INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids (2011)
Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/fld.2612

A high order spectral volume solution to the Burgers' equation using the Hopf–Cole transformation

Ravi Kannan [1,*,†] and Z. J. Wang [2]

[1] *CFD Research Corporation, Huntsville, AL, USA*
[2] *Iowa State University, Ames, IA, USA*

SUMMARY

A limiter free high order spectral volume (SV) formulation is proposed in this paper to solve the Burgers' equation. This approach uses the Hopf–Cole transformation, which maps the Burgers' equation to a linear diffusion equation. This diffusion equation is solved in an SV setting. The local discontinuous Galerkin (LDG) and the LDG2 viscous flux discretization methods were employed. An inverse transformation was used to obtain the numerical solution to the Burgers' equation. This procedure has two advantages: (i) the shock can be captured, without the use of a limiter; and (ii) the effects of SV partitioning becomes almost redundant as the transformed equation is not hyperbolic. Numerical studies were performed to verify. These studies also demonstrated (i) high order accuracy of the scheme even for very low viscosity; (ii) superiority of the LDG2 scheme, when compared with the LDG scheme. In general, the numerical results are very promising and indicate that this procedure can be applied for obtaining high order numerical solutions to other nonlinear partial differential equations (for instance, the Korteweg–de Vries equations) which generate discontinuous solutions. Copyright © 2011 John Wiley & Sons, Ltd.

Received 24 December 2010; Revised 14 April 2011; Accepted 15 April 2011

KEY WORDS: Burgers' equation; spectral volume; high-order; LDG2; LDG; Hopf–Cole transformation

1. INTRODUCTION

The spectral volume (SV) method is a high order method, originally developed by Wang, Liu, and their collaborators for hyperbolic conservation laws on unstructured grids [1–6]. The SV method can be viewed as an extension of the Godunov method to higher order by adding more degrees of freedom (DOFs) in the form of subcells in each cell (simplex). The simplex is referred to as an SV, and the subcells are referred to as control volumes (CV). All the SVs are partitioned in a geometrically similar manner in a simplex, and thus a single reconstruction is obtained. The DOFs are then updated to high-order accuracy using the usual Godunov method.

The SV method shares many similar properties with the discontinuous Galerkin (DG) [7] and the spectral difference [8] methods, such as discontinuous solution space, sharing multiple DOFs within a single element and compactness. They mainly differ on how the DOFs are chosen and updated. Because the DG is a derivative of the finite element method, most implementations use the elemental nodal values as DOF, though some researchers use the equally valid modal approaches. Although both of the approaches are mathematically identical, at least for linear equations, different choices of DOFs are used by various researchers result in different efficiency and numerical properties. The SV being a derivative of the finite volume has subcell averages as its DOF, whereas the spectral difference has point wise values as DOF. In terms of complexity, DG requires both volume and surface

*Correspondence to: Ravi Kannan, CFD Research Corporation, 215 Wynn Drive, Huntsville, AL 35805, USA.
†E-mail: sunshekar@gmail.com

integrations. In contrast, SV requires only surface integrations and the spectral difference requires differentiations.

The SV method was successfully implemented for 2D Euler [4] and 3D Maxwell equations [6]. Recently, Sun *et al* [9] implemented the SV method for the Navier–Stokes equations using the LDG [7] approach to discretize the viscous fluxes. Kannan and Wang [10] conducted some Fourier analysis for a variety of viscous flux formulations. Kannan implemented the SV method for the Navier–Stokes equations using the LDG2 (which is an improvised variant of the LDG approach) [11] and direct discontinuous Galerkin approaches [12]. Even more recently, Kannan extended the SV method to solve the moment models in semiconductor device simulations [13–15]. The latest SV developments include incorporating the moving body method [16], discretizing the higher spatial derivative terms [17] and solving the elastohydrodynamic problem [18]. These past studies have demonstrated the efficacy of the SV method for a wide range of engineering applications and have established its robustness.

In this paper, we propose a limiter free high order SV formulation that is proposed in this paper to solve the Burgers' equation. Burgers' equation is an essential partial differential equation from fluid mechanics and is also used extensively in other areas of engineering such as gas dynamics, traffic flow modeling, acoustic wave propagation, and so on. Hopf [19] and Cole [20] have shown that a one-dimensional Burgers' equation can be reduced to a linear homogeneous heat equation for any initial condition. This was extended to two dimensions by Liao [21], Ohwada [22], Soliman [23], and other researchers.

In this paper, the transformed diffusion equation is solved in an SV setting. This needed a high order computation of the initial and the boundary conditions for the transformed equation. The LDG and the LDG2 viscous flux discretization methods were employed. The LDG formulation is a classical method to discretize the viscous fluxes. The LDG2 scheme was recently formulated by Kannan *et al* [11] as an improved variant of the LDG method. Because the transformed equation is smooth, there is no need to employ a limiter. The numerical solution to the Burgers' equation is obtained using an inverse transformation. Numerical experiments were performed in both 1D and 2D and they demonstrated (i) shock capturing without the use of a limiter, even for very low viscosity; (ii) high order accuracy even for very low viscosity; and (iii) the superiority of the LDG2 scheme when compared with the LDG scheme.

The paper is organized as follows. In the next section, we review the basics of the SV method. The Hopf–Cole transformation is discussed in Section 3. Section 4 presents with the different test cases conducted in this study. Finally, conclusions from this study are summarized in Section 5.

2. BASICS OF THE SPECTRAL VOLUME METHOD

2.1. General formulation

Consider the general conservation equation

$$\frac{\partial Q}{\partial t} + \frac{\partial (f_i(Q))}{\partial x} + \frac{\partial (g_i(Q))}{\partial y} = 0, \tag{2.1}$$

in domain Ω with appropriate initial and boundary conditions. In (2.1), x and y are the Cartesian coordinates and (x,y) $\in \Omega$, t $\in$ [0,T] denotes time, Q is the vector of conserved variables, and f_i and g_i are the inviscid fluxes in the x and y directions, respectively. Domain Ω is discretized into I nonoverlapping triangular (2D) cells. In the SV method, the simplex grid cells are called SVs, denoted S_i, which are further partitioned into CVs, denoted C_{ij}, which depend on the degree of the polynomial reconstruction. Figure 1 shows linear, quadratic, and cubic partitions in 1D. Figure 2 shows the same in 2D.

We need N unknown control volume solution averages (or DOFs) to construct a degree k polynomial. N is calculated using the formula (in 2D)

$$N = \frac{(k+1)(k+2)}{2}, \tag{2.2}$$

DOI: 10.1002/fld

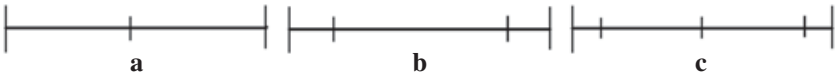

Figure 1. Partitions of a spectral volume in 1D. Linear, quadratic, and cubic reconstructions are shown in a, b, and c, respectively.

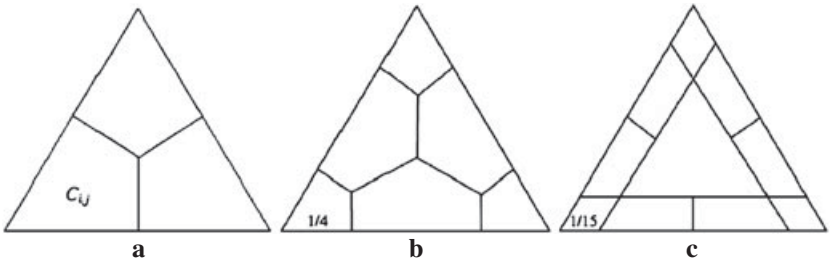

Figure 2. Partitions of a triangular spectral volume. Linear, quadratic, and cubic reconstructions are shown in a, b, and c, respectively.

where k is the degrees of the polynomial, constructed using the CV solution averages. The CV averaged conserved variable for C_{ij} is defined as

$$\overline{Q}_{i,j} = \frac{1}{V_{i,j}} \int_{C_{i,j}} Q dV, j = 1 \ldots N, i = 1 \ldots I, \tag{2.3}$$

where $V_{i,j}$ is the volume of C_{ij}. Given the CV averaged conserved variables, a degree k polynomial can be constructed such that it is $(k+1)^{th}$ order approximation to Q. In other words, we can write the polynomial as

$$p_i(x, y) = \sum_{j=1}^{N} L_j(x, y) \overline{Q}_{i,j}, \tag{2.4}$$

where the shape functions L_j (x,y) satisfy

$$\frac{1}{V_{i,j}} \int_{C_{i,j}} L_n(x, y) dV = \delta_{j,n}. \tag{2.5}$$

Equation 2.1 is integrated over the C_{ij}. This results in the following equation

$$\frac{\partial \overline{Q}}{\partial t} + \frac{1}{V_{i,j}} \sum_{r=1}^{K} \int_{Ar} (\vec{F}.\vec{n}) dA = 0, \tag{2.6}$$

where $\vec{F} = (f_i - f_v, g_i - g_v)$, where A_r represents the r^{th} face of C_{ij}, $\vec{n}$ is the outward unit normal vector of A_r, and K is the number of faces in C_{ij}. The surface integration on each face is done using a $(k+1)^{th}$ order accurate Gauss quadrature formula. The fluxes are discontinuous across the SV interfaces. The inviscid fluxes are handled using a numerical Riemann flux such as the Rusanov flux [24], the Roe flux [25] or AUSM flux [26]. The handling of the viscous fluxes is discussed in the next section.

2.2. Spectral volume formulation for the diffusion equation

The following diffusion equation is considered first in domain Ω with appropriate initial and boundary conditions

$$\frac{\partial u}{\partial t} - \nabla \cdot (\mu \nabla u) = 0, \tag{2.7}$$

DOI: 10.1002/fld

where μ is a positive diffusion coefficient. We define an auxiliary variable

$$\vec{q} = \nabla u. \tag{2.8}$$

Equation 2.7 then becomes

$$\frac{\partial u}{\partial t} - \nabla \cdot (\mu \vec{q}) = 0. \tag{2.9}$$

Using the Gauss–divergence theorem, we obtain

$$\overline{\vec{q}}_{ij} V_{ij} = \sum_{r=1}^{K} \int_{Ar} u \cdot \vec{n} dA, \tag{2.10}$$

$$\frac{d\overline{u}_{ij}}{dt} V_{ij} - \sum_{r=1}^{K} \int_{A_r} \mu \vec{q} \cdot \vec{n} dA = 0, \tag{2.11}$$

where $\overline{\vec{q}}_{ij}$ and $\overline{u}_{ij}$ are the CV averaged gradient and solution in C_{ij}. As the solution u is cell-wise continuous, u and $\vec{q}$ at SV boundaries are replaced by numerical fluxes $\underline{\vec{q}}$ and $\underline{u}$. The previous equations thus become

$$\overline{\vec{q}}_{ij} V_{ij} = \sum_{r=1}^{K} \int_{Ar} \underline{u} \cdot \vec{n} dA, \tag{2.12}$$

$$\frac{d\overline{u}_{ij}}{dt} V_{ij} - \sum_{r=1}^{K} \int \mu \underline{\vec{q}} \cdot \vec{n} dA = 0. \tag{2.13}$$

2.2.1. LDG approach The commonly used approach for obtaining the numerical fluxes is the LDG approach. In this approach, the numerical fluxes are defined by alternating the direction in the following manner [7, 15]

$$\underline{u} = u_L, \tag{2.14}$$

$$\underline{\vec{q}} = \vec{q}_R, \tag{2.15}$$

where $u_L$and u_R are the left and right state solutions of the CV face in consideration, and $\vec{q}_L$ and $\vec{q}_R$ are the left and right state solution gradients of the face (of the CV) in consideration. Thus, if the CV face lies on the SV boundary, $u_L \neq u_R$ and $\vec{q}_L \neq \vec{q}_R$.

2.2.2. LDG2 approach The LDG2 can be explained using the following notations:

1. e_{CV}: refers to the CV boundary.
2. e_{SV}: refers to the SV boundary.
3. $e_{CV1} = e_{CV} \cap e_{SV}$: refers to the intersection of the CV and SV boundaries.
4. $e_{CV2} = e_{CV}/e_{CV1}$: refers to the set theoretic complement of e_{CV} and e_{CV1} (i.e. set of all surfaces present in e_{CV} but not in e_{CV1}).
5. $u_A = \frac{(u_L + u_R)}{2}$: is the average of the left and right state solutions.

The crux of the LDG2 approach is maintaining two gradients for the residual computation:

1. The first gradient is the right-sided gradient ($\overrightarrow{qr}$). The CV averaged $\overrightarrow{qr}$ is computed using u_R.
2. The second gradient is the averaged gradient ($\overrightarrow{qa}$). The CV averaged $\overrightarrow{qa}$ is computed using u_A.

DOI: 10.1002/fld

Thus, for a given SV, the CV averaged values of $\overrightarrow{qa}$ depend only on the CV averaged solution of the current SV and its closest neighbors. The CV averaged values of the $\overrightarrow{qr}$ depend only on the CV averaged solution of the current SV and the SV to its right. Given the CV averages, reconstructions are performed to obtain $\overrightarrow{qr}_L$, $\overrightarrow{qr}_R$, $\overrightarrow{qa}_L$, $\overrightarrow{qa}_R$. $\overrightarrow{qr}_L$ is used for computing the viscous fluxes through the CV faces lying on e_{CV1}. Either $\overrightarrow{qa}_L$ or $\overrightarrow{qa}_R$ can be used for computing the viscous fluxes through the CV faces lying on e_{CV2}. This procedure ensures compactness.

The LDG2 has the following desirable attributes:

1. Still is compact
2. Still avoids the need for penalizing terms. Obtaining an accurate length scale for the penalizing term can be cumbersome in higher dimensions, especially for unstructured and nonuniform grids.
3. Clearly has a higher degree of symmetry than the original LDG: as all the internal CV faces utilize averaged gradients.

Some of the advantages mentioned were demonstrated earlier by Kannan and Wang [11] for aerodynamic applications. These can also be used for other areas involving high order viscous fluxes like turbulent explosions [27, 28] and multiphase flows [27–29] and are currently being investigated by researchers for the aforementioned areas.

3. THE HOPF–COLE TRANSFORMATION

In this section, we briefly describe the Hopf–Cole transformation for obtaining the numerical solution to the Burgers' equation.

3.1. Transformation in 1D

Consider the one dimensional Burgers' equation:

$$\frac{\partial u}{\partial t} + uu_x = \mu u_{xx},\ a < x < b, \tag{3.1}$$

with the initial condition: $u(x,0) = \phi(x)$ and the boundary conditions: $u(a,t) = f(t))$, $u(b,t) = g(t))$.
Using the Hopf–Cole [19, 20] transformation,

$$u(x,t) = -2\mu\frac{\theta_x}{\theta}, \tag{3.2}$$

Equation (3.1) is transformed into the linear heat conduction equation:

$$\frac{\partial \theta}{\partial t} = \mu\theta_{xx},\ a < x < b,\ t > 0. \tag{3.3}$$

The initial condition for equation (3.2) is

$$\theta(x,0) = \exp\left(-\frac{\int_0^x u(y,0)dy}{2\mu}\right). \tag{3.4}$$

The boundary conditions for equation (3.2) are $\theta_x(a,t) = \theta_x(b,t) = 0$. Thus, the final equation is a linear heat conduction equation, which can be solved using the high order SV method, with different viscous flux discretization techniques. The final equation, being very smooth, does not require any limiters to satisfy the maximum principle rule.

DOI: 10.1002/fld

3.2. Transformation in 2D

Consider the following system of coupled two-dimensional Burgers' equations:

$$\left.\begin{array}{l} \dfrac{\partial u}{\partial t} + uu_x + vu_y = \mu(u_{xx} + u_{yy}) \\ \dfrac{\partial v}{\partial t} + uv_x + vv_y = \mu(v_{xx} + v_{yy}) \end{array}\right\}, (x, y) \in D, \tag{3.5}$$

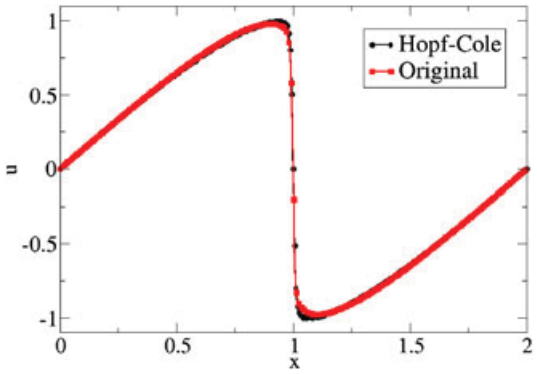

Figure 3. Numerical solution of the second order 1D Burgers' equation, using $\mu = 0.003$ at $T = 0.42$ seconds.

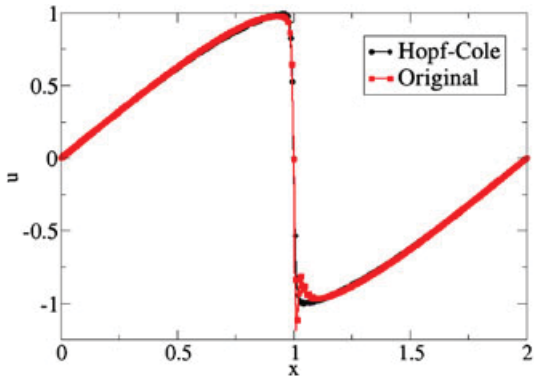

Figure 4. Numerical solution of the second order 1D Burgers' equation, using $\mu = 0.003$ at $T = 0.45$ seconds.

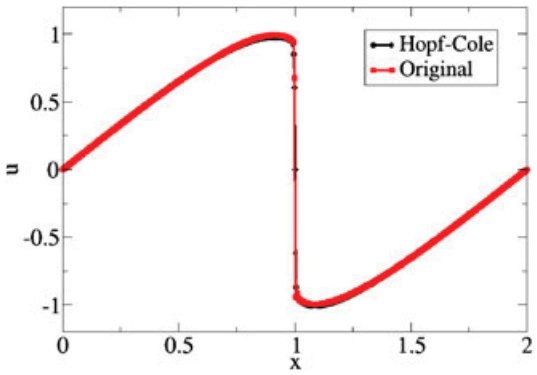

Figure 5. Numerical solution of the third order 1D Burgers' equation, using $\mu = 0.0015$ at $T = 0.42$ seconds.

DOI: 10.1002/fld

with the initial conditions: $u(x,y,0) = u_0(x,y)$,$v(x,y,0) = v_0(x,y)$ and the boundary conditions:

$$\begin{matrix} u(x,y,t) = f(x,y,t) & (x,y,t) \in \partial D \\ v(x,y,t) = g(x,y,t) & (x,y,t) \in \partial D \end{matrix}. \tag{3.6}$$

If we choose a function $\phi(x,y,t)$ such that

$$u(x,y,t) = -2\mu\frac{\phi_x}{\phi},\ v(x,y,t) = -2\mu\frac{\phi_y}{\phi}, \tag{3.7}$$

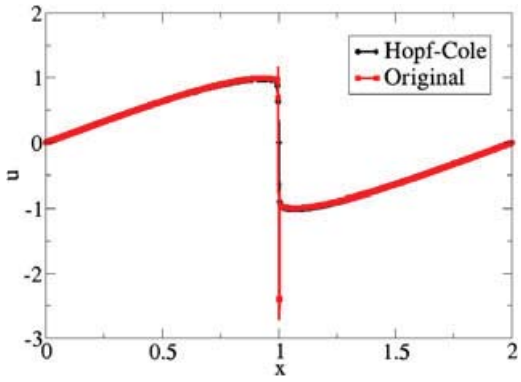

Figure 6. Numerical solution of the third order 1D Burgers' equation, using $\mu = 0.0015$ at $T = 0.4365$ seconds.

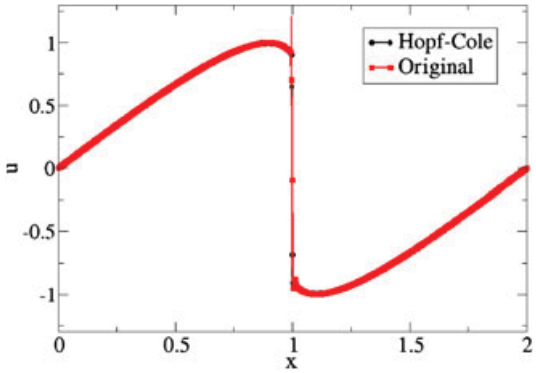

Figure 7. Numerical solution of the fourth order 1D Burgers' equation, using $\mu = 0.0005$ at $T = 0.40$ seconds.

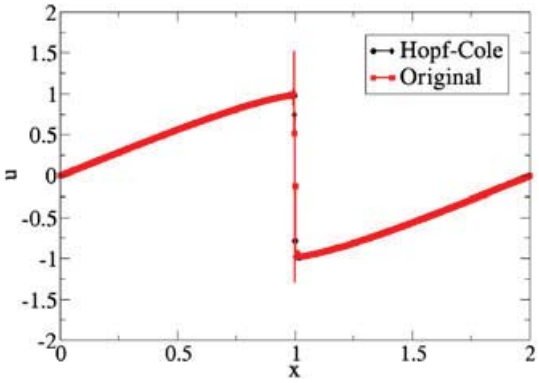

Figure 8. Numerical solution of the fourth order 1D Burgers' equation, using $\mu = 0.0005$ at $T = 0.50$ seconds.

DOI: 10.1002/fld

the transformed equation would be a heat conduction equation in 2D:

$$\frac{\partial \phi}{\partial t} = \mu(\phi_{xx} + \phi_{yy}), \tag{3.8}$$

with the initial conditions (IC) and boundary conditions (BC) for ϕ obtained using the ICs and BCs of u and v.
It must be noted that the following symmetry condition needs to be valid for the transformation [21]:

$$u_y = v_x \tag{3.9}$$

3.2.1. Initial conditions in 2D In this subsection, we outline a procedure to obtaining an initial condition for $\phi(x, y)$. We borrow ideas from Liao *et al* [21], who had charted the procedure for finite difference simulations.

Table I. Accuracy study for the 1D Burgers' equation using the second order spectral volume formulation.

Method	μ	#N	L_1 error	L_1 order	L_∞ error	L_∞ order
LDG	2e-3	125	0.0284	–	0.143	–
		250	0.00979	1.54	0.0505	1.50
		500	2.74e-03	1.84	0.0140	1.85
		1000	7.08e-04	1.95	3.58e-03	1.97
		2000	1.79e-04	1.98	8.96e-04	2.00
		4000	4.57e-05	1.99	2.24e-04	2.00
LDG2	2e-3	125	0.0221	–	0.112	–
		250	0.00710	1.64	0.0366	1.61
		500	1.93e-03	1.88	9.85e-03	1.89
		1000	4.93e-04	1.97	2.49e-03	1.98
		2000	1.25e-04	1.98	6.23e-04	2.00
		4000	3.13e-05	1.99	1.56e-04	2.00
LDG	5e-3	125	8.97e-03	–	0.0429	–
		250	2.48e-03	1.85	0.0119	1.85
		500	6.37e-04	1.96	3.04e-03	1.97
		1000	1.60e-04	1.99	7.60e-04	2.00
		2000	4.01e-05	2.00	1.90e-04	2.00
		4000	1.00e-05	2.00	4.75e-05	2.00
LDG2	5e-3	125	6.47e-03	–	0.0312	–
		250	1.74e-03	1.89	8.37e-03	1.90
		500	4.44e-04	1.97	2.11e-03	1.99
		1000	1.11e-04	2.00	5.28e-04	2.00
		2000	2.77e-05	2.00	1.32e-04	2.00
		4000	6.94e-06	2.00	3.31e-05	2.00
LDG	1e-2	125	2.09e-03	–	8.45e-03	–
		250	5.33e-04	1.97	2.15e-03	1.97
		500	1.34e-04	1.99	5.39e-04	2.00
		1000	3.36e-05	2.00	1.35e-04	2.00
		2000	8.42e-06	2.00	3.36e-05	2.00
		4000	2.11e-06	2.00	8.42e-06	2.00
LDG2	1e-2	125	1.46e-03	–	5.93e-03	–
		250	3.66e-04	1.99	1.49e-03	1.99
		500	9.16e-05	1.99	3.72e-04	2.00
		1000	2.28e-05	2.00	9.29e-05	2.00
		2000	5.72e-06	2.00	2.32e-05	2.00
		4000	1.43e-06	2.00	5.81e-06	2.00

LDG, local discontinuous galerkin.

DOI: 10.1002/fld

The transformation given in equation (3.7) can be rewritten in the following manner:

$$\frac{\phi_x}{\phi} = \frac{u(x,y,t)}{-2\mu}. \tag{3.10}$$

Integrating equation (3.10), with respect to x results in

$$\phi(x,y,0) = \phi(0,y,0)\exp\left[\int_0^x -\frac{u(s,y,0)ds}{2\mu}\right]. \tag{3.11}$$

Equation (3.7) can also be rearranged in a different manner:

$$\frac{\phi_y}{\phi} = \frac{v(x,y,t)}{-2\mu}. \tag{3.12}$$

Integration of the previous equation with respect to y results in

$$\phi(x,y,0) = \phi(x,0,0)\exp\left[\int_0^y -\frac{v(x,s,0)ds}{2\mu}\right]. \tag{3.13}$$

Table II. Accuracy study for the 1D Burgers' equation using the third order spectral volume formulation.

Method	μ	#N	L_1 error	L_1 order	L_∞ error	L_∞ order
LDG	2e-3	125	2.57e-04	–	2.26e-03	–
		250	3.23e-05	2.99	2.84e-04	2.99
		500	4.06e-06	2.98	3.58e-05	2.99
		1000	5.79e-07	2.81	4.86e-06	2.88
		2000	8.91e-08	2.70	7.18e-07	2.76
		4000	1.75e-08	2.35	1.23e-07	2.54
LDG2	2e-3	125	1.49e-04	–	1.30e-03	–
		250	1.87e-05	2.99	1.64e-04	2.99
		500	2.38e-06	2.98	2.06e-05	2.99
		1000	3.18e-07	2.90	2.74e-06	2.91
		2000	4.66e-08	2.77	3.88e-07	2.82
		4000	8.60e-09	2.44	6.40e-08	2.60
LDG	5e-3	125	2.25e-05	–	1.79e-04	–
		250	2.82e-06	3.00	2.26e-05	2.99
		500	3.67e-07	2.94	2.91e-06	2.95
		1000	5.28e-08	2.80	4.10e-07	2.83
		2000	9.79e-09	2.43	7.01e-08	2.55
LDG2	5e-3	125	1.30e-05	–	1.02e-04	–
		250	1.63e-06	3.00	1.28e-05	2.99
		500	2.09e-07	2.96	1.65e-06	2.96
		1000	2.92e-08	2.84	2.27e-07	2.86
		2000	5.16e-09	2.50	3.74e-08	2.60
LDG	1e-2	125	1.22e-06	–	8.10e-06	–
		250	1.71e-07	2.83	1.16e-06	2.80
		500	3.00e-08	2.51	2.14e-07	2.44
		1000	6.81e-09	2.14	5.03e-08	2.09
LDG2	1e-2	125	7.09e-07	–	4.81e-06	–
		250	9.63e-08	2.88	6.58e-07	2.87
		500	1.59e-08	2.60	1.14e-07	2.53
		1000	3.39e-09	2.23	2.53e-08	2.17

LDG, local discontinuous galerkin.

DOI: 10.1002/fld

Substituting zero for x in equation (3.13) results in

$$\phi(0, y, 0) = \phi(0, 0, 0) \exp\left[\int_0^y -\frac{v(0, s, 0)ds}{2\mu}\right]. \tag{3.14}$$

Combining equations (3.11) and (3.14), we obtain the following initial condition for $\phi(x, y)$:

$$\phi(x, y, 0) = \phi(0, 0, 0) \exp\left[-\int_0^y \frac{v(0, s, 0)ds}{2\mu} - \int_0^x \frac{u(s, y, 0)ds}{2\mu}\right]. \tag{3.15}$$

In this paper, we fix $\phi(0, 0, 0) = 1$ and use a very high order numerical integration scheme (Boole's rule, which results in seventh order accuracy) to evaluate the integrals in equation (3.15).

3.2.2. Boundary conditions in 2D In this subsection, we outline a procedure to obtaining boundary conditions for $\phi(x, y)$. The assumption is that the computational domain is a square, defined by [0,b] x [0,b]. The boundaries can now be defined as $\Gamma_{BOTTOM} = (0 \leqslant x \leqslant b, y = 0)$, $\Gamma_{TOP} = (0 \leqslant x \leqslant b, y = b)$, $\Gamma_{LEFT} = (0 \leqslant y \leqslant b, x = 0)$ and $\Gamma_{RIGHT} = (0 \leqslant y \leqslant b, x = b)$. Integration of equation (3.10), results in the following BC on Γ_{BOTTOM}:

$$\phi(x, 0, t) = \phi(0, 0, t) \exp\left[\int_0^x -\frac{u(s, 0, t)ds}{2\mu}\right],\ 0 \leqslant x \leqslant b. \tag{3.16}$$

Table III. Accuracy study for the 1D Burgers' equation using the fourth order spectral volume formulation.

Method	μ	#N	L_1 error	L_1 order	L_∞ error	L_∞ order
LDG	2e-3	62	1.83e-05	–	4.09e-04	–
		124	1.17e-06	3.97	2.67e-05	3.94
		248	7.36e-08	3.99	1.68e-06	3.99
		496	4.60e-09	4.00	1.05e-07	4.00
LDG2	2e-3	62	1.21e-05	–	2.22e-04	–
		124	7.66e-07	3.98	1.43e-05	3.96
		248	4.79e-08	4.00	8.93e-07	4.00
		496	2.99e-09	4.00	5.59e-08	4.00
LDG	5e-3	62	4.53e-06	–	1.77e-05	–
		124	2.85e-07	3.99	1.12e-06	3.98
		248	1.78e-08	4.00	7.00e-08	4.00
		496	1.11e-09	4.00	4.38e-09	4.00
LDG2	5e-3	62	3.37e-06	–	1.31e-05	–
		124	2.11e-07	4.00	8.18e-07	4.00
		248	1.31e-08	4.00	5.11e-08	4.00
		496	8.24e-10	4.00	3.19e-09	4.00
LDG	1e-2	62	1.59e-06	–	5.16e-06	–
		124	9.93e-08	4.00	3.23e-07	4.00
		248	6.20e-09	4.00	2.02e-08	4.00
		496	3.88e-10	4.00	1.26e-09	4.00
LDG2	1e-2	62	1.01e-06	–	4.11e-06	–
		124	6.32e-08	4.00	2.57e-07	4.00
		248	3.95e-09	4.00	1.61e-08	4.00
		496	2.46e-10	4.00	1.00e-09	4.00

LDG, local discontinuous galerkin.

DOI: 10.1002/fld

The BC on Γ_{TOP} can be obtained, using the arguments and the derivations, which lead to equation (3.15):

$$\phi(x,b,t) = \phi(0,0,t)\exp\left[-\int_0^b \frac{v(0,s,t)ds}{2\mu} - \int_0^x \frac{u(s,b,t)ds}{2\mu}\right],\ 0 \leqslant x \leqslant b. \tag{3.17}$$

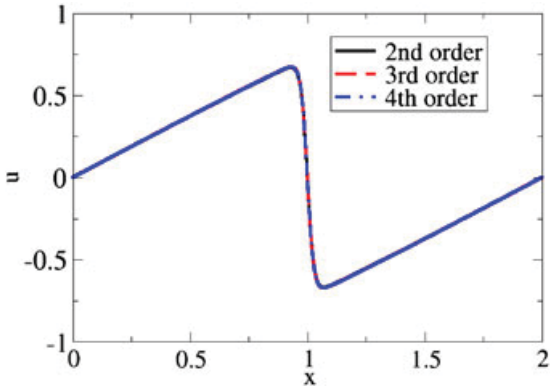

Figure 9. Numerical solution of the 1D Burgers' equation at time 1 second, using $\mu = 0.01$.

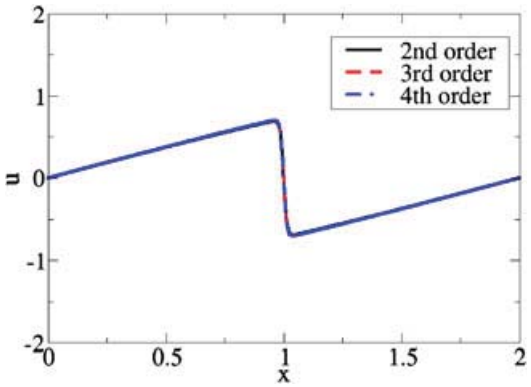

Figure 10. Numerical solution of the 1D Burgers' equation at time 1 second, using $\mu = 0.005$.

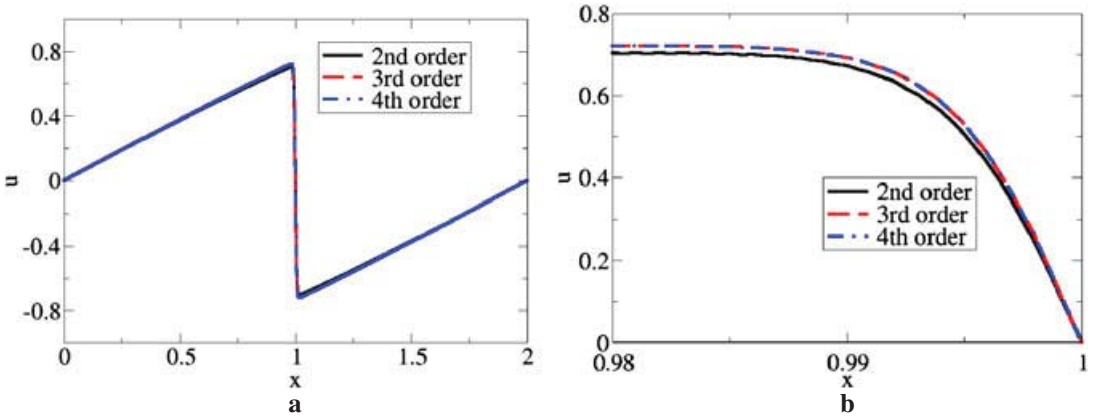

Figure 11. Numerical solution of the 1D Burgers' equation at time 1 second, using $\mu = 0.002$. Case (a): entire solution; Case(b): zoom-in of the solution near the discontinuity.

DOI: 10.1002/fld

The BCs on Γ_{LEFT} and Γ_{RIGHT} can be obtained in a similar manner:

$$\phi(0,y,t) = \phi(0,0,t)\exp\left[\int_0^y -\frac{v(0,s,t)ds}{2\mu}\right],\ 0 \leqslant y \leqslant b, \tag{3.18}$$

$$\phi(b,y,t) = \phi(0,0,t)\exp\left[-\int_0^b \frac{u(s,0,t)ds}{2\mu} - \int_0^y \frac{v(b,s,t)ds}{2\mu}\right],\ 0 \leqslant y \leqslant b. \tag{3.19}$$

Once again, the Boole's rule for numerical integration was used to numerically evaluate the previously mentioned integrals.

3.2.3. Obtaining $\phi(0,0,t)$ In this subsection, we outline a method for obtaining $\phi(0,0,t)$. Integrating equation (3.8), with respect to time, yields the equation:

$$\phi(x,y,t) = \phi(x,y,t_0) + \int_{t_0}^t \mu(\phi_{xx} + \phi_{yy})dt. \tag{3.20}$$

Differentiating the first equation of (3.7) with respect to x results in

$$\phi_{xx} = \phi\left(\frac{u^2}{4\mu^2} - \frac{u_x}{2\mu}\right). \tag{3.21}$$

Table IV. Accuracy study for the 2D Burgers' equation using the second order spectral formulation at T = 1.0 (Case 2 in section 4).

Method	μ	#N	L_1 error	L_1 order	L_∞ error	L_∞ order
LDG	0.001	10x10x2	2.14e-01	–	1.42e-00	–
		20x20x2	8.17e-02	1.39	5.85e-01	1.28
		40x40x2	2.73e-02	1.58	2.17e-01	1.43
		80x80x2	7.84e-03	1.80	6.87e-02	1.66
LDG2	0.001	10x10x2	1.72e-01	–	1.09e-00	–
		20x20x2	6.47e-02	1.41	4.40e-01	1.31
		40x40x2	2.12e-02	1.61	1.59e-01	1.47
		80x80x2	5.96e-03	1.83	4.81e-02	1.72
LDG	0.01	10x10x2	5.75e-02	–	3.90e-01	–
		20x20x2	2.06e-02	1.48	1.51e-01	1.37
		40x40x2	6.25e-03	1.72	5.17e-02	1.55
		80x80x2	1.74e-03	1.85	1.49e-02	1.79
LDG2	0.01	10x10x2	4.61e-02	–	3.02e-01	–
		20x20x2	1.61e-02	1.52	1.13e-01	1.42
		40x40x2	4.71e-03	1.77	3.77e-02	1.58
		80x80x2	1.29e-03	1.87	1.07e-02	1.82
LDG	0.1	10x10x2	1.91e-02	–	1.41e-01	–
		20x20x2	5.72e-03	1.74	4.62e-02	1.61
		40x40x2	1.55e-03	1.88	1.31e-02	1.82
		80x80x2	4.02e-04	1.95	3.48e-03	1.91
LDG2	0.1	10x10x2	1.54e-02	–	1.09e-01	–
		20x20x2	4.51e-03	1.77	3.50e-02	1.64
		40x40x2	1.21e-03	1.90	9.84e-03	1.83
		80x80x2	3.11e-04	1.96	2.60e-03	1.92

LDG, local discontinuous galerkin.

DOI: 10.1002/fld

Similarly, differentiating the second equation of (3.7) with respect to y results in

$$\phi_{yy} = \phi\left(\frac{v^2}{4\mu^2} - \frac{v_y}{2\mu}\right). \tag{3.22}$$

Combining equations (3.20), (3.21), and (3.22) and setting x and y to zeros, we obtain

$$\phi(0,0,t) = \phi(0,0,t_0) + \int_{t_0}^{t} \phi(0,0,t)\left(\frac{u(0,0,t)^2}{4\mu^2} - \frac{u(0,0,t)_x}{2\mu} + \frac{v(0,0,t)^2}{4\mu^2} - \frac{v(0,0,t)_y}{2\mu}\right) dt. \tag{3.23}$$

The Boole's rule for numerical integration was used to evaluate the previously mentioned integral. It must be noted that the Boole's rule gets activated if the number of data points is greater or equal to five. Hence, the current implementation stores four previous instances of $\phi(0,0,t)$, to solve the equation (3.23). When the number of data points is less than five, low order integration (for instance, the Simpson's rule or the trapezoidal rule) is employed. This does introduce some errors. These errors are generally very small because (i) these operations are carried out only couple of times during the entire simulation; and (ii) the time step is generally very small. In theory, it is possible to use a Richardson extrapolation to correct these errors. However, this procedure was not carried out, because the simulations yielded expected orders of accuracy.

Table V. Accuracy study for the 2D Burgers' equation using the third order spectral volume formulation at T = 1.0 (Case 2 in section 4).

Method	μ	#N	L_1 error	L_1 order	L_∞ error	L_∞ order
LDG	0.001	10x10x2	3.45e-02	–	2.46e-01	–
		20x20x2	5.20e-03	2.73	4.03e-02	2.61
		40x40x2	7.06e-04	2.88	5.71e-03	2.82
		80x80x2	1.12e-04	2.66	9.47e-04	2.59
LDG2	0.001	10x10x2	2.97e-02	–	2.04e-01	–
		20x20x2	4.35e-03	2.77	3.25e-02	2.65
		40x40x2	5.79e-04	2.91	4.48e-03	2.86
		80x80x2	9.29e-05	2.64	7.38e-04	2.60
LDG	0.01	10x10x2	9.03e-03	–	6.45e-02	–
		20x20x2	1.30e-03	2.80	9.65e-03	2.74
		40x40x2	1.99e-04	2.70	1.55e-03	2.64
		80x80x2	3.72e-05	2.42	2.87e-04	2.43
LDG2	0.01	10x10x2	7.94e-03	–	5.55e-02	–
		20x20x2	1.12e-03	2.82	8.14e-03	2.77
		40x40x2	1.72e-04	2.71	1.30e-03	2.65
		80x80x2	3.12e-05	2.46	2.32e-04	2.48
LDG	0.1	10x10x2	2.90e-03	–	2.29e-02	–
		20x20x2	4.05e-04	2.84	3.29e-03	2.80
		40x40x2	6.28e-05	2.69	5.31e-04	2.63
		80x80x2	1.33e-05	2.24	1.18e-04	2.17
LDG2	0.1	10x10x2	2.60e-03	–	2.04e-02	–
		20x20x2	3.56e-04	2.87	2.89e-03	2.82
		40x40x2	5.47e-05	2.70	4.54e-04	2.67
		80x80x2	1.18e-05	2.21	9.95e-05	2.19

LDG, local discontinuous galerkin.

DOI: 10.1002/fld

4. TEST RESULTS

In this section, we provide numerical examples to illustrate the capability of the LDG based SV formulation for solving equations containing third spatial derivative terms. A three stage SSP Runge–Kutta scheme was used for time advancement [30]:

$$\overline{u}_i^{(1)} = \overline{u}_i^n - \Delta t R_i(\overline{u}^n);$$

$$\overline{u}_i^{(2)} = \frac{3}{4}\overline{u}_i^n + \frac{1}{4}[\overline{u}_i^{(1)} - \Delta t R_i(\overline{u}^{(1)})];$$

$$\overline{u}_i^{n+1} = \frac{1}{3}\overline{u}_i^n + \frac{2}{3}[\overline{u}_i^{(2)} - \Delta t R_i(\overline{u}^{(2)})]. \tag{4.1}$$

4.1. Test case 1

We compute the steady state solution of the 1D Burgers' equation:

$$\frac{\partial u}{\partial t} + uu_x = \mu u_{xx}, 0 < x < 2, \tag{4.2}$$

Table VI. Accuracy study for the 2D Burgers' equation using the fourth order spectral volume formulation at T = 1.0 (Case 2 in section 4).

Method	μ	#N	L_1 error	L_1 order	L_∞ error	L_∞ order
LDG	0.001	10x10x2	6.29e-03	–	4.71e-02	–
		20x20x2	4.77e-04	3.72	3.86e-03	3.61
		40x40x2	3.33e-05	3.84	2.87e-04	3.75
		80x80x2	2.17e-06	3.94	1.95e-05	3.88
LDG2	0.001	10x10x2	5.99e-03	–	4.42e-02	–
		20x20x2	4.48e-04	3.74	3.57e-03	3.63
		40x40x2	3.10e-05	3.85	2.58e-04	3.79
		80x80x2	2.00e-06	3.96	1.72e-05	3.91
LDG	0.01	10x10x2	1.61e-03	–	1.21e-02	–
		20x20x2	1.18e-04	3.77	9.57e-04	3.66
		40x40x2	8.07e-06	3.87	6.87e-05	3.80
		80x80x2	5.22e-07	3.95	4.54e-06	3.92
LDG2	0.01	10x10x2	1.55e-03	–	1.13e-02	–
		20x20x2	1.13e-04	3.78	8.81e-04	3.68
		40x40x2	7.66e-06	3.88	6.29e-05	3.81
		80x80x2	4.89e-07	3.97	4.12e-06	3.93
LDG	0.1	10x10x2	5.01e-04	–	4.21e-03	–
		20x20x2	3.49e-05	3.84	3.24e-04	3.70
		40x40x2	2.32e-06	3.91	2.26e-05	3.84
		80x80x2	1.47e-07	3.98	1.47e-06	3.94
LDG2	0.1	10x10x2	4.83e-04	–	3.95e-03	–
		20x20x2	3.32e-05	3.86	3.02e-04	3.71
		40x40x2	2.20e-06	3.92	2.07e-05	3.86
		80x80x2	1.39e-07	3.98	1.35e-06	3.95

LDG, local discontinuous galerkin.

DOI: 10.1002/fld

with the initial condition: $u(x,0) = \sin(x)$ and the boundary conditions: $u(0,t) = 0$, $u(1,t) = 0$. The exact solution is given by

$$u(x,t) = \frac{2\pi\mu \sum_{n=1}^{\infty} a_n \exp(-n^2\pi^2\mu t) n \sin(n\pi x)}{a_0 + \sum_{n=1}^{\infty} a_n \exp(-n^2\pi^2\mu t)\cos(n\pi x)}, \tag{4.3}$$

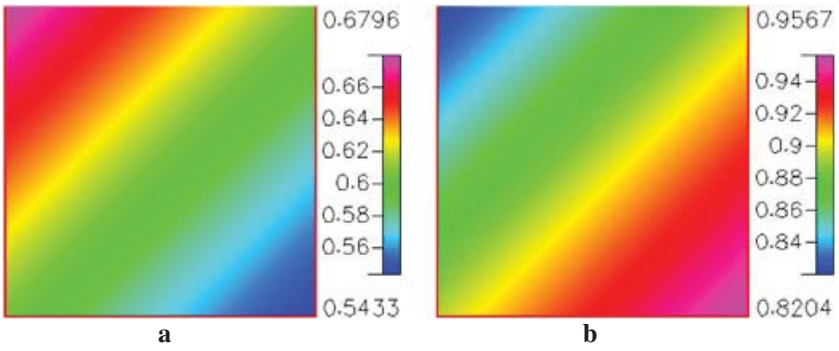

Figure 12. Fourth order solution of the 2D Burgers' equation, with $\mu = 0.1$ at T=1.0 (Case 2 in section 4). Case (a): u plot; Case (b): v plot.

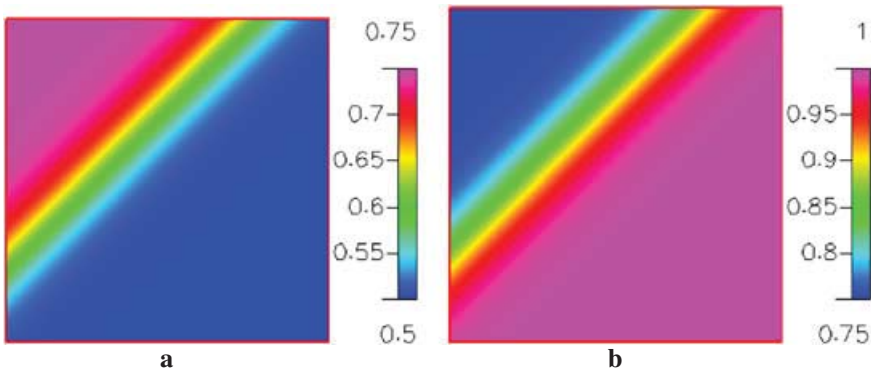

Figure 13. Fourth order solution of the 2D Burgers' equation, with $\mu = 0.1$ at T=1.0 (Case 2 in section 4). Case (a): u plot; Case (b): v plot.

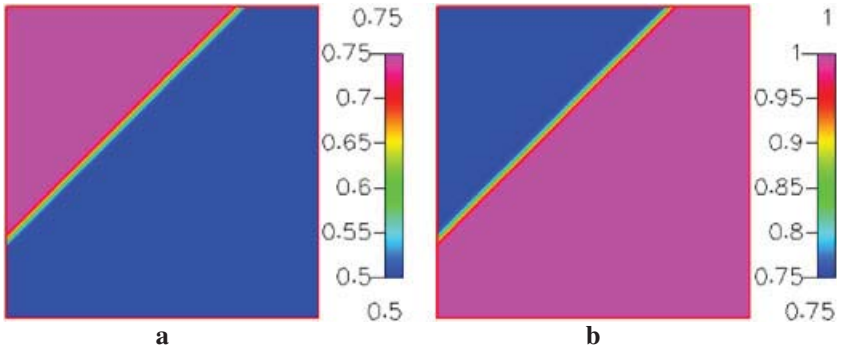

Figure 14. Fourth order solution of the 2D Burgers' equation, with $\mu = 0.1$ at T=1.0 (Case 2 in section 4). Case (a): u plot; Case (b): v plot.

DOI: 10.1002/fld

where the Fourier coefficients are defined by

$$a_0 = \int_0^2 \exp\left(-(2\pi\mu)^{-1}(1-\cos(\pi x))\right)dx$$
$$a_n = 2\int_0^2 \exp\left(-(2\pi\mu)^{-1}(1-\cos(\pi x))\right)\cos(n\pi x)dx \quad . \tag{4.4}$$

We first show the nonoscillatory nature of the solution and contrast it with the one obtained without using a transformation. Figures 3 and 4 show the second order numerical solutions obtained using both the methods at different time instances. It can be seen that the original method, starts becoming oscillatory, as the discontinuity develops. The current formulation is nonoscillatory at all times. The situation is identical for both the third (Figures 5 and 6) and fourth order simulations (Figures 7 and 8). The LDG formulation was employed for the previously mentioned simulations.

Accuracy studies were performed using the LDG and the LDG2 formulation. These are shown in Tables I–III. It can be seen that the second and the fourth order schemes reach their desired accuracy asymptotically irrespective of the value of the diffusion coefficient. We do experience a drop in order for the third order (odd ordered) scheme. This phenomenon has been reported by researchers in the past [10]. It can be observed that the LDG2 method generates much lower errors than the LDG scheme for all the three orders. This is in accord with the results of the analysis. [11].

Table VII. Accuracy study for the 2D Burgers' equation using the second order spectral volume formulation at T = 2.0 (Case 3 in section 4).

Method	μ	#N	L_1 error	L_1 order	L_∞ error	L_∞ order
LDG	0.001	10x10x2	8.13e-02	–	5.75e-01	–
		20x20x2	2.98e-02	1.45	2.24e-01	1.36
		40x40x2	8.66e-03	1.78	7.19e-02	1.64
		80x80x2	2.31e-03	1.91	2.06e-02	1.80
LDG2	0.001	10x10x2	5.67e-02	–	3.90e-01	–
		20x20x2	1.96e-02	1.53	1.39e-01	1.49
		40x40x2	5.30e-03	1.89	3.91e-02	1.83
		80x80x2	1.36e-03	1.96	1.03e-02	1.92
LDG	0.01	10x10x2	2.30e-02	–	1.52e-01	–
		20x20x2	7.64e-03	1.59	5.45e-02	1.48
		40x40x2	2.15e-03	1.83	1.60e-02	1.77
		80x80x2	5.52e-04	1.96	4.28e-03	1.90
LDG2	0.01	10x10x2	1.64e-02	–	1.02e-01	–
		20x20x2	5.15e-03	1.67	3.46e-02	1.56
		40x40x2	1.39e-03	1.89	9.59e-03	1.85
		80x80x2	3.52e-04	1.98	2.52e-03	1.93
LDG	0.1	10x10x2	8.20e-03	–	5.97e-02	–
		20x20x2	2.42e-03	1.76	1.86e-02	1.68
		40x40x2	6.53e-04	1.89	5.03e-03	1.89
		80x80x2	1.67e-04	1.97	1.30e-03	1.95
LDG2	0.1	10x10x2	5.91e-03	–	4.12e-02	–
		20x20x2	1.71e-03	1.79	1.24e-02	1.73
		40x40x2	4.52e-04	1.92	3.32e-03	1.90
		80x80x2	1.14e-04	1.98	8.55e-04	1.96

LDG, local discontinuous galerkin.

DOI: 10.1002/fld

The steady state solution can be seen in Figures 9–11. These correspond to diffusion coefficients of 0.01, 0.005, and 0.002, respectively. Five hundred fourth order SVs were used. It can be seen that the solutions are nonoscillatory and they capture the discontinuity very efficiently.

These results show that a high order accurate nonoscillatory limiter free solution can be obtained for the Burgers' equation using the SV formulation.

4.2. Test case 2

In this test case, we solve the 2D Burgers' equation, given in equation (3.5) over a square domain: [0:1] x [0:1]. The exact solutions for this case are

$$\begin{aligned} u(x,y,t) &= \frac{3}{4} - \frac{1}{4\{1+\exp((-4x+4y-t)/(32\mu))\}} \\ v(x,y,t) &= \frac{3}{4} + \frac{1}{4\{1+\exp((-4x+4y-t)/(32\mu))\}} \end{aligned} \quad . \tag{4.5}$$

The initial and the boundary conditions were derived. The simulation was run until 1.0 second, and the numerical solutions were compared with the exact solutions. Accuracy studies were performed using the LDG and the LDG2 formulation and are shown in Tables IV–VI. The findings are similar to that of the 1D simulation: (i) the second and the fourth order schemes reach their desired accuracy asymptotically irrespective of the value of the diffusion coefficient; (ii) once again, a drop in order for the third order (odd ordered) scheme is observed; (iii) the LDG2 method generates much lower errors than the LDG scheme for all the three orders.

Table VIII. Accuracy study for the 2D Burgers' equation using the third order spectral volume formulation at T = 2.0 (Case 3 in section 4).

Method	μ	#N	L_1 error	L_1 order	L_∞ error	L_∞ order
LDG	0.001	10x10x2	1.20e-02	–	8.91e-02	–
		20x20x2	1.75e-03	2.78	1.38e-02	2.69
		40x40x2	2.29e-04	2.93	1.90e-03	2.86
		80x80x2	3.55e-05	2.69	3.22e-04	2.56
LDG2	0.001	10x10x2	1.03e-02	–	7.79e-02	–
		20x20x2	1.48e-03	2.80	1.15e-02	2.76
		40x40x2	1.91e-04	2.95	1.56e-03	2.88
		80x80x2	2.92e-05	2.71	2.56e-04	2.61
LDG	0.01	10x10x2	3.22e-03	–	2.33e-02	–
		20x20x2	4.40e-04	2.87	3.23e-03	2.85
		40x40x2	6.77e-05	2.70	5.29e-04	2.61
		80x80x2	1.33e-05	2.35	1.14e-04	2.21
LDG2	0.01	10x10x2	2.85e-03	–	1.98e-02	–
		20x20x2	3.87e-04	2.88	2.69e-03	2.88
		40x40x2	5.92e-05	2.71	4.31e-04	2.64
		80x80x2	1.14e-05	2.38	9.13e-05	2.24
LDG	0.1	10x10x2	1.18e-03	–	8.80e-03	–
		20x20x2	1.61e-04	2.87	1.24e-03	2.82
		40x40x2	2.64e-05	2.61	2.20e-04	2.50
		80x80x2	6.30e-06	2.07	5.14e-05	2.10
LDG2	0.1	10x10x2	1.06e-03	–	7.74e-03	–
		20x20x2	1.43e-04	2.89	1.08e-03	2.84
		40x40x2	2.36e-05	2.60	1.88e-04	2.52
		80x80x2	5.50e-06	2.10	4.37e-05	2.11

LDG, local discontinuous galerkin.

DOI: 10.1002/fld

The surface plot of the solutions can be seen in Figures 12–14. 80x80x2 grids, employing fourth order SVs were used. Once again, the solutions look nonoscillatory and they capture the discontinuity very efficiently.

4.3. Test case 3

In this test case, we solve the single 2D Burgers' equation:

$$\frac{\partial u}{\partial t} + uu_x + uu_y = \mu(u_{xx} + u_{yy}) \tag{4.6}$$

Unlike the traditional coupled system of Burgers' equations, the previous example consists of a single equation. In order to employ the transformation methods discussed in this paper, we bring in an additional variable, $v(x, y, t) = u(x, y, t)$, so as to recreate the system outlined in equation 3.5.

The actual problem has the exact solution given by

$$u(x, y, t) = [1 + \exp\{(x + y - t)/2\mu\}]^{-1}, \tag{4.7}$$

with the domain being a square: [0:2] x [0:2]. As before, the initial and the boundary conditions are derived from the previously mentioned exact solution. The simulation was run until 2.0 seconds, and the numerical solution was compared with the exact solution. Accuracy studies were performed using the LDG and the LDG2 formulation and are shown in Tables VII–IX. The findings are similar to that of the earlier test cases.

Table IX. Accuracy study for the 2D Burgers' equation using the fourth order spectral volume formulation at T = 2.0 (Case 3 in section 4).

Method	μ	#N	L_1 error	L_1 order	L_∞ error	L_∞ order
LDG	0.001	10x10x2	2.21e-03	–	1.72e-02	–
		20x20x2	1.59e-04	3.80	1.36e-03	3.66
		40x40x2	1.02e-05	3.96	9.24e-05	3.88
		80x80x2	6.37e-07	4.00	5.86e-06	3.98
LDG2	0.001	10x10x2	2.13e-03	–	1.59e-02	–
		20x20x2	1.52e-04	3.81	1.23e-03	3.69
		40x40x2	9.62e-06	3.98	8.20e-05	3.91
		80x80x2	6.02e-07	4.00	5.16e-06	3.99
LDG	0.01	10x10x2	5.77e-04	–	4.10e-03	–
		20x20x2	3.89e-05	3.89	2.90e-04	3.82
		40x40x2	2.45e-06	3.99	1.93e-05	3.91
		80x80x2	1.53e-07	4.00	1.22e-06	3.99
LDG2	0.01	10x10x2	5.52e-04	–	3.91e-03	–
		20x20x2	3.75e-05	3.88	2.73e-04	3.84
		40x40x2	2.36e-06	3.99	1.80e-05	3.92
		80x80x2	1.47e-07	4.00	1.13e-06	3.99
LDG	0.1	10x10x2	1.99e-04	–	1.49e-03	–
		20x20x2	1.31e-05	3.93	1.01e-04	3.89
		40x40x2	8.16e-07	4.00	6.41e-06	3.97
		80x80x2	5.09e-08	4.00	4.01e-07	4.00
LDG2	0.1	10x10x2	1.87e-04	–	1.40e-03	–
		20x20x2	1.22e-05	3.94	9.38e-05	3.90
		40x40x2	7.62e-07	4.00	6.03e-06	3.96
		80x80x2	4.76e-08	4.00	3.76e-07	4.00

LDG, local discontinuous galerkin.

DOI: 10.1002/fld

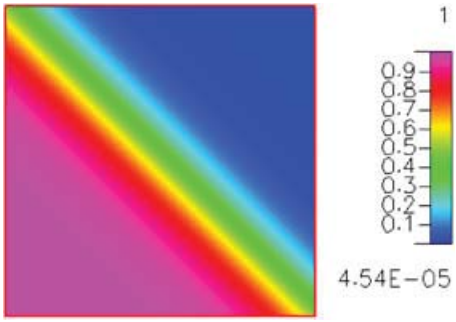

Figure 15. Fourth order solution of the 2D Burgers' equation, with $\mu = 0.1$ at T=2.0 (Case 3 in section 4).

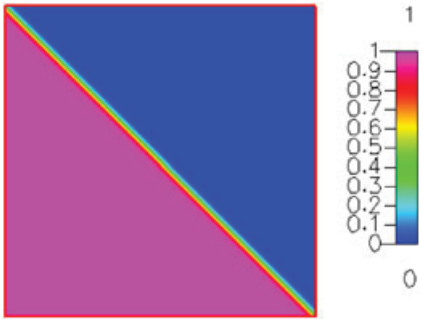

Figure 16. Fourth order solution of the 2D Burgers' equation, with $\mu = 0.01$ at T=2.0 (Case 3 in section 4).

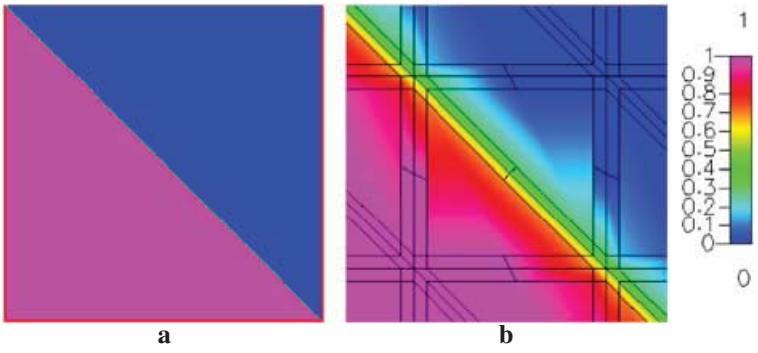

Figure 17. Fourth order solution of the 2D Burgers' equation, with $\mu = 0.001$ at T=2.0 (Case 3 in section 4). Case (a): solution across the entire domain; Case (b): solution near the discontinuity.

The surface plot of the solution can be seen in Figures 15–17. 80x80x2 grids, employing fourth order SVs were used. Once again, the solutions look nonoscillatory and they capture the discontinuity very efficiently.

Because the transformed equation is a diffusion equation, the CPU time per time step is smaller due to the absence of the inviscid fluxes. However, the gains are offset by the repeated usage of the Boole numerical integration during the repeated invocation calls to the BC routine (in 2D). In general, the new formulation is about 40% slower than the original one for the fourth order simulations and about 20% slower for the third order simulations.

DOI: 10.1002/fld

5. CONCLUSIONS

A limiter free high order SV formulation was developed to solve the Burgers' equation. The Hopf–Cole transformation, which maps the Burgers' equation to a linear diffusion equation, was employed for a purpose. The LDG and the LDG2 viscous flux discretization methods were used to solve this heat conduction equation in the SV context. The initial and the boundary conditions for the heat conduction equation were high order accurate transformations from the original Burgers' equation.

Numerical experiments were performed in both 1D and 2D. These experiments demonstrated:

1. The shock can be captured, without the use of a limiter
2. High order accuracy of the scheme even for very low viscosity
3. Non oscillatory nature of the solution
4. Superiority of the LDG2 scheme, when compared with the LDG scheme.

Future work will involve cases, wherein the constraint equation (3.9) can be eliminated and obtaining high order numerical solutions to other nonlinear partial differential equations (for instance the Korteweg–de Vries equations) which generate discontinuous solutions.

6. APPENDIX

6.1. Boole's rule for numerical integration.

Consider $y = f(x)$ over $[x_0, x_4]$. The grid locations in between are equidistant and are given by $x_1 = x_0 + h$, $x_2 = x_0 + 2h$, $x_3 = x_0 + 3h$, and $x_4 = x_0 + 4h$. Boole's rule of numerical integration is given by

$$\int_{x_0}^{x_4} f(x)dx = \frac{2h}{45}(7f(x_0) + 32f(x_1) + 12f(x_2) + 32f(x_3) + 7f(x_4)) - \frac{8}{945} f^6(c)h^7, \quad (6.1)$$

where $x_0 \leqslant c \leqslant x_4$.

REFERENCES

1. Wang ZJ. Spectral (finite) volume method for conservation laws on unstructured grids: basic formulation. *Journal of Computational Physics* 2002; **178**:210.
2. Wang ZJ, Liu Y. Spectral (finite) volume method for conservation laws on unstructured grids II: extension to two-dimensional scalar equation. *Journal of Computational Physics* 2002; **179**:665.
3. Wang ZJ, Liu Y. Spectral (finite) volume method for conservation laws on unstructured grids III: extension to one-dimensional systems. *Journal of Computational Physics* 2004; **20**:137.
4. Wang ZJ, Liu Y. Spectral (finite) volume method for conservation laws on unstructured grids IV: extension to two-dimensional Euler equations. *Journal of Computational Physics* 2004; **194**:716.
5. Wang ZJ, Liu Y. Extension of the spectral volume method to high-order boundary representation. *Journal of Computational Physics* 2006; **211**:154–178.
6. Liu Y, Vinokur M, Wang ZJ. Spectral (finite) volume method for conservation laws on unstructured grids V: extension to three-dimensional systems. *Journal of Computational Physics* 2006; **212**:454–472.
7. Cockburn B, Shu C-W. The local discontinuous Galerkin method for time-dependent convection diffusion system. *SIAM Journal on Numerical Analysis* 1998; **35**:2440–2463.
8. Liang C, Kannan R, Wang ZJ. A-p-multigrid Spectral Difference method with explicit and implicit smoothers on unstructured grids. *Computers & Fluids* February 2009; **38**(2):254–265.
9. Sun Y, Wang ZJ, Liu Y. Spectral (finite) volume method for conservation laws on unstructured grids VI: extension to viscous flow. *Journal of Computational Physics* 2006; **215**:41–58.
10. Kannan R, Wang ZJ. A study of viscous flux formulations for a p-multigrid spectral volume Navier–Stokes solver. *Journal of Scientific Computing* 2009; **41**(2):165–199.
11. Kannan R, Wang ZJ. LDG2: A variant of the LDG flux formulation for the spectral volume method. *Journal of Scientific Computing* 2011; **46**:314–328.
12. Kannan R, Wang ZJ. The direct discontinuous Galerkin (DDG) viscous flux scheme for the high order spectral volume method. *Computers & Fluids* 2010; **39**:2007–2021.

DOI: 10.1002/fld

13. Kannan R. An implicit LU-SGS spectral volume method for the moment models in device simulations: formulation in 1D and application to a p-multigrid algorithm. *International Journal for Numerical Methods in Biomedical Engineering*. DOI: 10.1002/cnm.1359. (formerly Communications in numerical methods in engineering), accepted on October 11, 2009, Published Online: 1 Feb 2010.
14. Kannan R. An implicit LU-SGS spectral volume method for the moment models in device simulations II: accuracy studies and performance enhancements using the penalty and the BR2 formulations. *International Journal for Numerical Methods in Biomedical Engineering* 2011; **27**:650–660. (formerly Communications in numerical methods in engineering).
15. Kannan R. An implicit LU-SGS spectral volume method for the moment models in device simulations III: accuracy enhancement using the LDG2 flux formulation for non-uniform grids. *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, (submitted).
16. Kannan R, Wang ZJ. A high order Spectral Volume method for moving boundary problems. *40th Fluid Dynamics Conference and Exhibit, AIAA 2010-4992*, Chicago, IL, USA, June 28–July 1 2010.
17. Kannan R. A high order spectral volume formulation for solving equations containing higher spatial derivative terms: formulation and analysis for third derivative spatial terms using the LDG discretization procedure. *Communications in Computational Physics*, Accepted on Jan 10.
18. Kannan R. A high order spectral volume method for elastohydrodynamic lubrication problems: formulation and application of an implicit p-multigrid algorithm for line contact problems. *Computers & Fluids*, Accepted on March 18, 2011. DOI: 10.1016/j.compfluid.2011.03.013.
19. Hopf E. The partial differential equation $u_t + uu_x = \mu u_{xx}$. *Communications on Pure and Applied mathematics* 1950; **3**:201–230.
20. Cole JD. On a quasilinear parabolic equations occurring in aerodynamics. *Quarterly of Applied Mathematics* 1951; **9**:225–236.
21. Liao W. A fourth-order finite-difference method for solving the system of two-dimensional Burgers' equations. *International journal for numerical methods in fluids* 2010; **64**:565–590.
22. Ohwada T. Cole–Hopf Transformation as numerical tool for the Burgers equation. *Applied Mathematics and Computation* 2009; **8**(1):107–113.
23. Soliman AA. The modified extended tanh-function method for solving Burgers-type equations. *Physica A* 2006; **361**:394–404.
24. Rusanov VV. Calculation of interaction of non-steady shock waves with obstacles. *USSR Computational Mathematics and Mathematical Physics* 1961; **1**:267–279.
25. Roe PL. Approximate Riemann solvers, parameter vectors and difference schemes. *Journal of Computational Physics* 1981; **43**:357–372.
26. Liou M-S, Steffen C. A new flux splitting scheme. *Journal of Computational Physics* 1993; **107**:23–39.
27. Balakrishnan K, Menon S. On the role of ambient reactive particles in the mixing and afterburn behind explosive blast waves. *Combustion Science and Technology* 2010; **182**(2):186–214.
28. Balakrishnan K, Menon S. On turbulent chemical explosions into dilute aluminium particle clouds. *Combustion Theory and Modelling* 2010; **182**:186–214.
29. Balakrishnan K, Nance DV, Menon S. Simulation of impulse effects from explosive charges containing metal particles. *Shock Waves* 2010; **20**(3):217–239.
30. Shu C-W. Total-variation-diminishing time discretizations. *SIAM Journal on Scientific and Statistical Computing* 1988; **9**:1073–84. Navier–Stokes equations on triangular meshes. AIAA J.

DOI: 10.1002/fld