Journal of
Applied & Computational Mathematics

Liu and Wang, J Appl Computat Math 2015, 4:2
DOI: 10.4172/2168-9679.1000208

Research Article Open Access

High Order Numerical Solutions to Convection Diffusion Equations with Different Approaches

Don Liu* and Yifan Wang

Mathematics and Statistics, Louisiana Tech University Ruston, LA 71272, USA

Abstract

Convection diffusion equations, one of the fundamental kinds of nonlinear partial differential equations (PDE), describe a variety of phenomena such as fluid convection and diffusion and traffic flow etc. This paper uses four different approaches to handle nonlinearity in convection diffusion equations. High order accuracy numerical solutions to nonlinear PDE were acquired with several discretization methods: spectral modal and nodal element, finite element, and compact difference methods. Numerical results were compared with exact solutions to demonstrate the error convergence rate. For irregular domains where exact solutions are unavailable, very high accuracy numerical solutions were used for error comparisons. Pros and cons of these four approaches for handling nonlinearity were reviewed and discussed. This study provides a reference in solving nonlinear PDE with different options.

Keywords: Spectral element method; Finite element method; Compact difference method; Convection diffusion equation; Lagrange interpolation

Introduction

Numerically solving nonlinear PDE has been on the center stage for computational sciences, especially in computational fluid dynamics (CFD), a mature interdisciplinary area with many applications. The key component of CFD is obtaining reliable and accurate numerical solutions to governing equations such as Burgers' equations, convection diffusion equations, and Navier-Stokes equations under a variety of boundary and initial conditions in different dimensions. Convection diffusion equation and Burgers' equation have been widely used to test numerical algorithms as they are simple nonlinear PDE describing convection and diffusion of a fluid, gas dynamics, shock waves, acoustic waves, and traffic flows etc. Handling nonlinear terms is central to solving convection diffusion equations and Burgers' equations.

Previous studies on convection diffusion and Burgers' equations are mainly analytical and numerical. Courant et al. [1] proposed the method of characteristics to solve Burgers' equation. But it has a limitation and would not work for the inviscid Burgers' equation because the characteristics may intersect at certain time and may result in discontinuities in solution. Hopf et al. [2] introduced a nonlinear transformation to convert an original PDE into a linear heat equation which can be solved with less effort. Zhu et al. [3] reported a discrete Adomain decomposition method, which derived a fully implicit finite difference scheme for Burgers' equation, and nonlinear terms was defined by the infinite series of Adomian's polynomials. Srivastava et al. [4,5] used finite difference method to discretize spatial derivatives and adopted Newton's root-finding method [6,7] to solve a system of nonlinear equations without linearizing Burgers' equation. Although this method has the advantage of solving Burgers' equation implicitly, generating Jacobian matrix of the system is computational expensive.

Nonlinear terms could be numerically solved either with a linearization or without a linearization such as using conservative forms. Linearization could relieve the constraint on the size of time steps due to stability condition because a linearized PDE could be solve implicitly with much larger time step than the time step of an explicit method. For very large systems of nonlinear PDE, linearization is especially useful in decoupling systems of PDE and making them possible to be solved individually and sequentially. For nonlinear PDE in multiple dimensions, linearization also facilitates decoupling and makes it possible to solve one component ahead of time. On the other hand, a relatively recent and successful approach is to cast a nonlinear term into a conservative form and use numerical fluxes to solve conservation laws [8-13]. This approach is very effective for hyperbolic equation where convection is dominant and information propagates along certain direction.

There are many methods to solve convection diffusion equations. Compact difference method (CDM) [14-18], due to its simple nature and convenience in implementation, offers high order algebraic convergence and uses smaller stencils than finite difference method. But the linear system is usually twice or even three times bigger than the one from finite difference. Finite element method (FEM) is known for handling geometrical complexity. Fletcher et al. obtained finite element solution to convection diffusion equations [19,20]. However, FEM is limited in acquiring higher order accuracy as basis functions become less orthogonal mutually once the order of basis functions increases beyond fifth and mass matrices become ill-conditioned. For high accuracy resolution, although one could use h-type discretization refinement, i.e., smaller and more elements, rounding errors would accumulate and even defeat accuracy at certain point. Spectral element method (SEM), first appeared in [21], could be a good alternative. It achieves very high accuracy (e.g. 15^{th}) by using orthogonal polynomial bases and zeros of orthogonal polynomials as quadrature points. SEM is capable of hp-refinement, and especially the p-type refinement which enhances resolution without extra numbers of elements [22-24]. Due to the capability of handling complexity geometry and both hp-type refinement for high accuracy, SEM has successfully appeared

***Corresponding author:** Don Liu, Mathematics and Statistics, Louisiana Tech University Ruston, LA 71272 USA, Tel: (+1) 318 257 4670; E-mail: DonLiu@LATech.edu

Received January 22, 2015; **Accepted** February 22, 2015; **Published** March 10, 2015

Citation: Liu D, Wang Y (2015) High Order Numerical Solutions to Convection Diffusion Equations with Different Approaches. J Appl Computat Math 4: 208. doi:10.4172/2168-9679.1000208

in computational fluid dynamics [21,25-34], especially in simulating microfluidic devices [35-41]. SEM could also be used to model advanced microfluidic biosensors [42] with applications in magnetic labeling, sorting, medical diagnostics, and weak magnetic field detection [43].

Another relatively new approach, discontinuous Galerkin finite element method (DG- FEM), has been successful in handling nonlinear PDE in which nonlinear terms are written in conservative forms [22,44-54]. Using discontinuous basis functions and appropriate numerical fluxes at boundaries, DG-FEM is free from the C0 constraint on basis functions and is capable of capturing discontinuity in the solution. How to choose numerical fluxes is crucial to some strong hyperbolic equations.

In this paper, we discuss four different approaches, namely Method I: Taylor expansion, Method II: Cole-Hopf transformation, Method III: Compact Difference Preprocessing, and Method IV: Conservative Form, to handle nonlinearity. With these approaches, we produce numerical solutions to one and two dimensional convection diffusion equations using a blend of CDM, FEM, and SEM. Convergence rates are demonstrated in h-type and/or p-type refinement tests. Pros and cons of different approaches are discussed to provide a reference for solving nonlinear PDE in multiple dimensions and systems of nonlinear PDE.

Methodology and Numerical Examples

We consider convection diffusion equations in one or two dimensional domains:

$$\frac{\partial \mathbf{u}}{\partial t}+(\mathbf{u}\cdot\nabla)\mathbf{u}=\mu\Delta\mathbf{u},\qquad in\ \Omega\ and\ dt\geq 0, \tag{1}$$

where μ is the dynamic viscosity. In some occasions, for the convenience of discussion, we treat the viscosity as a small value and drop the diffusion term; therefore, the convection diffusion becomes the inviscid Burgers' equation. In order to verify the accuracy of numerical solutions, we need an exact solution to compute the norm of errors at all quadrature points. Therefore, the convection diffusion equation was modified with the diffusion term replaced by a known function of space and time. In some examples, we solve the following modified convection diffusion equations instead:

$$\frac{\partial \mathbf{u}}{\partial t}+(\mathbf{u}\cdot\nabla)\mathbf{u}=\mathbf{f}(\mathbf{x},t). \tag{2}$$

In the following subsections, four different approaches are discussed to handle nonlinearity in one dimensional (1D) or two dimensional (2D) situations. Numerical solutions are provided with SEM, FEM, CDM or a blend of the above.

Linearization with Taylor expansion (Method I)

We start with the following simplest nonlinear PDE:

$$\frac{\partial u}{\partial t}+uu_x=f(x,t),\quad 1\leq x\leq 2,\quad t\geq 0, \tag{3}$$

$$u(x,0)=u_0(x), \tag{3a}$$

$$u(1,t)=f_1(t), \tag{3b}$$

and use two different discretization methods in time and space in subsequent discussions.

Backward Euler and finite element method

Discretizing Equ. (3) implicitly in time, we obtain:

$$\frac{u^{n+1}-u^n}{\Delta t}+u^{n+1}u_x^{n+1}=f^{n+1}. \tag{4}$$

Due to the nonlinear term, we could not directly treat the entire convective term implicitly, because after a Galerkin projection, there will be three different indices for the nonlinear term. Therefore, we attempted to linearize Equ. (4). The first approach to handle nonlinearity, denoted as Method I, applies the Taylor expansion to the coefficient of u^{n+1} in time and repetitively replace time derivatives with space derivatives using the exact information in the governing PDE. This linearization is similar to the general idea of Lax-Wendroff scheme [55,56,57], except we use first or higher order Taylor expansion [58] to achieve higher accu- racy and implement a finite element or high order difference method to compute the spatial derivatives. The CFL number here is no more than 1. The first order approximation for the nonlinear term has the second order truncation error in time:

$$u^{n+1}u_x^{n+1}=(u^n+\Delta tu_t^n+O(\Delta t^2))u_x^{n+1}\approx(u^n+\Delta tu_t^n)u_x^{n+1}. \tag{5}$$

Substitute Equ. (5) into Equ. (4), we have:

$$\frac{u^{n+1}-u^n}{\Delta t}+(u^n+\Delta tu_t^n)u_x^{n+1}=f^{n+1}. \tag{6}$$

Using the exact information from the PDE: $u_t^n=f^n-u^nu_x^n$, we substitute it into Equ. (6):

$$\frac{u^{n+1}-u^n}{\Delta t}+(u^n+\Delta tf^n-\Delta tu^nu_x^n)u_x^{n+1}=f^{n+1}. \tag{7}$$

Let $(u^n)*=u^n+\Delta tf^n-\Delta tu^nu_x^n$, which is known at the time step n, then Equ. (7) becomes a linearized ordinary differential equation:

$$\frac{u^{n+1}-u^n}{\Delta t}+(u^n)^*u_x^{n+1}=f^{n+1}. \tag{8}$$

Next we apply the nodal Galerkin projection to form a linear system $A\hat{U}=B$. We divide the domain into N elements. Within each element Ω_e, we expand u and u_x in k^{th} order Lagrangian basis functions over k+1 uniform points in order to simplify the evaluation of $(un)^*$, although an alternative way could use Lagrangian basis function over Gauss-Lobattolegendre points. In a typical element Ω_e:

$$u^n=\sum_{i=0}^{k}\hat{U}_i^n\phi_i(x),u^{n+1}=\sum_{i=0}^{k}\hat{U}_i^{n+1}\phi_i(x), \tag{9}$$

$$u_x^{n+1}=\sum_{i=0}^{k}\hat{U}_i^{n+1}\phi_{i'}(x). \tag{10}$$

Substituting Equs. (9) and (10) into Equ. (8) and applying Galerkin projection, we obtain the weak form of Equ. (8) within one element Ω_e:

$$\sum_{i=0}^{k}\int_{\Omega_e}\phi_i(x)\phi_j(x)dx\frac{\hat{U}_i^{n+1}-\hat{U}_i^n}{\Delta t}+\sum_{i=0}^{k}\int_{\Omega_e}\phi_i'(x)\phi_j(x)dx(U^n)_i^*\hat{U}_i^{n+1}=\int_{\Omega_e}f^{n+1}\phi_j(x)dx,\ j=0,1\ldots k-1,k, \tag{11}$$

where $(U^n)*$ denotes the expansion term for $(u^n)*$ at the point of i. After global assembling, the matrix-vector form of Equ. (8) is:

$$\underline{\underline{M}}\frac{\underline{\hat{U}}^{n+1}-\underline{\hat{U}}^n}{\Delta t}+\underline{\underline{K}}\underline{\hat{U}}^{n+1}=\underline{F}^{n+1}, \tag{12}$$

$$\underline{F}^{n+1}=\int_{\Omega_e}f(x,n+1)\phi_j(x)dx, \tag{13}$$

where, M is the global mass matrix; F^{n+1} is obtained by global assembly; the matrix K is obtained by the scalar multiplies global advection

matrix. If we choose a linear expansion basis, the matrix K has a tridiagonal form as below, while if quadratic and cubic basis function and pentadiagonal and seven-diagonal matrix for using respectively:

$$\begin{bmatrix} K_{22}U_2 & K_{22}U_2 & 0 & \cdots & \cdots & \cdots & 0 \\ K_{32}U_3 & K_{33}U_3 & K_{34}U_3 & 0 & \ddots & \ddots & \vdots \\ 0 & K_{43}U_4 & K_{44}U_4 & K_{45}U_4 & \ddots & \ddots & \vdots \\ \vdots & \cdots & \cdots & \cdots & \cdots & \cdots & \vdots \\ \vdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & 0 & K_{N-1N-2}U_{N-1} & K_{N-1N-1}U_{N-1} & K_{N-1N}U_{N-1} \\ 0 & \cdots & \cdots & \cdots & 0 & K_{NN-1}U_N & K_{NN}U_N \end{bmatrix}$$

In which, $U_l = \hat{U}_l^n + \Delta t F_l^n - \Delta t(\hat{U}^n \hat{U}_x^n)_l$, l stands for the numbering of global points, and each Klm corresponds to the entry of global advection matrix at l, m. Here, l and m is from 2 to N, since boundary points at 1 and N+1 are known. The key idea here is to treat $(u^n)^*$ explicitly and take it as a scalar multiplier for u^{n+1}, which is treated implicitly, then include it into the global advection matrix to form matrix K. Therefore, we obtain a set of discretized equations with only two indices which are represent Table 1 in a linear system.

Finally, Equ. (12) becomes:

$$(\underline{\underline{M}} + \Delta t \underline{\underline{K}})\underline{\hat{U}}^{n+1} = \underline{\underline{M}}\underline{\hat{U}}^n + \Delta t \underline{F}^{n+1}. \quad (14)$$

To examine the accuracy of this approach, we set up the problem as below:

$$\frac{\partial u}{\partial t} + uu_x = f(x,t) = -e^{(x-t)} + e^{(2x-2t)}, 1 \le x \le 2, t \ge 0, \quad (15)$$

subject to the boundary and initial conditions:

$u(1, t) = e^{1-t}$, $u(x, 0) = e^x$. (16)

The analytic solution to this well-posed problem is:

$u(x, t)=e^{x-t}$. (17)

Since this problem is time-dependent, time integrations are crucial in numerical solutions. We attempted both implicit and explicit time integration. The CFL condition (for the latter) and the diffusion condition were satisfied by restricting the time step to be small enough:

$$\Delta t < \frac{\Delta x}{U_{max}}, \quad (18)$$

$$\Delta t < \frac{\Delta x^2}{\mu}, \quad (19)$$

where U_{max} is the maximum of absolute phase velocity, Δt and Δx are time step and distance between quadrature points, respectively, and μ is the diffusion coefficient Table 2.

Order	α	β	A	B	C
4th	1/4	0	3/2	0	0
6th	1/3	0	14/2	1/9	0
8th	4/9	1/36	40/27	25/54	0

Table 1: Where the coefficients are.

order	α	β	γ	A	B	C	D	E	G	H
4th	1	3	-17/6	-17/6	3/2	3/2	-1/6	0	0	0
5th	1	4	-37/12	-37/12	2/3	3	-2/3	1/12	0	0
6th	1	5	-197/60	-197/60	-5/12	5	-5/3	5/12	-1/20	0
7th	1/10	1	-227/600	-227/600	-13/12	7/6	1/3	-1/24	1/300	0
8th	1/12	1	-79/240	-79/240	-77/60	55/48	5/9	-5/48	1/60	-1/720

Table 2: Where the coefficients are.

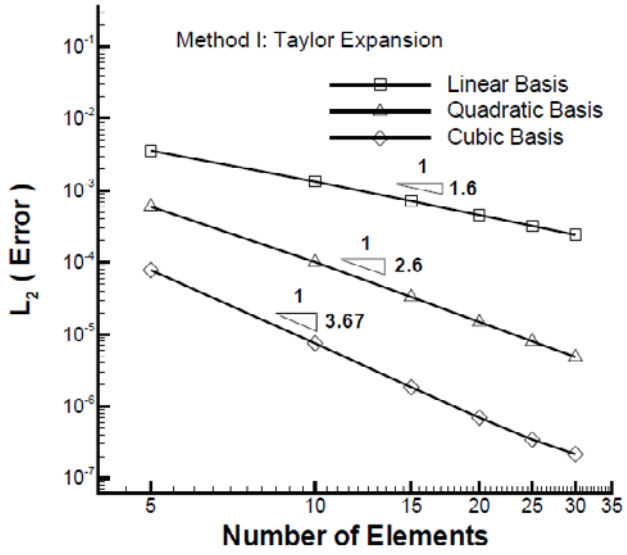

Figure 1: Convective Equ. (15) linearized with Method I: Algebraic convergence in space at a fixed time step $\Delta t=10^{-4}$.

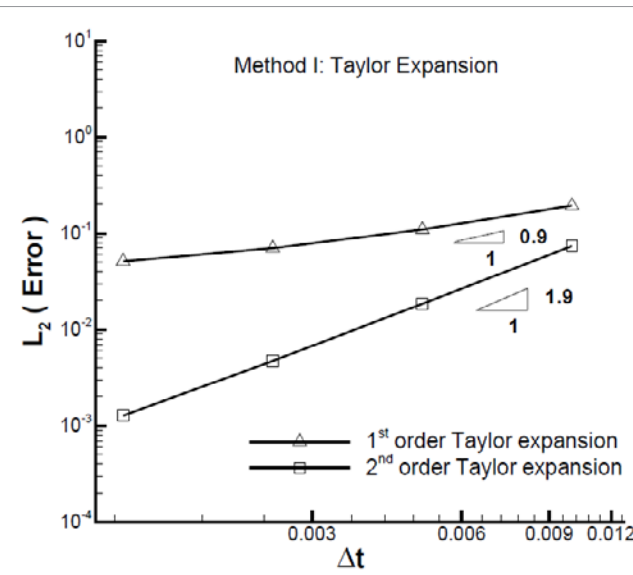

Figure 2: Convective Equ. (15) linearized with Method I: Temporal convergence at a fixedspatial resolution of using 10 elements for a total time of 0.2 dimensionless unit.

When we choose linear, quadratic, and cubic bases in space and backward Euler in time, the error estimations are order of $O(\Delta t^1+\Delta x^2)$, $O(\Delta t^1+\Delta x^3)$, $O(\Delta t^1+\Delta x^4)$, respectively. We compute the Euclidean norm of errors (L2 Error) at all nodes. Figure 1 presents the L_2 Error versus the number of elements in the log-log scale at a fixed time step $\Delta t=10^{-4}$. The convergence rates are basically one order higher than the order of expansion functions. To investigate the order of accuracy in time, we fix element number N=10, and vary Δt=0.01, 0.005, 0.0025, 0.00125. Using a first order Taylor expansion in the log-log scale, the first order convergence in time is observed in Figure 2. When we switch

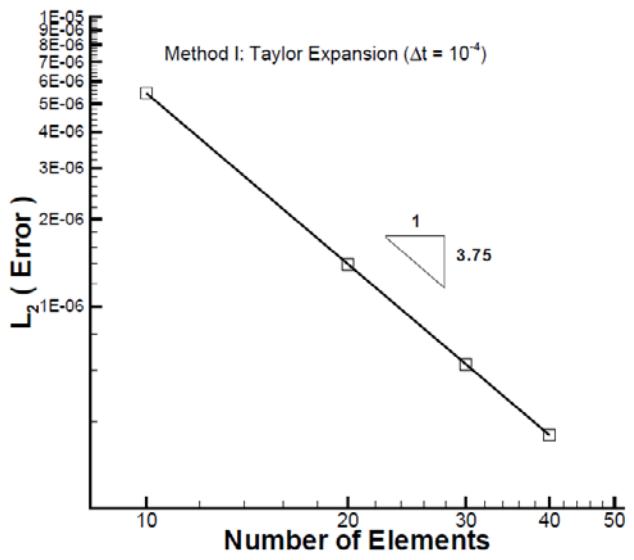

Figure 3: The convective equation (15) linearized with Method I: Convergence in space at afixed time step.

it to the second order Taylor expansion, a second order accuracy in time is obtained as shown in Figure 3.

Crank-Nicholson and compact difference method

We test the same approach for nonlinearity but using the Crank-Nicholson scheme in time and compact method in space for Equ. (3):

$$\frac{u_i^{n+1}-u_i^n}{\Delta t}+\frac{u_i^{n+1}u_{xi}^{n+1}+u_i^n u_{xi}^n}{2}=\frac{f_i^{n+1}+f_i^n}{2}. \quad (20)$$

To linearize the term $u_i^{n=1}u_{xi}^{n=1}$, we use the Taylor expansion in time again to express un+1 with some terms in the time level n, such that $u_i^{n=1}$ is replaced with u_i^*. Therefore, Equ. (20) becomes:

$$\frac{u_i^{n+1}-u_i^n}{\Delta t}+\frac{u_i^* u_{xi}^{n+1}+u_i^n u_{xi}^n}{2}=\frac{f_i^{n+1}+f_i^n}{2}. \quad (21)$$

In order to keep the 2nd order temporal accuracy, we use a few more terms in the expansion such that

$$u_i^{n+1}u_{xi}^{n+1}\approx[u_i^n+\Delta t u_{ti}^n+\frac{\Delta t^2}{2}u_{tti}^n+O(\Delta t^3)]u_{xi}^{n+1}. \quad (22)$$

After truncating the 2nd order term, we have:

$$u_i^{n+1}u_{xi}^{n+1}\approx[u_i^n+\Delta t u_{ti}^n]u_{xi}^{n+1}, \quad (23)$$

in which, u_t^n can be obtained from the PDE exactly:

$$u_i^n u_{xi}^n=f_i^n-u_{ti}^n. \quad (24)$$

Hence, Equ. (20) becomes linearized as below:

$$\frac{u_i^{n+1}-u_i^n}{\Delta t}+\frac{[u_i^n+\Delta t(f_i^n-u_i^n u_{xi}^n)]u_{xi}^{n+1}+u_i^n u_{xi}^n}{2}=\frac{f_i^{n+1}+f_i^n}{2}. \quad (25)$$

Rearrange the above equation, we have:

$$u_i^{n+1}+[\frac{\Delta t}{2}(u_i^n+\Delta t(f_i^n-u_i^n u_{xi}^n))]u_{xi}^{n+1}=u_i^n-[\frac{\Delta t}{2}u_i^n]u_{xi}^n+\frac{\Delta t}{2}(f_i^{n+1}+f_i^n). \quad (26)$$

We use the following 4th order Pa´de scheme to compute u^{n+1} and u_x^{n+1} simultaneously:

$$u_{xi-1}+4u_{xi}+u_{xi+1}=\frac{3u_{i+1}-3u_{i-1}}{h}+O(h^4). \quad (27)$$

To examine the spatial and temporal accuracy of using Crank-Nicholson and Compact Difference, we use the same problem as before, Equs. (15) and (16), with the same exact solution Equ. (17). For the spatial error only, we fix Δt=0.0001, compute u(x, 0.0001), and compare it with the exact solution in L_2 error. We notice that the number of elements N=10, 20, 30, 40 increases, the error goes down consistently. We observed that the order of convergence rate is close to 4th in space, as shown in Figure 3.

Figure 4 illustrates a temporal convergence rate close to the second order at the total time of 1. In this Figure 5, we set the time step to be Δt=0.001, 0.0005, 0.00025, 0.000125, use only 21 elements in space.

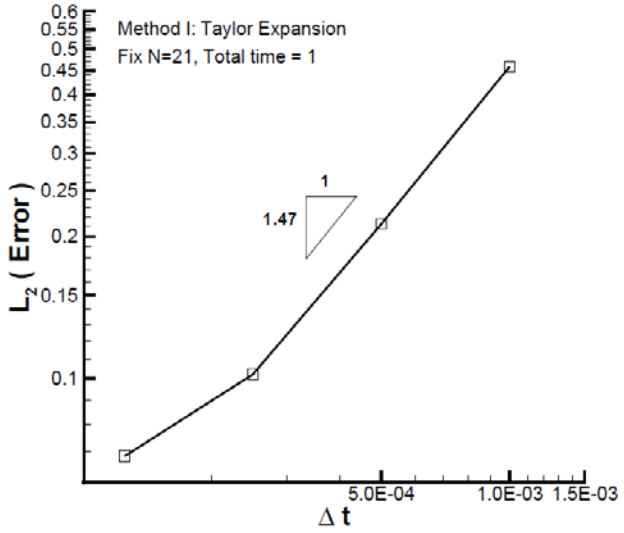

Figure 4: The convective equation (15) linearized with Method I: Convergence in time at afixed spatial resolution.

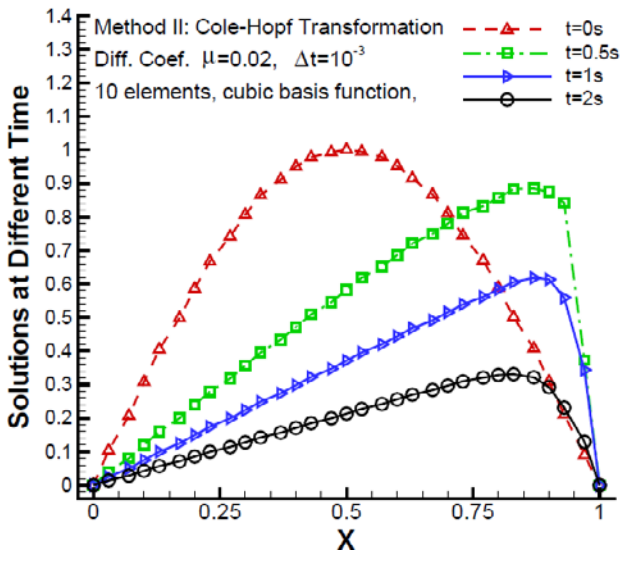

Figure 5: Evolution of numerical solution of 1D convection diffusion equation (28) linearizedwith Method II: Cole-Hopf transformation.

Since the Crank-Nicholson was used in time and 4th order Pa´de schemes in space, the second order convergence in time is anticipated.

In summary, the idea of using Taylor expansion to express an unknown of time level n+1 explicitly to approximate nonlinearity is convenient and stable, provided that CFL condition is obeyed. This idea is rather simple and could be used as long as an explicit expression for time derivative is possible or other approaches are difficult to use.

Linearization with Cole-Hopf transformation (Method II)

One-dimensional convection diffusion equation: Now we include the diffusion and consider the one dimensional convection diffusion equation:

$$\frac{\partial u}{\partial t}+uu_x=\mu u_{xx}, \tag{28}$$

with the initial condition and two Dirichlet boundary conditions below:

$$u(x,0)=f(x), 0\le x\le 1, \tag{28a}$$

$$u(0,t)=\alpha(t), u(1,t)=\beta(t), t\ge 0. \tag{28b}$$

With the Cole-Hopf transformation [2]:

$$u(x,t)=-2\mu\frac{w_x(x,t)}{w(x,t)}, \tag{29}$$

the original equation becomes a linear heat equation of in w(x, t),

$$\frac{\partial w}{\partial t}=\mu w_{xx}, 0\le x\le 1, t\ge 0 \tag{30}$$

with new initial and boundary conditions as below:

$$w(x,0)=e^{-\int_0^x \frac{f(s)}{2\mu}ds}, \quad 0\le x\le 1, \tag{30a}$$

$$2\mu w_x(0,t)+\alpha(t)w(0,t)=0, \quad t>0, \tag{30b}$$

$$2\mu w_x(1,t)+\beta(t)w(1,t)=0, \quad t>0. \tag{30c}$$

We solve the transformed linear equation, then use the inverse transformation, i.e., Equ. (29), to acquire the solution to the original problem.

Two-dimensional convection diffusion equation: For two-dimensional domains, the Cole-Hopf transformation [59] is given below:

$$u(x,y,t)=-2\mu\frac{w_x(x,y,t)}{w(x,y,t)}, \tag{31}$$

$$v(x,y,t)=-2\mu\frac{w_y(x,y,t)}{w(x,y,t)} \tag{32}$$

We could transform the vector form of convection diffusion equation, Equ. (1), into a scalar equation,

$$\frac{\partial w}{\partial t}=\mu\Delta w, \ in\Omega and\ t\ge 0 \tag{33}$$

To obtain numerical solutions, we compute w in Equ. (33) with the nodal spectral element method using high order Lagrangian interpolants. Then we use high order finite difference schemes to compute wx and wy from w. Finally, u and v could be obtained from w using Equs. (31) and (32), respectively. When we use the kth order Lagrangian interpolants as the basis functions, in order to guarantee the exponential convergence, we used k + 1st order difference schemes to compute wx and wy .

Numerical results: One-dimensional Example:

To validate Method II in one dimension (1D), we solve Equ. (28) with μ=0.02, u(x,0)=sin(πx), 0 ≤ x ≤ 1, and α(t)=β(t)=0. After the Cole-Hopf transformation, we have the heat equation in the new variable w(x) with new initial and boundary conditions:

$$w(x,0)=e^{\frac{\cos(\pi x)-1}{2\pi\mu}}, \quad 0\le x\le 1, \tag{34}$$

$$w_x(0,t)=w_x(1,t)=0, \quad t\ge 0. \tag{35}$$

Since the original nonlinear PDE becomes linear, we use implicit treatment in time and finite element method in space. After we obtain the solution in w(x), we use Cole-Hopf inverse transformation of to obtain the numerical solution to the original problem. Figure 5 shows the numerical solution to the 1D problem at different time. The number of elements is fixed to be 10. As in the figure, the wave propagates towards right and forms a wave front, and the magnitude of the wave decreases with the time due to the viscous dissipation. We used cubic basis functions, as the number of elements varying from 10, 20, 30, 40, to 50, the 4th order convergence rate in space was obtained as shown in Figure 6, where the time step was Δt=0.0001. For the following two dimensional (2D) situations, we test two examples using different initial and boundary conditions.

Two-dimensional Example I:

We set the viscosity μ=0.02 and the initial and boundary conditions for Equ. (1) as below:

$$u(x,y,0)=\sin(\pi x)\sin(\pi y), v(x,y,0)=y, \Omega: 0\le x,y\le 1, t\ge 0. \tag{36}$$

$$u(x,y,t)=0, v(x,y,t)=y, x,y\in\partial\Omega, \tag{37}$$

We perform the Cole-Hopf transformation to reduce a vector PDE

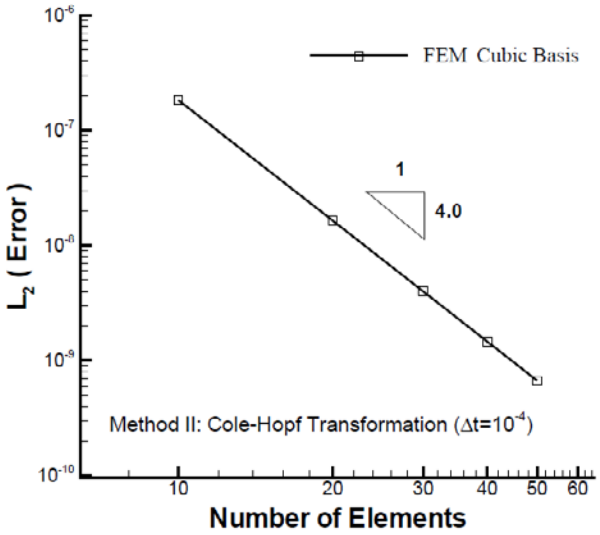

Figure 6: Spatial convergence for the convection diffusion equation (28) linearized withMethod II: Cole-Hopf transformation.

in u(x,y,t) into a scalar one in w(x,y,t). The domain was divided into 4 elements. Nodal SEM with 5th order basis function and Lagrangian interpolants over Gauss-Lobatto-Legendre points were used to solve the 2D scalar heat equation. We still use Lagrangian interpolation and difference method to compute wx and wy. By Equs. (31) and (32), we found the solutions to u(x,y) for the original problem. Figure 7a and b show the contour lines for the velocity component u and v at t=0.1, respectively. Figure 7c illustrates the elements and quadrature points. Figure 7d shows the vector field of u=(u,v). Because velocity components u and v were asymmetric and coupled together, the vector field convects in both x and y directions.

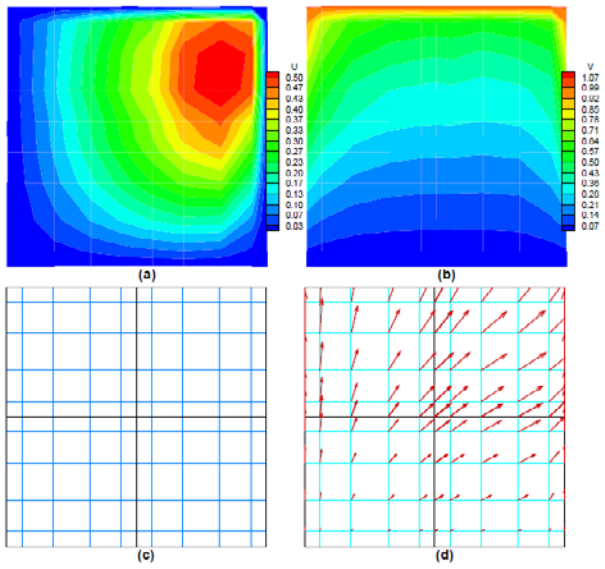

Figure 7: The two-dimensional convection diffusion equation (28) (Example I) linearized withMethod II: Velocity components u, v, the computational elements and quadrature points.

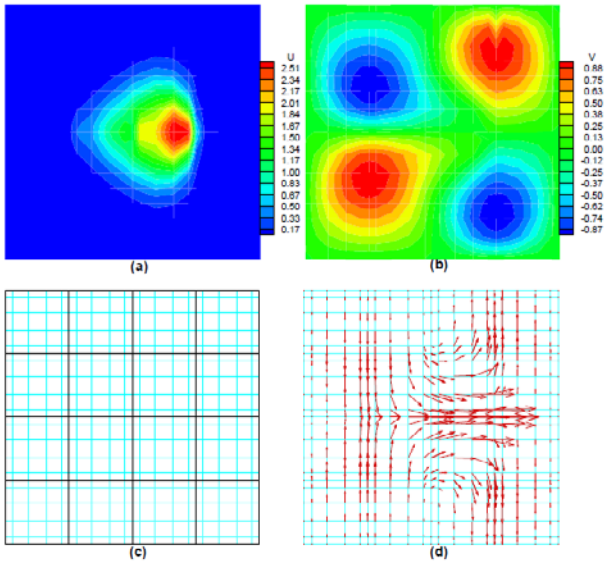

Figure 8: The two-dimensional convection diffusion equation (28) (Example II) linearizedwith Method II: Velocity components u, v, the mesh with 16 elements and quadrature points.

Two-dimensional Example II:

In the second test, we change the domain into $\Omega=\{(x, y): 0 \le x, y \le 2\pi\}$, set $\mu=0.005$, and use new initial conditions

$$u(x,y,0)=e^{(1-(x-\pi)2-(y-\pi)2)}, \quad (38)$$

$$v(x,y,0)=\sin(x)\sin(y), \quad (39)$$

and new boundary conditions on the boundary $\partial\Omega$

$$u(x,y,t)=0, v(x,y,0)=0, x,y \in \partial\Omega. \quad (40)$$

The domain was divided into 16 elements. We used the nodal SEM with 5th order basis function in each direction in each element to solve the 2D heat equation. Then, we used Lagrange interpolation and compact schemes to compute the first derivatives. Finally, solu- tions for u and v were obtained from Equs. (31) and (32). Numerical results were shown at time t=0.5. Figure 8a and 8b show contour lines for u and v, respectively. Figure 8c illustrates 16 elements and quadrature points. Figure 8d is the vector field for u=(u, v).

Generally speaking, Cole-Hopf transformation offers two advantages: reducing a vector equation into a linear scalar one and allowing an implicit temporal treatment in which the time step is unrestricted by the stability condition and time integration is unconditionally stable. However, Cole-Hopf transformation could only be applied to certain nonlinear PDE [58], such as hyperbolic PDE which contains second or higher spatial derivatives [60] and the Korteweg-de Vries equation (KdV equation) [61]. Chu et al. [62] listed a system of PDE with second order spatial derivatives that Cole-Hopf transformation is applicable.

Compact Difference Preprocessing (Method III)

Consider the 1D nonlinear problem in Equs. (3), (3a), (3b), and (3c), we linearize it by treating ux explicitly as the coefficient of u. This is called Method III in this paper, which is an explicit and non-conservative method [58]. The linearized equation below could be solved with an explicit scheme:

$$\frac{u^{n+1}-u^n}{\Delta t}+((u_x)^n)u^n=f^n, \quad (41)$$

or implicit scheme:

$$\frac{u^{n+1}-u^n}{\Delta t}+((u_x)^n)u^{n+1}=f^n. \quad (42)$$

The success of Method III relies on the accuracy of evaluating ux. We compute the nodal values of ux with high order compact difference [15-17,63] schemes before multiplying u. To demonstrate the idea of Method III, we compute ux at interior points with sixth order compact difference scheme as shown below:

$$12(u_x)_{i-1}+36(u_x)_i+12(u_x)_{i+1}=h^{-1}(-u_{i-2}-28u_{i-1}+28u_{i+1}+u_{i+2})+o(h^6), \quad (43)$$

where h=1 . For boundary points, we use Dirichlet boundary conditions (for simplicity) and the following sixth order scheme:

$$-60(u_x)_{i-1}-300(u_x)_i=h^{-1}(197u_{i-1}+25u_i-300u_{i+1}+100u_{i+2}-25u_{i+3}+3u_{i+4})+o(h^6). \quad (44)$$

Once we computed the values of $(u_x)_i^n$ at all points, Equs. (41) and (42) are linearized.We divide the whole domain into a total of N elements (Ωe) and express un and un+1 with expansions of basis functions (for example, linear) on each element:

$$u^n = \sum_{i=0}^{1} \hat{U}_i^n \phi_i(x), \tag{45}$$

$$u^{n+1} = \sum_{i=0}^{1} \hat{U}_i^{n+1} \phi_i(x). \tag{46}$$

Then we use the Galerkin projection to acquire the weak form of Equ. (42):

$$\sum_{i=0}^{1} \int_{\Omega_e} \phi_i(x)\phi_j(x)dx \frac{\hat{U}_i^{n+1} - \hat{U}_i^n}{\Delta t} + \sum_{i=0}^{1} \int_{\Omega_e} \phi_i(x)\phi_j(x)dx (u_x)_i^n \hat{U}_i^{n+1} = \int_{\Omega_e} f^n \phi_j(x)dx, j = 0,1. \tag{47}$$

After global assembling procedure, the linear system of Equ. (42) in vector form is:

$$\underline{\underline{M}} \frac{\underline{\hat{U}}^{n+1} - \underline{\hat{U}}^n}{\Delta t} + \underline{\underline{M}}^* \underline{\hat{U}}^{n+1} = \underline{F}^n, \tag{48}$$

in which, $\underline{\underline{M}}$ is the global mass matrix; $\underline{F}^n$ is the assembled RHS of the Equ. (47) over all element. The matrix $\underline{\underline{\ }}$ consists of the products of the global mass matrix and the explicit values of $(u_x)_i^n$ given by Equs. (43) and (44). The matrix M∗ has the following form:

$$\begin{bmatrix} M_{22}(u_x)_2^n & M_{23}(u_x)_2^n & 0 & \dots & \dots & 0 \\ M_{32}(u_x)_2^n & M_{33}(u_x)_2^n & M_{34}(u_x)_2^n & 0 & \dots & \dots \\ 0 & M_{43}(u_x)_2^n & M_{44}(u_x)_2^n & M_{45}(u_x)_2^n & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & 0 \\ \dots & \dots & 0 & M_{N-1N-2}(u_x)_{N-1}^n & M_{N-1N-1}(u_x)_{N-1}^n & M_{N-1N}(u_x)_{N-1}^n \\ 0 & \dots & \dots & 0 & M_{NN-1}(u_x)_N^n & M_{NN}(u_x)_N^n \end{bmatrix}$$

For the convenience of time integration, Equ. (48) can be written as:

$$(\underline{\underline{M}} + \Delta t \underline{\underline{M}}^*)\underline{\hat{U}}^{n+1} = \underline{\underline{M}}\underline{\hat{U}}^n + \Delta t \underline{F}^n. \tag{49}$$

Method III can readily be a high order scheme by adopting higher order compact schemes to evaluate ux on equispaced grids and use higher order polynomials as basis functions in a Galerkin finite element method. Using cubic or quartic basis functions could provide solu- tions with decent algebraic convergence. Beyond fifth order, the convergence rate starts to deteriorate due to the Runge's phenomenon

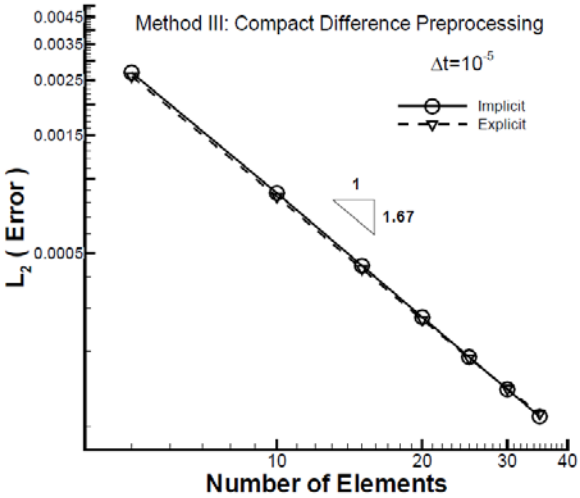

Figure 9: The 1D convective equation linearized with Method III: Convergence in space ata fixed time step.

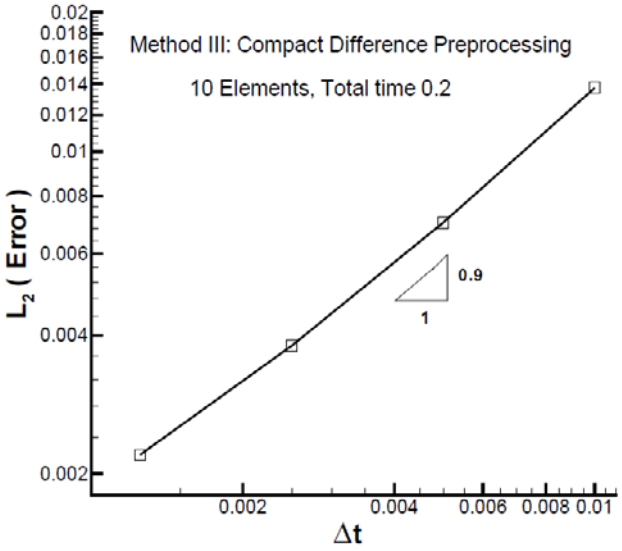

Figure 10: The 1D convective equation linearized with Method III: Convergence in time ata fixed spatial resolution.

and a spectral element method using orthogo- nal bases is a better alternative. We demonstrate this by using Lagrangian interpolation for ux from a uniform finite difference grid to Gauss-Lobatto-Legendre quadrature points. For two dimensional situation, the same idea works since Lagrangian interpolation in 2D can be utilized to calculate first derivatives of u at Gauss-Lobatto-Legendre points.

To test the accuracy of Method III in 1D, we consider the previous problem with the initial and boundary conditions in Equs. (15) and (16). We choose linear basis functions in space and forward Euler method in time, and use the 2nd order central difference scheme to compute value of ux. Therefore, the estimated error is of order $O(\Delta t^1 + \Delta x^2)$. The CFL condition is obeyed with $\Delta t = 10^{-4}$. By varying the spatial resolution, the number of elements from N=5 to 35, we present the spatial convergence in the log-log scale in Figure 9. The order of convergence is almost 2.

To examine the order of accuracy in time, we fix N=10 and decrease Δt=0.01, 0.005, 0.0025, 0.00125. The time convergence rate is almost one, as shown in the log-log scale in Figure 10. This convergence rate is expected because the forward Euler method was used.

For two-dimensional convection diffusion equations, we denote velocity components with u, v along x and y and let f, g be corresponding forcing terms. We perform linearization and set up the following implicit schemes:

$$\frac{u^{n+1} - u^n}{\Delta t} + (u_x)^n u^{n+1} + (u_y)^n v^{n+1} = f^{n+1}, \tag{50}$$

$$\frac{u^{n+1} - u^n}{\Delta t} + (u_x)^n u^{n+1} + (u_y)^n v^{n+1} = f^{n+1}, \frac{v^{n+1} - v^n}{\Delta t} + (v_x)^n u^{n+1} + (v_y)^n v^{n+1} = g^{n+1}. \tag{51}$$

The derivative terms, ux, uy , vx and vy , at interior points, were computed from u and v at time level n using compact difference method given below [64] and were treated as known coefficients in Equs. (50) and (51):

$$\beta f'_{i-2} + \alpha f'_{i-1} + f'_i + \alpha f'_{i+1} + \beta f'_{i+2} = C\frac{f_{i+3} - f_{i-3}}{6h} + B\frac{f_{i+2} - f_{i-2}}{4h} + A\frac{f_{i+1} - f_{i-1}}{2h}, \tag{52}$$

$$\alpha f_1' + \beta f_2' + \gamma f_3' = \frac{1}{h}(Af_1 + Bf_2 + Cf_3 + Df_4 + Ef_5 + Gf_6 + Hf_7), \quad (53)$$

After a Galerkin projection, Equs. (50) and (51) could form a 2N by 2N system:

$$\begin{bmatrix} \underline{\underline{M}} + \Delta t * \underline{\underline{M_1}} & \Delta t * \underline{\underline{M_2}} \\ \Delta t * \underline{\underline{M_3}} & \underline{\underline{M}} + \Delta t * \underline{\underline{M_4}} \end{bmatrix} \begin{bmatrix} \hat{U}^{n+1} \\ \hat{V}^{n+1} \end{bmatrix} = \begin{bmatrix} \underline{\underline{M}} * \hat{U}^n + \Delta t * \underline{F}^{n+1} \\ \underline{\underline{M}} * \hat{V}^n + \Delta t * \underline{G}^{n+1} \end{bmatrix} \quad (54)$$

where M is global mass matrix. M1, M2, M3 and M4 are constructed by following the same pattern as the products of global mass matrix and corresponding spatial differentiation (ux)n, (uy)n, (vx)n and (vy)n respectively.

To facilitate validation, we choose exact solutions of u, v as following:

$$u = sin(x-t)sin(y-t), \quad (55)$$

$$v = sin(\frac{x}{2} - t), \quad (56)$$

$$f(x,y,t) = -cos(x-t)sin(y-t) - sin(x-t)cos(y-t) + sin(x-t)cos(x-t)sin^2(y-t) + sin(\frac{x}{2} - t)sin(x-t)cos(y-t), \quad (57)$$

$$g(x,y,t) = -cos(\frac{x}{2} - t) + \frac{1}{2}sin(x-t)sin(y-t)cos(\frac{x}{2} - t), \quad (58)$$

and the initial and boundary conditions are acquired from Equs (55) and (56). After we computed the first derivatives of velocity components with respect to x and y, we use two dimensional Lagrangian interpolants to compute their values at Gauss-Lobatto-Legendre quadrature points. Figure 11a and 11b show the contour lines of u and v at time t=0.5. Figure 11c shows four elements and the interior quadrature points. Figure 11d shows the velocity vector field. For all elements, the highest order of polynomial is 8. We pre-compute ux, uy,

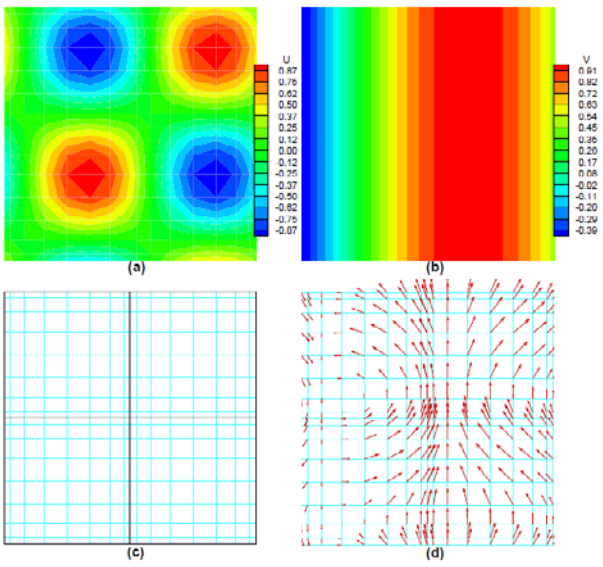

Figure 11: Two dimensional convection diffusion equation linearized with Method III: Velocity components u, v and computational mesh.

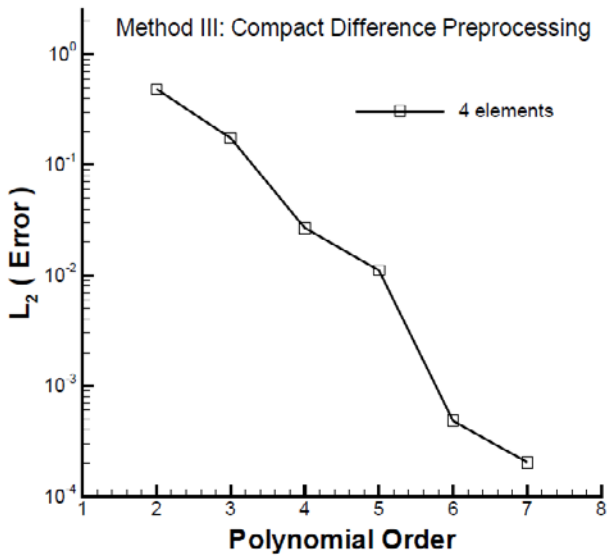

Figure 12: Two dimensional convection diffusion equations linearized with Method III: p-Refinement.

vx and vy with compact difference schemes of 8th order accuracy. Then we interpolate values of u and v on uniform grids to Gauss-Lobatto-Legendre points and use the nodal SEM with 8th order basis expansions, which are Lagrangian interpolants on Gauss-Lobatto-Legendre points. The values of u and v could be computed with high accuracy.

Figure 12 presents exponential convergence of L^2 norm of errors of u as the polynomial order varying from 2 to 8. We only derived and tested 8th compact difference schemes for pre-processing first derivatives. It has been observed that Method III works well for convection diffusion equations. For irregular domains, we have to use a mapping and pre-compute the first derivatives on a mapped grid or using full inclusion of metrics [17] which could add difficulty to Method III.

In summary, the approach of Compact Difference Preprocessing uses previously obtained values in prior time steps to compute unknowns explicitly. This way of handling nonlinearity provides convenience in decoupling PDE and especially for systems of PDE or multiple dimensions. It opens up the possibility of using an implicit method in time in which a large time step is feasible and is especially good for long time integration.

Using conservative form (Method IV): The one dimensional convection diffusion equation, Equ. (28), could be written in conservative form [65], since it is always conservative in one dimensional situation (See the Appendix in Section 4). The approaching of casting it into conservative form is called Method IV.

$$\frac{\partial u}{\partial t} + (\frac{u^2}{2})_x = u_{xx}. \quad (59)$$

We use the forward Euler scheme in time for the above equation:

$$\frac{u^{n+1} - u^n}{\Delta t} + (\frac{u^2}{2})_x^n = u_{xx}^n, \quad (60)$$

and discretize the whole domain into N elements. Within a typical element Ω_e, we expand terms such as un, $(u^2)n$ and un with the kth

order Lagrangian interpolants φ on the k+1 2 xxx Gauss-Lobatto-Legendre points:

$$u^n = \sum_{i=0}^{k} \hat{U}_i^n \phi_i(x), \tag{61}$$

$$\left(\frac{u^2}{2}\right)_x^n = \sum_{i=0}^{k} \left(\frac{\hat{U}_i^2}{2}\right)^n \phi_{i'}(x), \tag{62}$$

$$u_{xx}^n = \sum_{i=0}^{k} \hat{U}_i^n \phi_{i''}(x). \tag{63}$$

We substitute the above expansions into Equ. (60) and apply a Galerkin projection to obtain the weak form of Equ. (59) within Ω_e:

$$\sum_{i=0}^{k} \int_{\Omega_e} \phi_i(x)\phi_j(x)dx \frac{\hat{U}_i^{n+1} - \hat{U}_i^n}{\Delta t} + \sum_{i=0}^{k} \int_{\Omega_e} \phi_{i'}(x)\phi_j(x)dx \left(\frac{\hat{U}_i^2}{2}\right)^n = -\sum_{i=0}^{k} \int_{\Omega_e} \phi_{i'}(x)\phi_{j'}(x)dx \hat{U}_i^n, \quad j = 0,1...k-1,k, \tag{64}$$

Note that within each time step, (u2)n is obtained explicitly from un. Assembling Equ. (64) over all elements, we obtain the global form of Equ. (60):

$$\underline{\underline{M}}\,\underline{\hat{U}}^{n+1} = \underline{\underline{M}}\,\underline{\hat{U}}^n - \Delta t\left(\underline{\underline{K}}\left(\frac{\underline{\hat{U}}^2}{2}\right)^n + \underline{\underline{L}}\,\underline{\hat{U}}^n\right). \tag{65}$$

In which, M , K and L are global mass, convection and stiffness matrices, respectively.

Two dimensional convection diffusion equation: For two dimensional convection diffusion equations, Equ. (1), subject to some initial and boundary conditions, we assume u and v are velocity components along x and y directions, respectively, and write Equ. (1) in the component form:

$$\frac{\partial u}{\partial t} + uu_x + vu_y = \mu(u_{xx} + u_{yy}), \tag{66}$$

$$\frac{\partial v}{\partial t} + uv_x + vv_y = \mu(v_{xx} + v_{yy}). \tag{67}$$

For Equs. (66) and (67), Method IV is only conditionally conservative [58]. The proof of this property is given in the Appendix in Section 4. Now we change Equs. (66) and (67) into the partial conservation form:

$$\frac{\partial u}{\partial t} + \left(\frac{u^2}{2}\right)_x + vu_y = \mu(u_{xx} + u_{yy}), \tag{68}$$

$$\frac{\partial v}{\partial t} + uv_x + \left(\frac{v^2}{2}\right)_y = \mu(v_{xx} + v_{yy}). \tag{69}$$

Unless we know that u=(u, v) is a conservative vector field, i.e., $u_y=v_x$, or incompressible flow field, i.e., $u_x+v_y=0$, we cannot take it for granted to change coupled terms vu_y and uv_x into conservative forms. In Equ. (68), the coupled term vu_y remains nonlinear. We could linearize it by treating v explicitly as known coefficient for u_y in Method IV. Similarly, the coupled term uv_x in Equ. (69) is linearized by treating u explicitly as the known coefficient for v_x.

In terms of spatial discretization, we first use rectangular elements (irregular elements are discussed later) and choose the kth order Lagrangian interpolants, Φ, on Gauss-Lobatto-Legendre points as the basis functions in both x and y directions, although different orders of bases could be used in x and y. In a typical element Ω_e, we expand u(x, y) as below:

$$u(x,y) = \sum_{i=0}^{k}\sum_{j=0}^{k} \hat{U}_{ij}\Phi_i(x)\Phi_j(y). \tag{70}$$

Similarly for v. The derivatives of u with respect to x and y, respectively, are:

$$\frac{\partial u(x,y)}{\partial x} = \sum_{i=0}^{k}\sum_{j=0}^{k} \hat{U}_{ij}\frac{\partial \Phi_i(x)}{\partial x}\Phi_j(y), \tag{71}$$

$$\frac{\partial u(x,y)}{\partial y} = \sum_{i=0}^{k}\sum_{j=0}^{k} \hat{U}_{ij}\Phi_i(x)\frac{\partial \Phi_j(y)}{\partial y}. \tag{72}$$

We perform a Galerkin projection with the test function w:

$$(x,y) = \Phi_{pq}(x,y) = \sum_{p=0}^{k}\sum_{q=0}^{k} \Phi_p(x)\Phi_q(y) \tag{73}$$

to acquire a linear algebraic system about global variables.

Numerical examples one dimensional example: Consider the same 1D problem, Equs. (15) and (16) as in Method I, we use Method IV to solve it. We discretize the whole domain into 2 elements and use the forward Euler method in time. To achieve exponential convergence in space, we use SEM and perform a p-refinement by varying the order of polynomial basis functions. Figure 13 shows the exponential convergence with the order of polynomial basis functions versus the L2 norm of point-wise errors. Notice that, when polynomial order is less than 4, the exponential convergence of L2 norm of errors is not reached, because the approximation space have not been well set up yet. When the polynomial order is greater than 16, the L2 norm of error will not decrease any more and oscillate around 10^{-13} due to double decision rounding error.

Two Dimensional Rectangular Domain

The two dimensional computational domain is set to be Ω={(x, y) : 0 ≤ x, y ≤ 1, t ≥ 0}. Equ. (1) is the governing equation and the initial and boundary conditions are generated from the exact solution of u and v which are given below:

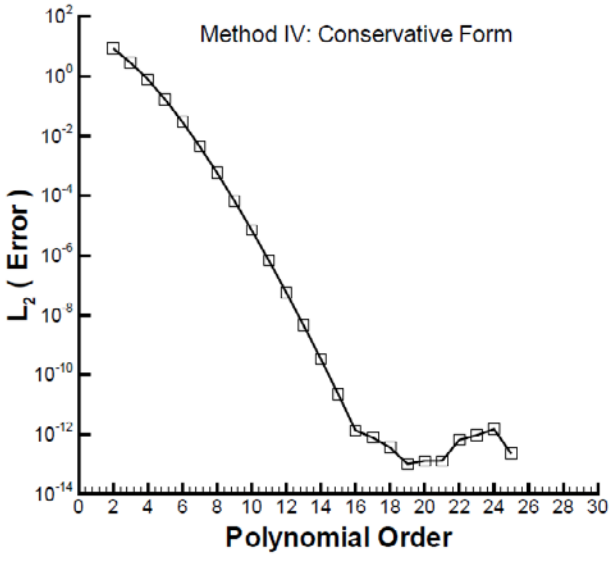

Figure 13: One dimensional convection diffusion equation (28) with Method IV: p-Refinement.

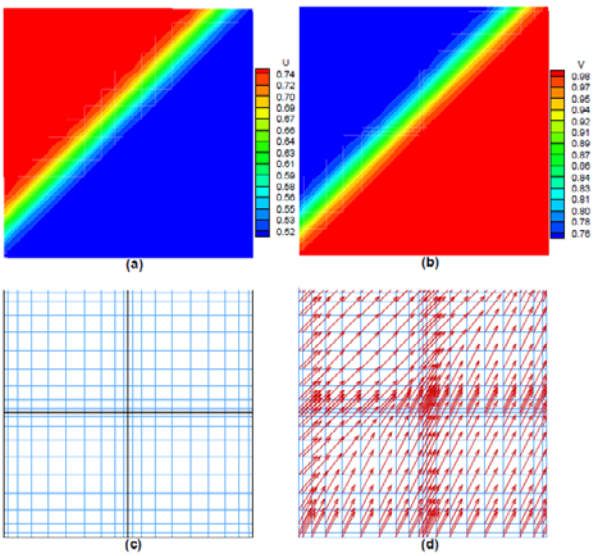

Figure 14: Two dimensional convection diffusion equation, Equ. (1) with Method IV: Velocity components u, v with 4 elements N=4 and the polynomial order Po=10.

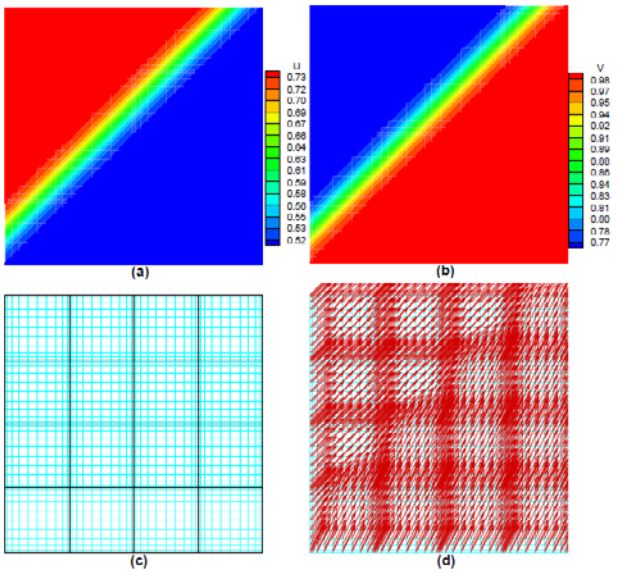

Figure 15: Two dimensional convection diffusion equation, Equ. (1) with Method IV: Velocity components u, v with 16 elements N=16 and the polynomial order Po=10.

$$u(x,y,t)=\frac{3}{4}-\frac{1}{4(1+e^{200(-t-4x+4y)/32})} \tag{74}$$

$$v(x,y,t)=\frac{3}{4}+\frac{1}{4(1+e^{200(-t-4x+4y)/32})}. \tag{75}$$

We use Method IV to solve the 2D problem on a rectangular domain, which was divided into 4 and 16 elements, respectively. Numerical results on 4 elements are shown in Figure 14 and on 16 elements are shown in 15. In both figures, the top plots ((a) and (b)) show contour lines of the velocity component u and v, separately at t = 0.5. The plot (c) illustrates the elements and the quadrature points. The plot (d) shows the velocity vector field. The Reynolds number is about 200; therefore, the magnitude of solution waves decrease with time due to viscous dissipation. Figure 15 gives more details about the convective flow since it has higher resolution than Figure 14 using 4 elements.

The exponential convergence in space is shown in Figure 16. Polynomial order were up to 40. The time step Δt was chosen to satisfy the CFL and diffusion conditions. Specifically, we chose Δt=0.001 and compared numerical result of u at time t=0.001 with the exact solution in the infinity norm. Using 16 elements gives faster convergence rate than using 4 elements and the error reaches machine zero at the polynomial order of 24.

Two-dimensional Irregular Domain

For the same problem as given in Equs. (66) and (67) but defined on an irregular domain with similar initial and boundary conditions that we designed here, we map the physical coordinates of (x, y) of a quadrilateral element Ωe to the coordinates (ξ1, ξ2) of a standard element (square) Ωst with the Jacobian:

$$J_{2D}=\begin{vmatrix}\frac{\partial x}{\partial \xi_1} & \frac{\partial x}{\partial \xi_2}\\ \frac{\partial y}{\partial \xi_1} & \frac{\partial y}{\partial \xi_2}\end{vmatrix}=\frac{\partial x}{\partial \xi_1}\cdot\frac{\partial y}{\partial \xi_2}-\frac{\partial x}{\partial \xi_2}\cdot\frac{\partial y}{\partial \xi_1}. \tag{76}$$

For any convex quadrilateral element, we denote its vertices as A, B, C and D, and their coordinates as A_x, A_y, B_x, B_y, C_x, C_y, D_x and D_y. The components of the Jacobian are:

$$\frac{\partial x}{\partial \xi_1}=\frac{\xi_2(A_x-B_x+C_x-D_x)+(-A_x+B_x+C_x-D_x)}{4}, \tag{77}$$

$$\frac{\partial y}{\partial \xi_1}=\frac{\xi_2(A_y-B_y+C_y-D_y)+(-A_y+B_y+C_y-D_y)}{4}, \tag{78}$$

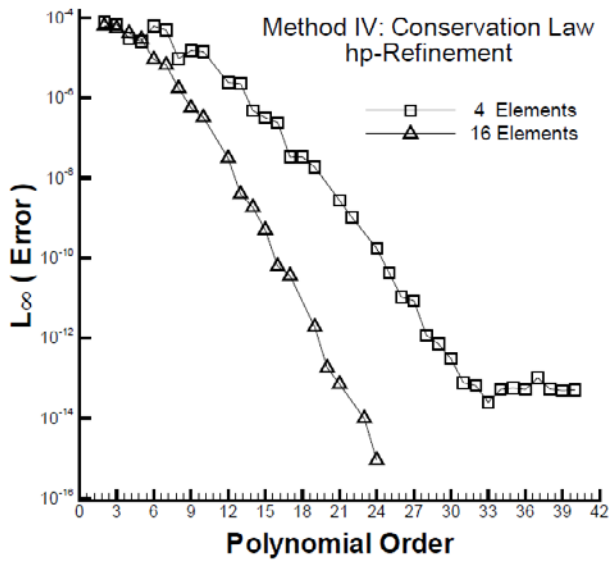

Figure 16: Two dimensional convection diffusion equation, Equ. (1) with Method IV: hp-refinement in rectangular domain

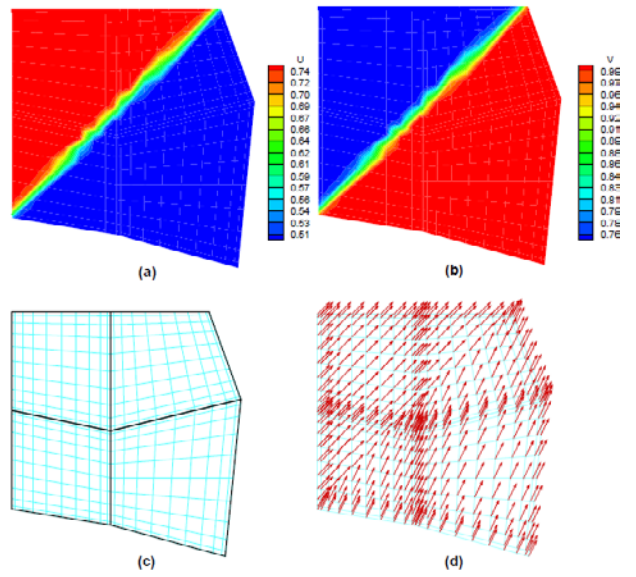

Figure 17: Two dimensional convection diffusion equation, Equ. (1) with Method IV: Velocity components u, v and quadrature points in irregular domain

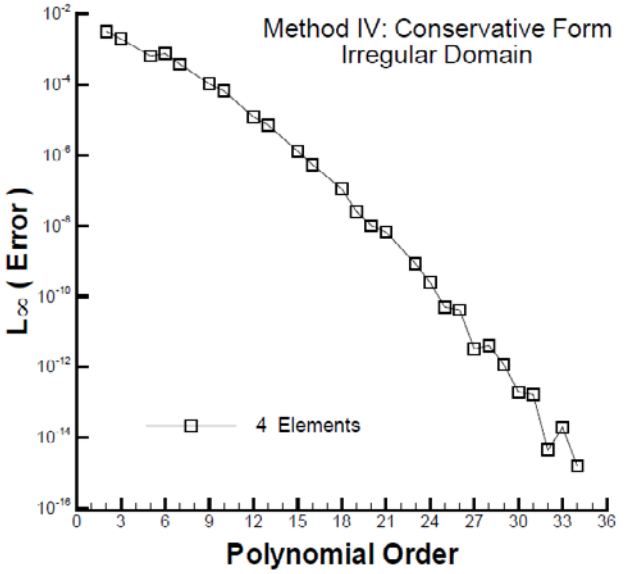

Figure 18: Two dimensional convection diffusion equation, Equ. (1) with Method IV: p-refinement in irregular domain

$$\frac{\partial x}{\partial \xi_2} = \frac{\xi_1(A_x - B_x + C_x - D_x) + (-A_x - B_x + C_x + D_x)}{4}, \quad (79)$$

$$\frac{\partial y}{\partial \xi_2} = \frac{\xi_1(A_y - B_y + C_y - D_y) + (-A_y - B_y + C_y + D_y)}{4}. \quad (80)$$

With a known mapping and Jacobian, we could construct the mass, convection, and stiff- ness matrices for any convex quadrilateral element. Similar procedures were performed and Method IV was used to obtain numerical solutions.

Figure 17 shows the velocity component u, v at time t=0.5, quadrature points, and the velocity vector field. In this figure, the domain was divided into 4 elements and the spectral nodal element method with polynomials of order Po=10 in each direction and each element was used as in the simulation. We completed a very high order solution with nodal basis polynomial functions of order Po=35 using 36 nodes, i.e., Gauss-Legendre-Lobatto points in each direction. This high accuracy solution was used as our "exact" solution to compute nodal errors for lower order runs. Figure 18 shows the L∞ norm of errors at all nodes versus the order of expansion polynomial order. A spectral convergence rate was shown in this figure. The solutions are consistent as the order increases.

In summary, Method IV is capable of capturing discontinuity developed in the solution and preserves good convergence rates which is unaffected by irregularity in geometry. This approach is very effective for strong hyperbolic equations provided that the conservative form exists.

Conclusion

In this paper, four different approaches for solving convection diffusion equations in 1D and 2D are discussed. Method I, II and III use linearization procedures which give the advantage of using an implicit time discretization and a large time step without the stability constraint. For the linearized problem, an implicit method in time such as Crank-Nicholson or backward Euler is preferred especially for long time integration. Spatial discretization methods used in this paper include spectral modal and nodal element, finite element, and compact difference. Of course other methods such as discontinuous Galerkin finite element or finite volume etc. are possible.

Since Method I uses a Taylor expansion in time to linearize a convective term, the trun- cation rror in time is consistent with the temporal discretization method. This method is simple and effective for strongly nonlinear terms such as (un+1)2un+1 as long as the first temporal derivative could be expressed in terms of the rest terms in the original PDE. The drawback of Method I is that it requires an explicit expression for the time derivative which could be difficult to derive for PDE with second or higher order temporal derivative and may introduce complexity in linearized equation.

Method II is an excellent approach for convection diffusion equations and possibly for other nonlinear equations if a Cole-Hopf transformation is applicable. The success of this method relies on performing a nonlinear transformation, which could reduce the original nonlinear vector form of PDE into a linear one. This is the major advantage of Method.

II. Apparently, both explicit and implicit time schemes can be used afterwards, although implicit is better in time integration. Nevertheless, the Cole-Hopf transformation is difficult for higher order spatial derivatives. Depending on the specific equation, it could be difficult to find analytical forms of transformed initial and boundary conditions. In general, it is not a trivial work to find a suitable Cole-Hopf transformation for a nonlinear equation.

Method III is a good idea in this paper which uses compact difference method to pre-compute first order spatial derivatives so that a convective term could be approximated by the pre-computed value multiplying the unknown which could be treated implicitly. Basically, Method III uses previously obtained values to compute a part of the unknown explicitly. Handling nonlinearity in this approach provides convenience in decoupling PDE and especially for systems of PDE or

multiple dimensions. It gives the convenience to use an implicit method in time in which a large time step is feasible and is especially good for long time integration. However, for irregular domains, Method III becomes complicated due to the spatial discretization and a mapping may be necessary. If SEM is used for numerical solutions, Method III requires deriving difference schemes on uniform grids to compute the first derivatives and then interpolate them to Gauss-Lobatto-Legendre nodes. Method III could be improved by using schemes of full inclusion of metrics [16,17] on non-uniform grids or an efficient mapping between grids to compute the first derivatives.

Compared with the above approaches, in terms of convenience and efficiency, Method IV could be superior in dealing with strong nonlinearity and complex geometry. It is straight forward for general weighted residual methods and is simple for multiple dimensions, since generating conservative form does not rely on mesh grids. However, the disadvantage is that using conservative forms to treat nonlinear terms could make the state vector and the matrix system much larger than without using it. This could rely on more memory in computation.

In general, for nonlinear PDE, if possible, one could use Method II and design a Cole-Hopf Transformation, to reduce a multi-dimensional nonlinear PDE into a scalar linear PDE so as to bypass nonlinearity. Depending on the dimension and problem, alternatively, one could select an appropriate linearization with controlled errors to open an option for implicit time treatment, which relax the constraint on time step size and could be beneficial to long time integration. For solutions develops discontinuity or there is discontinuous initial or boundary conditions, Method IV, using conservative form is very effective and capable of capturing discontinuity. The accuracy is usually unaffected by complexity of geometry or discontinuity in the domain.

Appendix

For Method IV, the one dimensional convection diffusion equation, Equ. (28), is always conservative. To prove this claim, we use the chain rule backward for the convective term:

$$\frac{\partial u}{\partial t}+\left(\frac{u^2}{2}\right)_x=u_{xx}, \tag{81}$$

then we introduce new variables F and G, respectively:

$$F=\frac{u^2}{2}, \tag{82}$$

$$G=u_x, \tag{83}$$

then we change Equ. (81) into the following:

$$\frac{\partial u}{\partial t}+F_x-G_x=0, \tag{84}$$

which is conservative. Therefore, the one dimensional convection diffusion equation is always conservative.

However, the two dimensional convection diffusion equations are not always conservative [58] because of the existence of coupled terms. Again, we use the chain rule backward in the two dimensional component-wise convection diffusion equation for the convective terms to obtain the partial conservative forms:

$$\frac{\partial u}{\partial t}+\left(\frac{u^2}{2}\right)_x+vu_y=\mu(u_{xx}+u_{yy}), \tag{85}$$

$$\frac{\partial u}{\partial t}+\left(\frac{u^2}{2}\right)_x+vu_y=\mu(u_{xx}+u_{yy}), \tag{86}$$

Focusing on two coupled terms above, we use the chain rule to form below equations:

$$(vu)_y=v_y u+vu_y, \tag{87}$$

$$(uv)_x=u_x v+uv_x. \tag{88}$$

If the irrotaional condition is valid for the vector field which is defined as u=(u, v), i.e., conservation of the total pressure in an irrotational flow [66], then we have the following:

$$\frac{\partial u}{\partial y}=\frac{\partial v}{\partial x}\Leftrightarrow u_y=v_x, \tag{89}$$

which is equivalent to the vorticity-free condition in fluid mechanics:

$$\nabla\times u=0, \tag{90}$$

By Equ. (89), we replace vuy with vvx in Equ. (85) and put vvx in conservative form:

$$\frac{\partial u}{\partial t}+\left(\frac{u^2}{2}\right)_x+\left(\frac{v^2}{2}\right)_x=\mu(u_{xx}+u_{yy}). \tag{91}$$

By Equ. (89), we replace uvx with uuy in Equ. (86) to and put uuy in conservative form:

$$\frac{\partial v}{\partial t}+\left(\frac{u^2}{2}\right)_y+\left(\frac{v^2}{2}\right)_y=\mu(v_{xx}+v_{yy}). \tag{92}$$

Now Equs. (91) and (92) are both conservative.

Alternatively, if the incompressible flow condition is satisfied by the vector field which is defined as u = (u, v), i.e., conservation of mass, then we have the following equations:

$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}=0\Leftrightarrow\frac{\partial u}{\partial x}=-\frac{\partial v}{\partial y}\Leftrightarrow u_x=-v_y \tag{93}$$

which is equivalent to the divergence-free condition in fluid mechanics:

$$\nabla\cdot u=0, \tag{94}$$

then we replace −vy with ux in the variation of the convective term vuy in Equ. (85) so that vuy becomes:

$$vu_y=(vu)_y-v_y u=(vu)_y+u_x u=(vu)_y+\left(\frac{u^2}{2}\right)_x, \tag{95}$$

then Equ. (85) becomes:

$$\frac{\partial u}{\partial t}+\left(\frac{u^2}{2}\right)_x+\left[(vu)_y+\left(\frac{u^2}{2}\right)_x\right]=\frac{\partial u}{\partial t}+(u^2)_x+(uv)_y=\mu(u_{xx}+u_{yy}). \tag{96}$$

Now the new form of Equ. (85) is already in conservative form. Similarly, we replace ux with −vy in the variation of the convective term uvx in Equ. (86) so that uvx becomes:

$$uv_x=(uv)_x-u_x v=(uv)_x+vv_y=(uv)_x+\left(\frac{v^2}{2}\right)_y, \tag{97}$$

then Equ. (86) becomes:

$$\frac{\partial v}{\partial t}+\left[(uv)_x+\left(\frac{v^2}{2}\right)_y\right]+\left(\frac{v^2}{2}\right)_y=\frac{\partial v}{\partial t}+(uv)_x+(v^2)_y=\mu(v_{xx}+v_{yy}). \tag{98}$$

Now the new form of Equ. (86) is also in conservative form. To summarize these proofs:

1. The one dimensional convection diffusion equation is conservative.

2. If ∇×u=0 is valid, then the dimensional convection diffusion equation is in conservative form:

$$\frac{\partial u}{\partial t}+\left(\frac{u^2}{2}\right)_x+\left(\frac{v^2}{2}\right)_x=\mu(u_{xx}+u_{yy}), \tag{99}$$

$$\frac{\partial v}{\partial t}+\left(\frac{u^2}{2}\right)_y+\left(\frac{v^2}{2}\right)_y=\mu(v_{xx}+v_{yy}). \tag{100}$$

If $\nabla \cdot u = 0$ is valid, then the two dimensional Burger's equation is in conservative form:

$$\frac{\partial u}{\partial t}+(u^2)_x+(uv)_y=\mu(u_{xx}+u_{yy}), \tag{101}$$

$$\frac{\partial v}{\partial t}+(uv)_x+(v^2)_y=\mu(v_{xx}+v_{yy}). \tag{102}$$

Acknowledgement

This study was supported by National Science Foundation under grants DMS-1115546, DMS-1115527, and Louisiana Board of Regents grant LEQSF (2007-10)-RDA-22. We thank the support from the Louisiana Optical Network Initiative (LONI) and Center for Computation and Technology, Louisiana State University. We are especially grateful to the encouragement from Professor Chi-Wang Shu (Brown University) and his continued constructive comments and insightful advice to improve the quality of this research.

Reference

1. Richard C, David H (1962) Methods of mathematical physics.
2. Hopf E (1950) The partial differential equation ut + uux=μuxx, Comm. Pure Appl Math 3: 201-230.
3. Zhu H (2010) Numerical solutions of two-dimensional Burgers' equation by discrete Adomian decomposition method. Com Math App 60: 840-848.
4. Bahadir (2003) A fully implicit finite-difference scheme for two-dimensional Burgers' equations. Appl Math Comput 131-137.
5. Srivastava V (2011) A comparison of finite element and finite difference solutions of the one-and two-dimensional Burgers' equations, Int J Scient Eng Res 2
6. Burden R, Faires J (2005) Numerical Analysis, 8th Edition, Thomson Brooks/ Cole, Belmont, CA, USA
7. Pepper D, Heinrich J (2006) The Finite Elements Method: Basic Concepts and Applications. Taylor and Francis, USA
8. Jiang G, Shu C (2000) Efficient implementation of weighted ENO schemes.
9. Shu C (1987) TVB boundary treatment for numerical solutions of conservation laws. Mathematics of Computation 49: 123-134.
10. Shu C (2001) An overview on high order numerical methods for convection dominated pdes, Hyperbolic Problems: Theory, Numerics, Applications.
11. Shu C (1987) TVB uniformly high order schemes for conservation laws. Mathematics of Computation 49: 105-121.
12. Xing Y, Shu C (2005) High order finite difference WENO schemes with the exact conservation property for the shallow water equations, J Com Phy 208: 206-227.
13. LeVeque R (1992) Numerical Methods for Conservation Laws. Birkhauser Verlag, Zurich.
14. Dai W, Nassar R (2001) A compact finite difference scheme for solving a one-dimensional heat transport equation at the microscale. J Com App Mathe 132: 431-441.
15. Gamet L, Ducros F, Nicoud F, Poinsot T (1999) Compact finite difference schemes on non-uniform meshes. Application to direct numerical simulations of compressible flows. Int J Numerical Methods in Fluids 29: 159-191.
16. Lele S (1992) Compact finite difference schemes with spectral-like resolution. J com phy.
17. Liu D, Kuang W, Tangborn A (2009) High-order compact implicit difference methods for parabolic equations in Geodynamo simulation.
18. Mahesh K (1989) A family of high order finite difference schemes with good spectral resolution, J Com Phy 145: 332-358.
19. Aksan E (2005) A numerical solution of Burgers' equation by finite element method constructed on the method of discretization in time. App Mathe Com 170: 895-904.
20. Fletcher C (1983) Crank-Nicholson scheme for numerical solution of two-dimensional coupled Burgers' equations. J Com Phy 51: 159-188.
21. Patera A (1984) A spectral element method for fluid dynamics-Laminar flow in a channel expansion. J Com Phy 54: 468-488.
22. Karniadakis G, Sherwin S (2005) Spectral/hp Element Methods for Computational Fluid Dynamics. Oxford University Press,USA
23. Mathews J, Fink K (2004) Numerical Method Using Matlab. Prentice Hall, USA
24. Pozrikidis C (2005) Introduction to Finite and Spectral Element Method using MATLAB. Taylor and Francis Group, USA.
25. Beskok A, Warburton T (2002) An unstructured hp fnite-element scheme for fluid flow and heat transfer in moving domains. J Comp Phys 174: 492-509.
26. Dong S, Liu D, Maxey M, Karniadakis G (2004) Spectral distributed Lagrange multiplier method: Algorithm and Benchmark test. J Com Phy 195: 695-717.
27. Giraldo F, Warburton T (2005) A nodal triangle-based spectral element method for the shallow water equations on the sphere. J Comp Phys 207: 129-150.
28. Hesthaven J, Gottlieb D (1999) Stable spectral methods for conservation laws on triangles with unstructured grids. J Comput Methods Appl Mech Engg 175: 361-381.
29. Huang L, Chen Q (2009) A spectral collocation model for solitary wave attenuation and mass transport over viscous fluid mud. J Eng Mechanics 135: 881-891.
30. Karniadakis G, Israeli M, Orszag S (1991) High-order splitting methods for the incompressible Navier-Stokes equations. J Comp Phys 97: 414-443.
31. Liu D, Chen Q, Wang Y (2011) Spectral element modeling of sediment transport in shear flows. J Comput Methods Appl Mech Engrg 200: 1691-1707.
32. Liu D, Keaveny E, Maxey M, Karniadakis G (2009) Force-coupling method for flows with ellipsoidal particles. J Computational Physics 228: 3559-3581.
33. Xia M, Symeonidis V, Karniadakis G (2003) A spectral vanishing viscosity method for stabilizing viscoelastic flows, J Non-Newtonian Fluid Mechanics 115: 125-155.
34. Xia M, Karniadakis G (2002) A low-dimensional model for simulating three-dimensional cylinder flow, Journal of Fluid Mechanics 458: 181-190.
35. Beskok A, Karniadakis G (1999) A model for flows in channels, pipes and ducts at micro and nano scales, J Microscale Thermophysical Engineering 3: 43-77.
36. Karniadakis G, Beskok A (2002) Microflows: Fundamentals and Simulation. Springer-Verlag, New York.
37. Karniadakis G, Beskok A, Aluru N (2005) Microflows and Nanoflows: Fundamentals and Simulation. Springer, New York.
38. Liu D, Lvov Y, Dai W (2011) Joint simulations of confined diffusion inside nanotubules. J Computational and Theoretical Nanoscience 8: 1-11.
39. Liu D, Maxey M, Karniadakis G (2002) A fast method for particulate microflows. J Micro-electromechanical System 11: 691-702.
40. Liu D, Maxey M, Karniadakis G (2004) Modeling and optimization of colloidal micro-pumps. J Micromechanics and Microengineering 14: 567-575.
41. Liu D, Maxey M, Karniadakis G (2005) Simulations of dynamic self-assembly of paramagnetic microspheres in confined microgeometries. J Micromechanics and Microengineering 15: 2298-2308.
42. Bellamkonda R, John T, Mathew B, DeCoster M, Hegab H, Davis D (2010)

Fabrication and testing of a CoNiCu/Cu CPP-GMR nanowire-based microfluidic biosensor. J Micromechanical and Microengineering 20: 025012.

43. Cox B, Liu D, Davis D (2012) GMR sensors: Technologies and medical applications, Journal of Recent Patents on Nanomedicine 1: 130-137.
44. Cockburn B, Karniadakis G, Shu C (2000) Discontinuous Galerkin Methods. Theory, Computation and Applications, Springer-Verlag, Berlin.
45. Hesthaven J, Warburton T (2008) Nodal Discontinuous Galerkin Methods: Algorithms,Analysis, and Applications. Springer-Verlag, New York.
46. Pietro DD, Ern A (2011) Mathematical Aspects of Discontinuous Galerkin Methods. Springer-Verlag, Berlin.
47. Cockburn B, Lin S, Shu C (1989) TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: one dimensional systems. J Computational Physics 84: 90-113.
48. Cockburn B, Shu C (1989) TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws II: general framework, Mathematics of Computation 52: 411-435.
49. Engsig-Karup A, Hesthaven J, Bingham H, Madsen P (2006) Nodal DG-FEM solutions of high-order Boussinesq-type equations. J Eng Math 56: 351-370.
50. Engsig-Karup A, Hesthaven J, Bingham H, Warburton T (2008) Nodal DG-FEM solutions of high-order Boussinesq-type equations. Coast. Eng. 55: 197-208.
51. Eskilsson C, Sherwin S (2002) A discontinuous spectral element model for Boussinesq-type equations. J Sci Comp 17: 143-153.
52. Eskilsson C, Sherwin S (2006) Spectral/hp discontinuous Galerkin methods for modelling 2D Boussinesq equations, J Comp Phys 212: 566-589.
53. Jiang B, Carey G (1988) A stable least-squares finite element method for non-linear hyper-bolic problems, International journal for numerical methods in fluids 8: 933-942.
54. Li B (2006) Discontinuous Finite Elements in Fluid Dynamics and Heat Transfer.
55. Lax P, Wendroff B (1960) Systems of conservation laws, Comm. Pure Appl Mathematics 13: 217-237.
56. Lax P, Wendroff B (1964) Difference schemes for hyperbolic equations with high-order of accuracy. Comm Pure Appl Mathematics 17: 381-398.
57. Richtmyer R, Morton K (1967) Difference Methods for Initial-Value Problems. John Wiley and Sons, New York.
58. Sheu T, Chen C, Hsieh L (2001) Development of a sixth-order two-dimensional convection-diffusion scheme via Cole-Hopf transformation, Comput Methods Appl Mech Engrg 191: 2979-2995.
59. Sachdev P (1978) A generalized Cole-Hopf transformation for nonlinear parabolic and hyper-bolic equations, J Applied Mathematics and Physics 29: 963-970.
60. Whitham G (1974) Linear and nonlinear wave. John Wiley and Sons, New York.
61. Chu C (1965) A class of reducible system of quasi-linear partial differential equations, Quart Appl Math 23: 275-278.
62. Roache P (1976) Computational Fluid Dynamics. Hermosa Publishers, Albuquerque.
63. Gaitonde D, Visbal M (1998) High-order schemes for Navier-Stokes equations: Algorithm and implementation into FDL3DI.
64. Shu C (2012) Efficient algorithm for solving partial differential equations with discontinuous solutions.
65. Mohanty A (1994) Fluid Mechanics. Prentice-Hall of India Private Limited, New Delhi.