Z, Yu X J and Wu D 2010 *Appl. Math. Comput.* **216** 3671
[25] Li B Q 2006 *Discontinuous Finite Elements in Fluid Dynamics and Heat Transfer* (Berlin: Springer)
[26] Abazari R Borhanifar A 2010 *Comput. Math. Appl.* **59** 2711