Preprint accepted for publication in Computers and Fluids, 2016

http://dx.doi.org/10.1016/j.compfluid.2015.10.020

Third-Order Active-Flux Scheme for Advection Diffusion: Hyperbolic Diffusion, Boundary Condition, and Newton Solver

Hiroaki Nishikawa *

National Institute of Aerospace, Hampton, VA 23666

Philip L. Roe

Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI 48109

Abstract

In this paper, we construct active flux schemes for advection diffusion. Active flux schemes are efficient third-order finite-volume-type schemes developed thus far for hyperbolic systems. This paper extends the active flux schemes to advection diffusion problems based on a first-order hyperbolic system formulation that is equivalent to the advection-diffusion equation in pseudo steady state. An active flux scheme is first developed for a generic hyperbolic system with source terms, applied then to a hyperbolized diffusion system, extended to advection diffusion by incorporating the advective term as a source term, and enabled for unsteady problems by implicit time integration. Boundary conditions are discussed in relation to a non-uniqueness issue, and a weak boundary condition is shown to resolve the issue. Both for steady problems and for sub-iterations within unsteady problems, a globally coupled system of residual equations is solved by Newton's method. Numerical results show that third-order accuracy is obtained in both the solution and the gradient on irregular grids with rapid convergence of Newton's method, i.e., four or five residual evaluations are sufficient to obtain the design accuracy in both space and time.

1 Introduction

Active flux schemes have been developed for hyperbolic systems of conservation laws in Refs.[1, 2], built upon Scheme V of Van Leer [3], as a viable alternative to other high-order methods. Active flux schemes are

*Corresponding author. (hiro@nianet.org)

finite-volume-based compact high-order schemes. These schemes are substantially different from other high-order schemes and have attractive features for a practical implementation. First, active flux schemes do not rely on a typical one-dimensional flux across a control-volume face, but incorporate multi-dimensional physics into the residual and do not introduce unphysical discontinuities into solution [4]. The numerical flux at a face is determined not by solving a one-dimensional Riemann problem, but calculated by the method of spherical mean, which is an exact solution to a multi-dimensional initial-value problem. It is equivalent to a solution to the characteristic equations in one dimension. Second, the memory requirement is much reduced compared with discontinuous Galerkin methods due to sharing of degrees of freedom among elements. In addition to cell-averages, active flux schemes carry point-values at faces; the latter are shared by adjacent cells, thus resulting in 2 degrees of freedom per cell for third-order accuracy in one dimension, 3 in two dimensions, and only 2.2 in three dimensions. The active flux methodology has been developed for systems of hyperbolic conservation laws in Refs.[1, 2], but its extension to diffusive equations has not been well studied yet. Towards the development of practical third-order active-flux schemes for viscous flow simulations, in this paper, we focus on the construction of active-flux schemes for diffusion and advection diffusion problems.

One possible approach to the construction of active flux schemes for diffusion is the recovery approach proposed in Ref.[5]. Specifically, a quartic polynomial is constructed over two adjacent cells, based on a quadratic polynomial defined within each cell, and a diffusive flux is directly evaluated by differentiation at the face. However, our experience shows that the resulting explicit time-stepping scheme is subject to a severe stability restriction, and thus limiting its potential use (an analysis is given Ref.[6]). In order to develop high-order diffusion schemes while preserving the advantages of the active flux scheme, we consider the construction of diffusion schemes based on the first-order hyperbolic system method [7] where the diffusion equation is discretized in the form of a first-order hyperbolic system. One of the advantages of this method is that schemes developed for hyperbolic systems can be directly applied to diffusion. The method was first introduced for this purpose in Ref.[7], extended to the advection-diffusion equation in Ref.[8], to the Navier-Stokes equations in Ref.[9], to time-dependent advection-diffusion problems in Ref.[10, 11]. At the same time, the method has also been employed for the development of first-, second-, and third-order edge-based finite-volume schemes as in Refs.[12, 13, 14, 15, 16], as well as high-order residual-distribution schemes in Refs.[10, 11, 17, 18]. Through these papers, the method has been shown to offer a number of advantages over conventional methods, not only the drastic simplification in discretization (i.e., advection scheme for diffusion), but also significant acceleration in steady convergence by explicit and implicit solvers, providing an equal order of accuracy for the solution and the gradients (viscous/heat fluxes) and exceptionally high-quality gradients on fully irregular grids (see e.g., [17]).

This paper demonstrates that the active flux scheme for diffusion can be constructed by applying the active flux scheme developed for hyperbolic systems as presented in Refs.[1, 2] in combination with the physical time integration by the backward difference formulas. In doing so, we have found that active flux schemes need a careful construction for hyperbolic systems with source terms, which include the hyperbolic diffusion system.

This paper shows how to construct active flux schemes for hyperbolic systems with source terms, and then how the resulting schemes can be immediately turned into diffusion schemes. Boundary conditions are also discussed in relation to a non-uniqueness problem. Active-flux schemes are shown to allow infinitely many solutions if boundary fluxes are fixed by boundary conditions. We demonstrate that the issue is successfully resolved by a weak boundary procedure.

The hyperbolic formulation of diffusion employed here and in Ref.[7] is related, as mentioned in Ref.[7], from the pioneering work of Cattaneo [19] and Vernotte [20]. In these references, the hyperbolic model for diffusion was introduced as an alternative model to resolve the paradox of the infinite propagation speed associated with the classical diffusion equation expressed by second-order derivatives. The equivalence between the hyperbolic model and the diffusion equation is established in the limit of vanishing relaxation time [21]. Therefore, in order to employ the hyperbolic model to solve the diffusion equation, the relaxation time needs to be very small, often leading to a hyperbolic system with stiff source terms. There have been efforts in constructing numerical schemes for such a model [22], and for a particular generalization to advection diffusion [22, 23, 24, 25], focusing on the treatment of the stiff source terms. Also, there has been a series of works [26, 27, 28, 29, 30, 31] concerning the use of Cattaneo's model for advection-diffusion problems targeting at applications in which the finite propagation speed has a physical importance. As stated in Ref.[31], their advection-diffusion system is not equivalent to the classical advection-diffusion equation with a second-derivative diffusion term in the steady state. In contrast, the hyperbolic method considered here deliberately designs a first-order hyperbolic system to recover the classical advection-diffusion equation in the steady state [8]; and the same for the Navier-Stokes equations [9, 15, 16]. In this context, the 'relaxation time' is a free parameter that can be chosen to enhance convergence to the steady state

Recently, high-order explicit time-integration schemes have been developed for advection-diffusion-reaction problems [32, 33, 34] based on a hyperbolic formulation similar to the one considered in Ref.[8]. In their work, the relaxation time is carefully defined to retain time accuracy with explicit time integration schemes. As a consequence, the explicit time step is subject to a typical $O(h^2)$-type restriction, (where h is a mesh spacing) although a larger time step is allowed compared with conventional schemes. In the hyperbolic method considered here, we choose an artificial relaxation time, so that explicit time-stepping schemes (used to reach the steady state) allow $O(h)$ time steps, or equivalently the condition number of a linearized system associated with implicit solvers becomes $O(1/h)$, not $O(1/h^2)$, and thus significantly improved convergence is achieved in iterative solvers. When a time-accurate solution is needed, we prefer to employ "sub-cycling", where each (large) time step is executed as the solution to a pseudo-steady problem, as discussed further below.

Given a successful construction of active-flux diffusion schemes, we discuss an extension to advection diffusion problems. It should be pointed out that the extension is not as straightforward as adding the diffusion scheme to the advection scheme. Such a naive extension will destroy third-order accuracy easily. This is a well-known issue for schemes that require compatible discretizations, including the residual-distribution method [35], the third-

order edge-based finite-volume method [13], and the active-flux method. One way to ensure the compatibility is to formulate the advection-diffusion equation as a single hyperbolic system [8]. Then, the construction of the active-flux scheme will be trivially simple for the advection-diffusion equation. However, this strategy is currently not applicable to the compressible Navier-Stokes equations because a complete characteristic decomposition has not been discovered yet for hyperbolic formulations of the compressible Navier-Stokes equations [9, 15, 16]. As a practical alternative, we propose a strategy of adding the advective term to the diffusion scheme as a source term. The idea is applied to the computation of the face values, and the cell-averages are updated by the usual finite-volume method with the sum of advective and diffusive fluxes to guarantee discrete conservation. We demonstrate that the resulting advection-diffusion scheme yields third-order accurate solution and gradients for both steady and unsteady advection diffusion problems.

This paper also presents a highly efficient Newton solver for a system of globally coupled residual equations, which needs to be solved both for steady problems and for sub-iterations within unsteady problems. It is shown that the convergence is significantly improved by Newton's method: four or five Newton iterations (i.e., only four or five residual evaluations) are sufficient both for steady problems and for the sub-iterations within unsteady problems that were mentioned above.

The present study focuses on linear advection-diffusion problems in one dimension to illustrate the basic ideas on extending the active flux method to diffusion and advection diffusion. All numerical results are therefore presented for linear problems. Essential ideas are applicable to nonlinear problems as well as higher dimensions, but algorithmic details remain a subject for future research and beyond the scope of the paper.

The paper is organized as follows. In Section 2, we present a hyperbolic diffusion system and its characteristic form. In Section 3, we construct an active-flux scheme for a generic hyperbolic system with source terms. In Section 4, we apply the developed active-flux scheme to the hyperbolic diffusion system. In Section 5, we extend the scheme to unsteady problems. In Section 6, we discuss an extension to advection diffusion problems. In Section 7, we describe a Newton solver used to solve the residual equations. In Section 8, we present numerical results for steady and unsteady advection diffusion problems. Finally, in Section 9, concluding remarks are given.

2 Hyperbolic Diffusion System

Consider the diffusion equation:

$$\partial_\tau u = \nu\, \partial_{xx} u + s_1, \tag{1}$$

where ν is a constant diffusion coefficient and $s_1 = s_1(x, u)$ is a source term. In this Section, and also in Sections 3 and 4, we focus on steady problems, and thus the variable τ is a pseudo time. Steady solutions can be obtained

by solving, instead of the diffusion equation, the following first-order hyperbolic system:

$$\partial_\tau \mathbf{u} + \partial_x \mathbf{f} = \mathbf{s}, \tag{2}$$

where

$$\mathbf{u} = \begin{bmatrix} u \\ p \end{bmatrix}, \quad \mathbf{f} = \begin{bmatrix} -\nu p \\ -u/T_r \end{bmatrix}, \quad \mathbf{s} = \begin{bmatrix} s_1 \\ s_2 \end{bmatrix}, \tag{3}$$

where $p = \partial_x u, \qquad s_1 = s, \qquad s_2 = -p/T_r$, and T_r is a relaxation time that remains to be chosen.

Note that the system is equivalent to the diffusion equation in the pseudo steady state for any nonzero value of T_r. The equivalence in the steady state is the key idea as first proposed in Ref.[7] for constructing diffusion schemes. Therefore, the relaxation time does not have to be determined by any physical consideration, but can be determined solely by numerical consideration, e.g., fast convergence to the steady state. A typical choice is the following:

$$T_r = \frac{L_r^2}{\nu}, \quad L_r = \frac{1}{2\pi}, \tag{4}$$

which can be derived by requiring Fourier modes to propagate to enhance the iterative convergence as discussed in Ref.[7, 8]. The same formula has been derived based on a similar argument applied to a first-order finite-volume scheme as described in Ref.[13]. It is noted that the relaxation time defined as above is $O(1)$ and independent of the mesh spacing. This is a distinguished feature of the hyperbolic method considered here.

Consider the flux Jacobian, $\mathbf{A} = \partial \mathbf{f}/\partial \mathbf{u}$,

$$\mathbf{A} = \begin{bmatrix} 0 & -\nu \\ -1/T_r & 0 \end{bmatrix}, \tag{5}$$

which has the following real eigenvalues:

$$\lambda_1 = -\lambda, \quad \lambda_2 = \lambda, \quad \lambda = \sqrt{\frac{\nu}{T_r}} = \frac{\nu}{L_r}, \tag{6}$$

and linearly independent right and left eigenvectors,

$$\mathbf{R} = \frac{1}{2}\begin{bmatrix} 1 & -1 \\ 1/L_r & 1/L_r \end{bmatrix}, \quad \mathbf{L} = \mathbf{R}^{-1} = \begin{bmatrix} 1 & L_r \\ -1 & L_r \end{bmatrix}. \tag{7}$$

Therefore, the system is hyperbolic. It is noted that the eigenvalues have the same magnitude with opposite

signs and thus the system describes a pair of waves traveling to the left and right at the same speed. For such a hyperbolic system, upwind schemes require no logic and lead to a symmetric stencil suitable for diffusion because $|\mathbf{A}| = \lambda\mathbf{I}$, where $\mathbf{I}$ is the identify matrix. The same is true for the hyperbolized viscous terms [9, 15, 16].

The system can be diagonalized by multiplying it by $\mathbf{L}$ from the left, which yields the set of characteristic equations:

$$\partial_\tau w_1 + \lambda_1 \partial_x w_1 = s_1^w, \quad \partial_\tau w_2 + \lambda_2 \partial_x w_2 = s_2^w, \tag{8}$$

where

$$w_1 = L_r p + u, \quad w_2 = L_r p - u, \tag{9}$$

$$s_1^w = L_r s_2 + s_1, \quad s_2^w = L_r s_2 - s_1. \tag{10}$$

In principle, active flux schemes developed for hyperbolic systems can be directly applied to the hyperbolic diffusion system, but the hyperbolic diffusion system has a source term; the one in the second equation of Eq.(2), i.e., s_2, is intrinsic to the hyperbolic diffusion system. Previous studies did not consider hyperbolic systems with source terms, and it has been found that active flux schemes require a careful construction for such systems. Also, it has been observed that improperly implemented boundary conditions can degrade the design accuracy of active flux schemes. In the next section, we discuss these issues in details for a generic hyperbolic system with source terms and then in Section 4 show how this applies to a diffusion problem.

3 Active Flux Scheme for Hyperbolic System with Source Terms

3.1 Hyperbolic System with Source Terms

Consider a 2×2 hyperbolic system with source terms in $x \in (0, 1)$:

$$\partial_\tau u - a\partial_x v = s_u, \tag{11}$$

$$\partial_\tau v - a\partial_x u = s_v, \tag{12}$$

where τ is a pseudo time, a is a positive constant, and s_u and s_v are source terms, which may depend on the solution (e.g., a reactive term). We are interested to compute the steady state solution with the Dirichlet boundary conditions:

$$u(0) = u(1) = v(0) = v(1) = 0. \tag{13}$$

Write the system in the vector form:

$$\partial_\tau \mathbf{u} + \partial_x \mathbf{f} = \mathbf{s}, \tag{14}$$

where

$$\mathbf{u} = \begin{bmatrix} u \\ v \end{bmatrix}, \quad \mathbf{f} = \begin{bmatrix} -av \\ -au \end{bmatrix}, \quad \mathbf{s} = \begin{bmatrix} s_u \\ s_v \end{bmatrix}. \tag{15}$$

The flux Jacobian, $\mathbf{A} = \partial \mathbf{f} / \partial \mathbf{u}$,

$$\mathbf{A} = \begin{bmatrix} 0 & -a \\ -a & 0 \end{bmatrix}, \tag{16}$$

has the following eigenvalues:

$$\lambda_1 = -a, \quad \lambda_2 = a, \tag{17}$$

and the right and left eigenvectors,

$$\mathbf{R} = \frac{1}{2} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \quad \mathbf{L} = \mathbf{R}^{-1} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}. \tag{18}$$

The characteristic equations are obtained by multiplying the system by $\mathbf{L}$ from the left:

$$\partial_\tau \mathbf{w} + \mathbf{\Lambda} \partial_x \mathbf{w} = \mathbf{s}^w, \tag{19}$$

where $\mathbf{w} = (w_1, w_2)$, and

$$\mathbf{w} = \mathbf{L}\mathbf{u} = \begin{bmatrix} u + v \\ u - v \end{bmatrix}, \quad \mathbf{\Lambda} = \begin{bmatrix} -a & 0 \\ 0 & a \end{bmatrix}, \quad \mathbf{s}^w = \mathbf{L}\mathbf{s}. \tag{20}$$

The characteristic system consists of two advection equations:

$$\partial_\tau (u + v) - a\, \partial_x (u + v) = s_1^w, \quad \partial_\tau (u - v) + a\, \partial_x (u - v) = s_2^w, \tag{21}$$

where $s_1^w = s_u + s_v$ and $s_2^w = s_u - s_v$, which is in the same form as the characteristic system of our target system (8).

3.2 Discretization

To discretize the system by the active flux scheme, we begin by storing the cell-averages within each cell and the point-values at each face in a one-dimensional grid. In each cell, we construct quadratic polynomials, $u(x)$ and $v(x)$, by interpolating the face values and requiring that the cell-average of the polynomial reduces to the cell-averaged solution. Integrating the system (14) over a space-(pseudo-)time control volume, we obtain

$$(\overline{\mathbf{u}}_j^{k+1} - \overline{\mathbf{u}}_j^k)\Delta x_j = -\Delta\tau \left[\mathbf{f}_{j+1/2} - \mathbf{f}_{j-1/2}\right] + \Delta\tau \int_{x_j-\Delta x_j/2}^{x_j+\Delta x_j/2} \mathbf{s}^{k+1}\, dx, \tag{22}$$

where $\overline{\mathbf{u}}_j^k$ and $\overline{\mathbf{u}}_j^{k+1}$ denote the cell-averaged solutions at pseudo time levels k and $k+1$, respectively, and $\Delta x_j = x_{j+1/2} - x_{j-1/2}$. The source term has been evaluated at the level $k+1$ for a point-implicit treatment, which is suitable for possible stiff source terms. Note that $\mathbf{f}_{j+1/2}$ and $\mathbf{f}_{j-1/2}$ are time-averaged fluxes, which can be evaluated by the trapezoidal or Simpson's rule and that the source term is independent of time. This leads to a pseudo-time marching scheme:

$$\overline{\mathbf{u}}_j^{k+1} = \overline{\mathbf{u}}_j^k - \frac{\Delta\tau}{\Delta x_j}\mathbf{Res}_j, \tag{23}$$

where $\mathbf{Res}_j$ represents the spatial discretization and is called the steady residual:

$$\mathbf{Res}_j = \mathbf{f}_{j+1/2} - \mathbf{f}_{j-1/2} - \int_{x_j-\Delta x_j/2}^{x_j+\Delta x_j/2} \mathbf{s}^{k+1}\, dx. \tag{24}$$

The source term integration may be evaluated exactly or by Simpson's rule. At this point, the discretization of the source terms is straightforward. It is the interface flux step where a special treatment is required.

In active flux schemes, the interface flux is found by integrating over each face the solution obtained from the characteristic equations (8). For example, at face $j+1/2$, we obtain

$$(w_1)_{j+1/2}^{k+1} = w_1(x_R) + \frac{1}{\lambda_1}\int_{x_R}^{x_{j+1/2}} s_1^w\, dx, \quad (w_2)_{j+1/2}^{k+1} = w_2(x_L) + \frac{1}{\lambda_2}\int_{x_L}^{x_{j+1/2}} s_2^w\, dx, \tag{25}$$

where $w_1(x_R) = u(x_R)+v(x_R)$ and $w_2(x_L) = u(x_L)-v(x_L)$ are interpolated values of the characteristic variables based on the quadratic functions $u(x)$ and $v(x)$ defined over the right and left cells, respectively. The coordinate $x_L = x_{j+1/2} - a\Delta\tau$, is the location on the left cell at time level k found by tracing back the characteristic from the face, and $x_R = x_{j+1/2} + a\Delta\tau$ is the location on the right cell found by tracing back the other characteristic. See Figure 1, which illustrates the case of the right face of the cell $j = 1$. In this work, we set $\Delta\tau = 0.5(h_{min}/a)$ in x_L and x_R, where h_{min} denotes the minimum cell volume in the grid. The source term integration can be

performed by the trapezoidal rule:

$$\int_{x_R}^{x_{j+1/2}} s_1^w \, dx = \frac{1}{2}\left[(s_1^w)_{j+1/2} + s_1^w(x_R)\right](x_{j+1/2} - x_R), \tag{26}$$

$$\int_{x_L}^{x_{j+1/2}} s_2^w \, dx = \frac{1}{2}\left[(s_2^w)_{j+1/2} + s_2^w(x_L)\right](x_{j+1/2} - x_L). \tag{27}$$

The system of integrated characteristic equations (25) can be written in terms of the original variables as

$$u_{j+1/2}^{k+1} + v_{j+1/2}^{k+1} = u(x_R) + v(x_R) + \frac{1}{\lambda_1}\int_{x_R}^{x_{j+1/2}} s_1^w \, dx, \tag{28}$$

$$u_{j+1/2}^{k+1} - v_{j+1/2}^{k+1} = u(x_L) - v(x_L) + \frac{1}{\lambda_2}\int_{x_L}^{x_{j+1/2}} s_2^w \, dx, \tag{29}$$

which can be solved for $u_{j+1/2}^{k+1}$ and $v_{j+1/2}^{k+1}$ at the face:

$$u_{j+1/2}^{k+1} = \frac{1}{2}\left[u(x_R) + u(x_L) + v(x_R) - v(x_L) + \frac{1}{\lambda_1}\int_{x_R}^{x_{j+1/2}} s_1^w \, dx + \frac{1}{\lambda_2}\int_{x_L}^{x_{j+1/2}} s_2^w \, dx\right], \tag{30}$$

$$v_{j+1/2}^{k+1} = \frac{1}{2}\left[v(x_R) + v(x_L) + u(x_R) - u(x_L) + \frac{1}{\lambda_1}\int_{x_R}^{x_{j+1/2}} s_1^w \, dx - \frac{1}{\lambda_2}\int_{x_L}^{x_{j+1/2}} s_2^w \, dx\right]. \tag{31}$$

We remark that the integration of the source terms along the characteristics is critical to keep the third-order accuracy of the active flux scheme. If the source term is ignored in the characteristic solve, the scheme will lose third-order accuracy. This is because the scheme will fail to preserve quadratic exact solutions if the source terms are ignored. Finally, given the face values, we can evaluate the time-averaged interface flux by the trapezoidal rule

$$\mathbf{f}_{j+1/2} = \begin{bmatrix} -a\,\dfrac{v_{j+1/2}^k + v_{j+1/2}^{k+1}}{2} \\ -a\,\dfrac{u_{j+1/2}^k + u_{j+1/2}^{k+1}}{2} \end{bmatrix}, \tag{32}$$

and similarly for $\mathbf{f}_{j-1/2}$. Note that Simpson's rule is not needed here and the trapezoidal rule is sufficient for third-order accuracy in the pseudo steady state. This completely defines the active flux scheme for the 2×2 hyperbolic system with source terms in the interior cells.

3.3 Boundary Conditions

In the cells adjacent to boundaries, the active flux scheme involves interface fluxes across boundaries. Consider the cell $j = 1$ adjacent to the left boundary ($x = 0$). The steady residual is given by

$$\mathbf{Res}_1 = \mathbf{f}_{3/2} - \mathbf{f}_b - \int_0^h \mathbf{s}^{k+1} \, dx = 0, \tag{33}$$

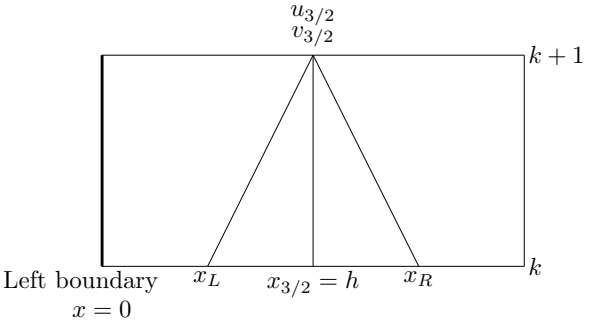

Figure 1: Finding the right face values $u_{3/2}$ and $v_{3/2}$ for the cell $j=1$, which is the left cell.

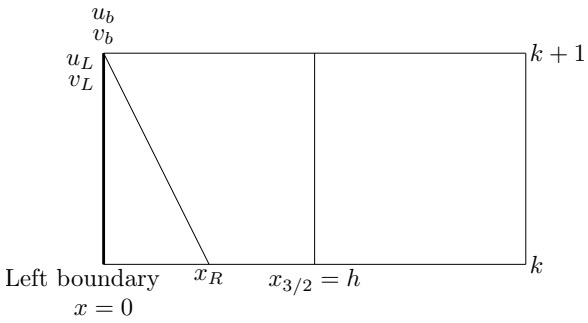

Figure 2: Finding the boundary values u_b and v_b for the cell $j=1$. Boundary condition is imposed on u_L and v_L

where h is the cell volume at $j=1$, and $\mathbf{f}_b$ is the boundary flux on the left of the boundary cell. The flux $\mathbf{f}_{3/2}$ is computed as described in the previous section (see Figure 1). If the boundary flux is fixed by the boundary condition, $\mathbf{f}_b = 0$, then we are left with

$$\mathbf{Res}_1 = \mathbf{f}_{3/2} - \int_0^h \mathbf{s}^{k+1}\, dx = 0. \tag{34}$$

This equation cannot be solved for the cell-averages, $\overline{\mathbf{u}}_1$ and $\overline{\mathbf{v}}_1$, because it involves only the combination $\overline{\mathbf{u}}_1 - \overline{\mathbf{v}}_1$ (see Eqs.(30) and (31) with $j=1$). In other words, it vanishes for infinitely many solutions that differ by a constant. Consequently, the scheme admits infinitely many solutions as it should, because we are only concerned to specify the incoming wave. A solution to this non-uniqueness is to specify the boundary conditions on the outer side of the boundary face (see Figure 2): set $u(x_L) = v(x_L) = 0$ and get the boundary values from the characteristic equations Eqs.(30) and (31):

$$u_b = \frac{1}{2}\left[u(x_R) + v(x_R) + \frac{1}{\lambda_1}\int_{x_R}^{x_{j+1/2}} s_1^w\, dx \right], \tag{35}$$

$$v_b = \frac{1}{2}\left[v(x_R) + u(x_R) + \frac{1}{\lambda_1}\int_{x_R}^{x_{j+1/2}} s_1^w\, dx \right], \tag{36}$$

where the source term integration has been ignored on the cell $j=0$ that does not exist, and R denotes the cell $j=1$ that is on the right side of the boundary face. In fact, this is a widely-used boundary procedure in finite-volume methods and also in high-order methods [36]: specify the outer value and let a numerical flux determine the boundary solution. The boundary flux $\mathbf{f}_b$ evaluated by these values will provide coupling between $\overline{\mathbf{u}}_1$ and $\overline{\mathbf{v}}_1$ in the residual equation:

$$\mathbf{Res}_1 = \mathbf{f}_{3/2} - \mathbf{f}_b - \int_0^h \mathbf{s}\, dx = 0, \tag{37}$$

which can now be solved to determine $\overline{\mathbf{u}}_1$ and $\overline{\mathbf{v}}_1$ uniquely.

The same procedure can be applied to the cell adjacent to the right boundary. This method gives third-order accurate face values (including boundary face values) as well as cell-averages in the steady state for any initial solution. It is simple to implement especially because we can use the same algorithm at all faces.

4 Active Flux Scheme for Diffusion

The hyperbolic diffusion system (2) is a 2×2 hyperbolic system with source terms. Therefore, it can be discretized in the same way as described in the previous section. Simply following the steps illustrated by Equations (25)-(31), we obtain the face values as

$$u_{j+1/2}^{k+1} = \frac{1}{2}\left[u(x_R) + u(x_L) + L_r(p(x_R) - p(x_L)) + \frac{1}{\lambda_1}\int_{x_R}^{x_{j+1/2}} s_1^w \, dx + \frac{1}{\lambda_2}\int_{x_L}^{x_{j+1/2}} s_2^w \, dx \right], \tag{38}$$

$$p_{j+1/2}^{k+1} = \frac{1}{2}\left[p(x_R) + p(x_L) + \frac{1}{L_r}\left(u(x_R) - u(x_L) + \frac{1}{\lambda_1}\int_{x_R}^{x_{j+1/2}} s_1^w \, dx - \frac{1}{\lambda_2}\int_{x_L}^{x_{j+1/2}} s_2^w \, dx \right) \right], \tag{39}$$

which are used to evaluate the interface flux. The steady solution can be computed by marching in the pseudo time:

$$\overline{\mathbf{u}}_j^{k+1} = \overline{\mathbf{u}}_j^k - \frac{\Delta\tau}{\Delta x_j}\mathbf{Res}_j, \tag{40}$$

where

$$\mathbf{Res} = \mathbf{f}_{j+1/2} - \mathbf{f}_{j-1/2} - \int_{x_j-\Delta x_j/2}^{x_j+\Delta x_j/2} \mathbf{s}^{k+1} \, dx, \tag{41}$$

$$\mathbf{f}_{j-1/2} = \begin{bmatrix} -\nu \dfrac{p_{j-1/2}^k + p_{j-1/2}^{k+1}}{2} \\ -\dfrac{u_{j-1/2}^k + u_{j-1/2}^{k+1}}{2T_r} \end{bmatrix}, \quad \mathbf{f}_{j+1/2} = \begin{bmatrix} -\nu \dfrac{p_{j+1/2}^k + p_{j+1/2}^{k+1}}{2} \\ -\dfrac{u_{j+1/2}^k + u_{j+1/2}^{k+1}}{2T_r} \end{bmatrix}. \tag{42}$$

The source term integration in the residual can be evaluated by Simpson's rule for s_1 and exactly for p that is quadratic inside the cell:

$$\int_{x_j-\Delta x_j/2}^{x_j+\Delta x_j/2} \mathbf{s}^{k+1} \, dx = \begin{bmatrix} \overline{s}_1{}_j^{k+1} \Delta x_j \\ -\overline{p}_j^{k+1} \Delta x_j / T_r \end{bmatrix}, \tag{43}$$

where

$$\overline{s}_1{}_j^{k+1} = \frac{s_1^{k+1}(x_{j+1/2}) + 4s_1^{k+1}(x_j) + s_1^{k+1}(x_{j-1/2})}{6}. \tag{44}$$

The resulting pseudo-time marching scheme becomes

$$\overline{u}_j^{k+1} = \overline{u}_j^k + \frac{\Delta\tau}{\Delta x_j}\left(\nu p_{j+1/2} - \nu p_{j-1/2} + \overline{s}_{1_j}^{k+1}\Delta x_j\right), \tag{45}$$

$$\overline{p}_j^{k+1} = \overline{p}_j^k + \frac{\Delta\tau}{\Delta x_j}\left(u_{j+1/2} - u_{j-1/2} - \overline{p}_j^{k+1}\Delta x_j\right)/T_r. \tag{46}$$

It is possible to reach the pseudo steady state by the pseudo-time marching scheme with a point-implicit procedure as implied in the above equations, but it can be inefficient especially when the grid is stretched to resolve a narrow layer as typically arises in advection diffusion problems. To obtain the steady solution efficiently, we directly solve the steady residual equations by Newton's method as we will describe later.

For boundary conditions, typically, two values are specified at the boundaries, e.g., $u(0) = \alpha$ and $u(1) = \beta$ or $\partial_x u(0) = \gamma$ and $u(1) = \beta$, etc., where α, β, and γ are constants. In the hyperbolic method, all conditions are of Dirichlet type because the condition on the gradient, such as $\partial_x u(0) = \gamma$, can be imposed via the variable p as $p(0) = \gamma$. In any case, there will always be one condition per boundary, which fixes one (or a combination) of the two variables; it may be considered as a characteristic condition as there is always one wave coming into the domain at a boundary. The other variable is then determined by the scheme, and may involve the solution at the previous pseudo-time step. For example, on the left boundary face, we specify $u(x_L) = \alpha$ as a boundary condition, and set

$$p(x_L) = p_b^k, \tag{47}$$

where p_b^k is the variable p at the face obtained at the previous pseudo-time step.

5 Physical Time Integration

In the hyperbolic method, time-accurate computation is possible by implicit time integration schemes as demonstrated in Refs.[10, 37] for residual-distribution schemes. Here, we follow Refs.[10, 37] and employ the backward difference formulas (BDF). Incorporating a physical time derivative discretized by the BDF scheme into the pseudo time scheme (45) and (46), we obtain

$$\overline{u}_j^{k+1} = \overline{u}_j^k + \frac{\Delta\tau}{\Delta x_j}\left(\nu p_{j+1/2} - \nu p_{j-1/2} - \partial_t \overline{u}^{k+1}\Delta x_j + \overline{s}_{1_j}^{k+1}\Delta x_j\right), \tag{48}$$

$$\overline{p}_j^{k+1} = \overline{p}_j^k + \frac{\Delta\tau}{\Delta x_j}\left(u_{j+1/2} - u_{j-1/2} - \overline{p}_j^{k+1}\Delta x_j\right)/T_r, \tag{49}$$

where the physical time derivative is given by the BDF formula:

$$\partial_t \overline{u}^{k+1} = \frac{\alpha \overline{u}_j^{k+1} + \alpha_n \overline{u}_j^n + \alpha_{n-1} \overline{u}_j^{n-1} + \alpha_{n-2} \overline{u}_j^{n-2}}{\Delta t}, \tag{50}$$

for a constant physical time step Δt. The first-order accurate BDF (BDF1) is given by

$$\alpha = 1, \quad \alpha_n = -1, \quad \alpha_{n-1} = 0, \quad \alpha_{n-2} = 0, \tag{51}$$

the second-order BDF (BDF2) is given by

$$\alpha = 3/2, \quad \alpha_n = -2, \quad \alpha_{n-1} = 1/2, \quad \alpha_{n-2} = 0, \tag{52}$$

and the third-order BDF (BDF3) is given by

$$\alpha = 11/6, \quad \alpha_n = -3, \quad \alpha_{n-1} = 3/2, \quad \alpha_{n-2} = -1/3. \tag{53}$$

In the pseudo steady state, i.e., $\overline{u}_j^{k+1} = \overline{u}_j^k$ and $\overline{p}_j^{k+1} = \overline{p}_j^k$, we obtain the solution at the next physical time level, $n+1$,

$$\overline{u}_j^{n+1} = \overline{u}_j^{k+1}, \tag{54}$$

$$\overline{p}_j^{n+1} = \overline{p}_j^{k+1}. \tag{55}$$

Note that the physical time derivative term, which has a form similar to a reactive term, can be treated as a source term to the first equation. Therefore, the computation of the face values can be performed exactly as described in the previous section if the physical time derivative is added to the source term.

$$s_1 \to s_1 - \partial_t u^{k+1}. \tag{56}$$

The BDF schemes are implicit time-stepping schemes, and therefore a pseudo steady problem, or simply a system of residual equations, needs to be solved at every physical time step. For this purpose, the explicit pseudo time stepping scheme is not very efficient. As mentioned earlier, we solve the pseudo steady state problem not by the pseudo time stepping scheme but by Newton's method as we will describe later.

6 Extension to Advection Diffusion

6.1 Advection-Diffusion Equation

Consider the advection-diffusion equation:

$$\partial_\tau u + a\,\partial_x u = \nu\,\partial_{xx} u + s_1, \tag{57}$$

where a is a constant advection speed, ν is a constant diffusion coefficient, and s_1 is a source term. The source term includes forcing functions as well as a physical time derivative, e.g., the backward difference formula (BDF), for time accurate computations. A reactive term can also be included in the source term.

As we have shown in the previous section, the active flux scheme is available for diffusion. Also, it is already available for advection [1, 3]. In some methods, typically those that allow semi-discrete formulations, a third-order advection-diffusion scheme can be easily constructed by adding the advection and diffusion schemes. Examples include the finite-volume methods and the discontinuous Galerkin methods. However, in other methods, such a simple construction is known to destroy the formal accuracy of the individual schemes, for example, in the residual-distribution method [35] and in the third-order edge-based finite-volume method [13]. The active flux scheme is a mixture of both as it consists of two steps: the characteristic flux computation step and the finite-volume discretization step. The simple construction is applicable to the finite-volume discretization step, but not to the flux computation step. The latter requires a careful construction to preserve accuracy as discussed in details for a hyperbolic system with source terms in the previous section. Here, we propose a strategy for constructing active flux schemes for the advection-diffusion equation based on two alternative forms of the advection-diffusion equation.

6.2 Conservative Form

We write the advection-diffusion equation as a first-order hyperbolic system:

$$\partial_\tau \mathbf{u} + \partial_x \mathbf{f} = \mathbf{s}, \tag{58}$$

where

$$\mathbf{u} = \begin{bmatrix} u \\ p \end{bmatrix}, \quad \mathbf{f} = \begin{bmatrix} au - \nu p \\ -u/T_r \end{bmatrix}, \quad \mathbf{s} = \begin{bmatrix} s_1 \\ s_2 \end{bmatrix}, \tag{59}$$

where $s_2 = -p/T_r$. We will employ this conservative form in the finite-volume discretization step to ensure conservation.

6.3 Source Form

The flux computation step relies on the solution of the characteristic equations, and therefore requires the complete eigen-structure for the hyperbolic advection-diffusion system (58). The complete eigen-structure is available for the advection-diffusion as described in Ref.[8], but currently not available for the compressible Navier-Stokes equations. To avoid the difficulty, we propose a method based on the following form of the advection-diffusion equation:

$$\partial_\tau u = \nu\, \partial_x p + s_1 - ap, \tag{60}$$

$$\partial_\tau p = \frac{1}{T_r}\left(\partial_x u - p\right), \tag{61}$$

where the advective term has been expressed by p, which will be equivalent to $\partial_x u$ in the pseudo steady state, and taken to the right hand side. A major advantage of this particular form is that it is exactly in the form of the hyperbolic diffusion system with source terms (2), for which the active flux scheme is already available. The eigenvalues and eigenvectors of the system are exactly the same as those of the hyperbolic diffusion system. Therefore, the flux computation step is exactly the same as described in Section 4, except that the source term includes the advective term as $-ap$. Note that since it requires only the eigen-structure of the diffusive term, it can be extended to the compressible Navier-Stokes equations for which a complete eigen-structure is known for hyperbolized viscous terms [9, 15, 16]. It is also possible to express the inviscid terms as source terms by using extra variables introduced to form a hyperbolic Navier-Stokes system [16].

6.4 Active Flux Scheme for Advection-Diffusion Equation

The discretization of the advection-diffusion system (58) is performed in exactly the same way for the diffusion system (2) as described in Section 4. The conservative finite volume discretization (22) with the following advection-diffusion flux:

$$\mathbf{f}_{j+1/2} = \begin{bmatrix} a\dfrac{u^k_{j+1/2} + u^{k+1}_{j+1/2}}{2} - \nu\dfrac{p^k_{j+1/2} + p^{k+1}_{j+1/2}}{2} \\ -\dfrac{u^k_{j+1/2} + u^{k+1}_{j+1/2}}{2T_r} \end{bmatrix}, \tag{62}$$

and similarly for $\mathbf{f}_{j-1/2}$. The face values, $u^{k+1}_{j+1/2}$ and $p^{k+1}_{j+1/2}$, are computed by Equations (38) and (39) with

$$s_1^w = L_r s_2 + (s_1 - ap), \quad s_2^w = L_r s_2 - (s_1 - ap). \tag{63}$$

Unsteady advection-diffusion schemes can be easily constructed as described earlier simply by incorporating a physical time derivative term into the source term, s_1.

7 Implicit Solver

For both steady and unsteady problems, it is necessary to solve the residual equation:

$$\mathbf{Res}(\mathbf{U}_h) = 0, \tag{64}$$

where $\mathbf{U}_h$ is a global numerical solution vector containing all cell-averages and face-values. The global residual vector $\mathbf{Res}(\mathbf{U}_h)$ consists of the steady residuals defined by Equation (24) together with the face-residuals $\mathbf{Res}_{j+1/2}$ defined by

$$\mathbf{Res}_{j+1/2} = \begin{bmatrix} u^{k+1}_{j+1/2} - u^{k}_{j+1/2} \\ \\ p^{k+1}_{j+1/2} - p^{k}_{j+1/2} \end{bmatrix}, \tag{65}$$

for the face $j + 1/2$. As in the previous study, the residual equation can be solved by the pseudo-time iteration (23) with the explicit face-value evaluation as in Equations (38) and (39). Thanks to the hyperbolic formulation for diffusion, the number of pseudo-time iterations exhibits linear increase with respect to the grid size, instead of the quadratic increase that is typical for diffusion [7]. However, it still requires hundreds of residual evaluations for convergence and it is highly desirable to improve the solver convergence. To this end, we construct Newton's method:

$$\mathbf{U}^{k+1}_h = \mathbf{U}^{k}_h + \Delta\mathbf{U}_h, \tag{66}$$

where the correction $\Delta\mathbf{U}_h$ is obtained by solving the linearized system,

$$\frac{\partial \mathbf{Res}}{\partial \mathbf{U}_h}\Delta\mathbf{U}_h = -\mathbf{Res}(\mathbf{U}^{k}_h). \tag{67}$$

The residual Jacobian on the left hand side is derived analytically by exactly differentiating the residual and then implemented in a code. For more complex equations, numerical evaluations by the automatic differentiation technique should be considered [38]. The linear system (67) is relaxed by the Gauss-Seidel relaxation with the under-relaxation parameter of 0.2 to reduce the linear residual by two orders of magnitude in the L_∞ norm. It is at this stage that we reap the benefit of the hyperbolic formulation being able to take much larger "time steps". Typically, we perform the Newton iteration (66) to reduce the residual by six orders of magnitude both for steady problems and for sub-iterations within unsteady problems.

The implicit active-flux advection-diffusion scheme may be compared with the third-order residual-distribution scheme developed in Ref.[37]. Both schemes are based on the hyperbolic advection-diffusion system and are third-order accurate for both the solution and the gradient on irregular grids. The implicit solver is not exactly

Newton's method for the third-order residual-distribution scheme, but the residual Jacobian is block tri-diagonal and the convergence is as rapid as Newton's method in practice. On the other hand, the solver in this paper is exactly Newton's method, but the Jacobian is not compact having more than two off-diagonal blocks.

Note that we may construct the linear system with the pseudo-time derivative incorporated:

$$\left(\mathbf{D} + \frac{\partial \mathbf{Res}}{\partial \mathbf{U}_h}\right) \Delta \mathbf{U}_h = -\mathbf{Res}(\mathbf{U}_h^k), \tag{68}$$

where $\mathbf{D}$ is a diagonal matrix containing $\Delta x_j / \Delta \tau$ for each cell-average and zero for face values. This formulation corresponds to the BDF1 scheme applied to the pseudo-time integration. However, as in Equation (67), we have ignored $\mathbf{D}$, and thus no pseudo-time stepping is performed. Therefore, the resulting unsteady scheme is not a dual time-stepping scheme. It is an implicit time-stepping scheme with the unsteady residual equation solved by Newton's method.

8 Results

8.1 Steady Advection-Diffusion Problem

We consider a steady advection-diffusion problem taken from Refs.[8, 10, 37]:

$$a\,\partial_x u = \nu\,\partial_{xx} u + s_1, \tag{69}$$

with the boundary conditions $u(0) = 0$ and $u(1) = 1$, and the forcing term,

$$s_1 = \frac{\nu\pi}{a}[a\cos(\pi x) + \nu\pi \sin(\pi x)]. \tag{70}$$

The exact solution is given by

$$u(x) = \frac{exp(-a/\nu) - exp(ax/\nu - a/\nu)}{exp(-a/\nu) - 1} + \frac{\nu}{a}\sin(\pi x). \tag{71}$$

The parameter a/ν is given the values $0.01, 0.1, 1.0, 10, 100$. The Dirichlet boundary condition is imposed weakly at both ends as described in Section 3.3. Therefore, the solution values are determined by the numerical scheme for all cells and faces, including the boundary faces. Steady convergence is taken to be achieved when the L_1 norm of the residual is reduced by six orders of magnitude. The initial solution is set by a randomly perturbed exact solution: the exact solution + perturbation of $O(0.1)$. It is noted that very similar results were obtained without the perturbation. Computations have been performed for a series of grids: 16, 32, 64, and 128 cells. The grids are uniformly spaced for $a/\nu = 0.01, 0.1, 1.0$, and slightly stretched for $a/\nu = 10, 100$.

Results are shown in Figures 3-12. As shown in Figures 3, 5, 7, 9, and 11, the numerical solutions are very

accurate through the boundaries even on the coarsest grid of 16 cells. Figures 4, 6, 8, 10, 12 show that third-order accuracy has been verified for both the solution and the gradient for all cases. In the case of $a/\nu = 100$, Figure 12 shows that fourth-order accuracy is achieved for the solution and the gradient at faces and the gradient in the cells. Although one order higher order of accuracy has been known to occur for a finite-volume hyperbolic advection-diffusion scheme when advection dominates [13], it is not immediately clear how it happens to the active-flux scheme. These results show that the implicit solver converged within five Newton iterations and the number of linear relaxations required to reduce the linear residuals by two orders of magnitude increases linearly with the number of nodes, not quadratically which is typical for diffusion problems. Note that these results demonstrate significant improvements over explicit pseudo-time stepping schemes as it only requires about five residual evaluations to obtain the solution, rather than hundreds of residual evaluations [39]. Although Newton's method involves hundreds of linear relaxations per iteration, the residual is fixed on the right hand side during the relaxation process as in Equation (67) and thus each relaxation step is significantly cheaper than a residual evaluation.

8.2 Unsteady Advection-Diffusion Problem

We consider the following unsteady advection-diffusion problem,

$$\partial_t u + a\,\partial_x u = \nu\,\partial_{xx} u, \tag{72}$$

with the following boundary conditions:

$$u(0) = 0, \tag{73}$$

$$u(1) = U\cos(\omega t), \tag{74}$$

where U and ω are arbitrary constants. The exact solution exists to this problem; it can be found in Ref.[40]. For numerical results, the parameters have been set as follows:

$$a = 4.246, \quad \nu = 2.123, \quad U = 2.0, \quad \omega = 2\pi. \tag{75}$$

Again, the boundary condition is imposed weakly and thus the solutions are computed by the numerical scheme at all cells and faces. At every physical time step, the initial solution is set as the solution at the previous physical time step, and the Newton sub-iteration is taken to be converged when the L_1 norm of the unsteady residual is reduced by six orders of magnitude. To start up the computation, we use BDF1 over the first step, BDF2 in the next step, and BDF3 thereafter. Ideally, it would be best to perform the first two steps with a small enough time step not to introduce large errors, but numerical results show that the low-order errors in

the first two steps do not greatly impact the accuracy of the solution at a later time.

We tested the scheme for a given grid of 20 cells with randomly distributed nodes by refining the time step: $0.4/2^m$, where $m = 0, 1, 2, 3$. The final time is 6.0, i.e., six periods in the unsteady boundary condition. The last time step is adjusted in order to complete the computation precisely at the final time. For this purpose, the BDF3 scheme with variable time steps is employed:

$$\partial_t \overline{u}^{k+1} = \tilde{\alpha}\overline{u}_j^{k+1} + \tilde{\alpha}_n \overline{u}_j^n + \tilde{\alpha}_{n-1}\overline{u}_j^{n-1} + \tilde{\alpha}_{n-2}\overline{u}_j^{n-2}, \tag{76}$$

where for $n > 2$

$$\tilde{\alpha} = -(\tilde{\alpha}_n + \tilde{\alpha}_{n-1} + \tilde{\alpha}_{n-2}), \quad \tilde{\alpha}_n = -\frac{(\Delta t_n + \Delta t_{n-1})(\Delta t_n + \Delta t_{n-1} + \Delta t_{n-2})}{\Delta t_n \Delta t_{n-1}(\Delta t_{n-1} + \Delta t_{n-2})}, \tag{77}$$

$$\tilde{\alpha}_{n-1} = \frac{\Delta t_n(\Delta t_n + \Delta t_{n-1} + \Delta t_{n-2})}{\Delta t_{n-1}\Delta t_{n-2}(\Delta t_n + \Delta t_{n-1})}, \quad \tilde{\alpha}_{n-2} = -\frac{\Delta t_n(\Delta t_n + \Delta t_{n-1})}{\Delta t_{n-2}(\Delta t_{n-1} + \Delta t_{n-2})(\Delta t_n + \Delta t_{n-1} + \Delta t_{n-2})}, \tag{78}$$

and the time steps are defined as $\Delta t_n = t^{n+1} - t^n$, $\Delta t_{n-1} = t^n - t^{n-1}$, and $\Delta t_{n-2} = t^{n-1} - t^{n-2}$. Results are shown in Figure 13, which confirms the formal third-order time accuracy for all variables.

Also shown is the iterative convergence for the Newton sub-iteration and for the linear relaxation. As can be seen, only four iterations, thus four residual evaluations, are required at every physical time step. The number of linear relaxations required to reduce the linear residual by two orders of magnitude decreases as the time step is reduced. It may be due to the fact that the initial solution for the Newton iteration, which is taken to be the solution at the previous time, is closer to the solution at the next time step. The four residual evaluations per physical time step is a tremendous improvement over the explicit pseudo-time stepping scheme employed in Ref.[39], which requires hundreds of residual evaluations per physical time step.

The cell-averaged solution and gradient at $t = 5.0, 5.25, 5.5, 5.75$ for the case of $\Delta t = 0.4$ are plotted in Figures 14, 15, 16, 17, respectively. These results show that the active-flux advection-diffusion scheme enables highly accurate unsteady simulations on a rather coarse grid even for irregular grids. For the sake of completeness, we also performed a grid refinement study to verify the spatial accuracy for the unsteady problem. We fixed the time step as $\Delta t = 0.0001$, and performed the computation for irregular grids of 8, 16, 32, and 64 cells. The errors were measured at the final time $t = 0.1$. Results are shown in Figure 18. Clearly, third-order accuracy has been confirmed for all variables. The average number of Newton sub-iterations is four for all grids (i.e., independent of the grid size), and the averaged number of linear relaxations increases linearly with the grid size as expected. Finally, we note that the residual reduction of six orders of magnitude may be relaxed by one or two orders to obtain essentially the same results, but not too much as it was found that the design order cannot be obtained or the errors get larger with two orders of magnitude reduction for fine grids.

9 Concluding Remarks

In this paper, we constructed active-flux schemes for diffusion and extended them to advection diffusion. The construction have been made in several stages. First, we have demonstrated that the active flux method can be constructed for hyperbolic systems with source terms. In so doing, we have shown that the characteristic equations need to be integrated along with the source terms to compute the face value. The developed scheme is then directly applicable to diffusion written in the form of a first-order hyperbolic system. This system is completely equivalent to the diffusion equation in the pseudo steady state. Unsteady schemes are then constructed by incorporating the backward-difference formulas into the source term.

Finally, we have extended the diffusion scheme to the advection-diffusion scheme by adding the advective term as a source term to the diffusion scheme. The solution gradient, which is computed simultaneously to third-order accuracy by the diffusion scheme, is used to express the advective term as a scalar source term. Third-order accuracy in the solution as well as in the gradient, and rapid convergence by Newton's method (four or five residual evaluations both for steady problems and for sub-iterations within unsteady problems) have been demonstrated for steady and unsteady problems on irregular grids.

Future work should focus on extensions to two and three dimensions as well as to the Navier-Stokes equations. The idea of treating the advective term as a source term can be extended to the compressible Navier-Stokes equations by a hyperbolic formulation for the Navier-Stokes equations proposed in Ref.[16].

Acknowledgments

This work has been funded by the NASA NRA/ROA-2011 Contract NNX12AJ70A.

References

[1] Timothy A. Eymann and Philip L. Roe. Active flux schemes for systems. In *Proc. of 20th AIAA Computational Fluid Dynamics Conference*, AIAA Paper 2011-3840, Hawaii, 2011.

[2] Timothy A. Eymann and Philip L. Roe. Multidimensional active flux schemes. In *Proc. of 21st AIAA Computational Fluid Dynamics Conference*, AIAA Paper 2013-2940, San Diego, California, June 2013.

[3] B. van Leer. Towards the ultimate conservative difference scheme. IV. a new approach to numerical convection. *J. Comput. Phys.*, 23:276–299, 1977.

[4] D. Fan and P. L. Roe. Investigations of a new scheme for wave propagation. In *Proc. of 22nd AIAA Computational Fluid Dynamics Conference*, AIAA Paper 2015-2449, Dallas, TX, 2015.

[5] B. van Leer and S. Nomura. Discontinuous Galerkin for diffusion. In *Proc. of 17th AIAA Computational Fluid Dynamics Conference*, AIAA Paper 2005-5108, Toronto, 2005.

[6] M. Sun and P. L. Roe. Viscous active flux method. Unpublished, June 2013.

[7] H. Nishikawa. A first-order system approach for diffusion equation. I: Second order residual distribution schemes. *J. Comput. Phys.*, 227:315–352, 2007.

[8] H. Nishikawa. A first-order system approach for diffusion equation. II: Unification of advection and diffusion. *J. Comput. Phys.*, 229:3989–4016, 2010.

[9] H. Nishikawa. New-generation hyperbolic Navier-Stokes schemes: $O(1/h)$ speed-up and accurate viscous/heat fluxes. In *Proc. of 20th AIAA Computational Fluid Dynamics Conference*, AIAA Paper 2011-3043, Honolulu, Hawaii, 2011.

[10] A. Mazaheri and H. Nishikawa. First-order hyperbolic system method for time-dependent advection-diffusion problems. *NASA-TM–2014–218175*, March 2014.

[11] A. Mazaheri and H. Nishikawa. Very efficient high-order hyperbolic schemes for time-dependent advection-diffusion problems: Third-, Fourth, and Sixth-Order. *Comput. Fluids*, 102:131–147, October 2014.

[12] H. Nishikawa. First-, second-, and third-order finite-volume schemes for diffusion. *J. Comput. Phys.*, 256:791–805, 2014.

[13] H. Nishikawa. First, second, and third order finite-volume schemes for advection-diffusion. *J. Comput. Phys.*, 273:287–309, 2014.

[14] H. Nishikawa. Accuracy-preserving boundary flux quadrature for finite-volume discretization on unstructured grids. *J. Comput. Phys.*, 281:518–555, 2015.

[15] H. Nishikawa. First, second, and third order finite-volume schemes for Navier-Stokes equations. In *Proc. of 7th AIAA Theoretical Fluid Mechanics Conference, AIAA Aviation and Aeronautics Forum and Exposition 2014*, AIAA Paper 2014-2091, Atlanta, GA, 2014.

[16] H. Nishikawa. Alternative formulations for first-, second-, and third-order hyperbolic Navier-Stokes schemes. In *Proc. of 22nd AIAA Computational Fluid Dynamics Conference*, AIAA Paper 2015-2451, Dallas, TX, 2015.

[17] A. Mazaheri and H. Nishikawa. High-order hyperbolic residual-distribution schemes on arbitrary triangular grids. In *Proc. of 22nd AIAA Computational Fluid Dynamics Conference*, AIAA Paper 2015-2445, Dallas, TX, 2015.

[18] A. Mazaheri and H. Nishikawa. Improved second-order hyperbolic residual-distribution scheme and its extension to third-order on arbitrary triangular grids. *J. Comput. Phys.*, 300:455–491, 2015.

[19] C. Cattaneo. A form of heat-conduction equations which eliminates the paradox of instantaneous propagation. *Comptes Rendus de l'Académie des Sciences*, 247:431–433, 1958.

[20] P. Vernotte. Les paradoxes de la theorie continue del'equation de la chaleur. *Comptes rendus de l'Académie des sciences*, 246:3154–3155, 1958.

[21] G. B. Nagy, O. E. Ortiz, and O. A. Reula. The behavior of hyperbolic heat equations' solutions near their parabolic limits. *J. Math. Phys.*, 35:4334–4356, 1994.

[22] P. L. Roe and M. Arora. Characteristic-based schemes for dispersive waves I. the method of characteristics for smooth solutions. *Numer. Meth. Part. D. E.*, 9:459–505, 1993.

[23] R. B. Lowrie and J. E. Morel. Methods for hyperbolic systems with stiff relaxation. *Int. J. Numer. Meth. Fluids*, 40:413–423, 2002.

[24] S. F. Liotta, V. Romano, and G. Russo. Central schemes for balance laws of relaxation type. *SIAM J. Numer. Anal.*, 38:1337–1356, 2000.

[25] S. Jin and C. D. Levermore. Numerical schemes for hyperbolic conservation laws with stiff relaxation terms. *J. Comput. Phys.*, 126:449–467, 1996.

[26] H. Gómez, I. Colominas, F. Navarrina, and M. Casteleiro. A discontinuous Galerkin method for a hyperbolic model for convection-diffusion problems in CFD. *Int. J. Numer. Meth. Eng.*, 71:1342–1364, 2007.

[27] F. Navarrina H. Gómez, I. Colominas and M. Casteleiro. A finite element formulation for a convection-diffusion equation based on cattaneo's law. *Computer Methods in Applied Mechanics and Engineering*, 196:1757–1766, 2007.

[28] F. Navarrina H. Gómez, I. Colominas and M. Casteleiro. A hyperbolic model for convection-diffusion transport problems in CFD: Numerical analysis and applications. *Rev. R. Acad. Cien. Serie A. Mat.*, 102(2):319–334, 2008.

[29] F. Navarrina H. Gómez, I. Colominas and M. Casteleiro. A hyperbolic model for convection-diffusion transport problems in cfd: Numerical analysis and applications. *Revista de la Real Academia de Ciencias Exactas, Fisicas y Naturales - Serie A: Matematicas*, 102(2):319–334, 2008.

[30] F. Navarrina H. Gómez, I. Colominas and M. Casteleiro. A mathematical model and a numerical model for hyperbolic mass transport in compressible flows. *Heat and Mass Transfer*, 45(2):219–226, 2008.

[31] F. Navarrina J. Paris H. Gómez, I. Colominas and M. Casteleiro. A hyperbolic theory for advection-diffusion problems: Mathematical foundations and numerical modeling. *Archives of Computational Methods in Engineering*, 17(2):191–211, 2010.

[32] G. I. Montecinos, Lucas O. Müller, and E. F. Toro. Hyperbolic reformulation of a 1D viscoelastic blood flow model and ader finite volume schemes. *J. Comput. Phys.*, 266:101–123, 2014.

[33] G. I. Montecinos and E. F. Toro. Reformulations for general advection-diffusion-reaction equations and locally implicit ader schemes. *J. Comput. Phys.*, 275:415–442, 2014.

[34] E. F. Toro and G. I. Montecinos. Advection-diffusion-reaction equations: Hyperbolisation and high-order ader discretizations. *SIAM J. Sci. Comput.*, 36:A2423–A2457, 2014.

[35] H. Nishikawa and P. L. Roe. On high-order fluctuation-splitting schemes for Navier-Stokes equations. In C. Groth and D. W. Zingg, editors, *Computational Fluid Dynamics 2004*, pages 799–804. Springer-Verlag, 2004.

[36] Freddie Witherden Antony Farrington Peter Vincent Spencer Sherwin Joaquim Peiro Gianmarco Mengaldo, Daniele De Grazia. A guide to the implementation of boundary conditions in compact high-order methods for compressible aerodynamics. In *Proc. of 7th AIAA Theoretical Fluid Mechanics Conference, AIAA Aviation and Aeronautics Forum and Exposition 2014*, AIAA Paper 2014-2923, Atlanta, GA, 2014.

[37] A. Mazaheri and H. Nishikawa. Very efficient high-order hyperbolic schemes for time-dependent advection-diffusion problems: Third-, fourth-, and sixth-order. *Computers and Fluids*, 102:131–147, 2014.

[38] Louis B. Rall and George F. Corliss. An introduction to automatic differentiation. In Martin Berz, Christian H. Bischof, George F. Corliss, and Andreas Griewank, editors, *Computational Differentiation: Techniques, Applications, and Tools*, pages 1–17. SIAM, Philadelphia, Penn., 1996.

[39] H. Nishikawa, P. L. Roe, and Timothy A. Eymann. Active flux schemes for diffusion. In *Proc. of 7th AIAA Theoretical Fluid Mechanics Conference, AIAA Aviation and Aeronautics Forum and Exposition 2014*, AIAA Paper 2014-2092, Atlanta, GA, 2014.

[40] Katate Masatsuka. I do like CFD, VOL.1, Second Edition. `http://www.cfdbooks.com`, 2013.

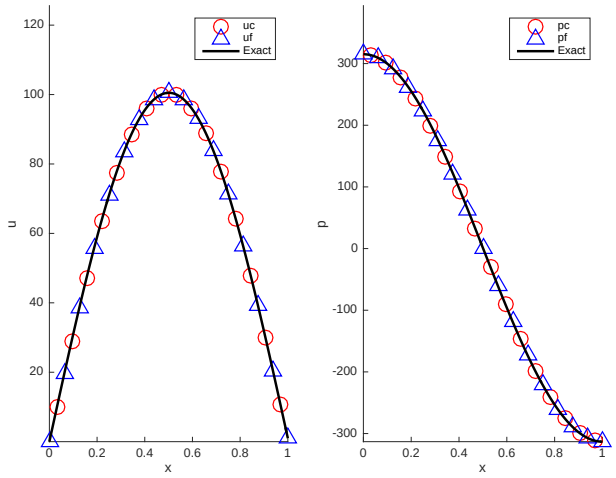

Figure 3: $a/\nu = 0.01$: Steady solution on the 16-cell grid. (uc, pc) are the cell-averaged values, and (uf, pf) are the point values at the face.

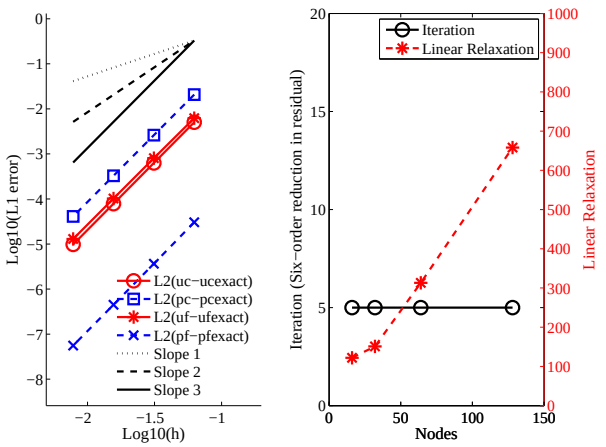

Figure 4: $a/\nu = 0.01$: L_1 Error and iterative convergence results for a steady problem. On the right, Iteration refers to the number of Newton iterations, and Linear Relaxation refers to the number of Gauss-Seidel linear relaxations applied to the linearized system (67).

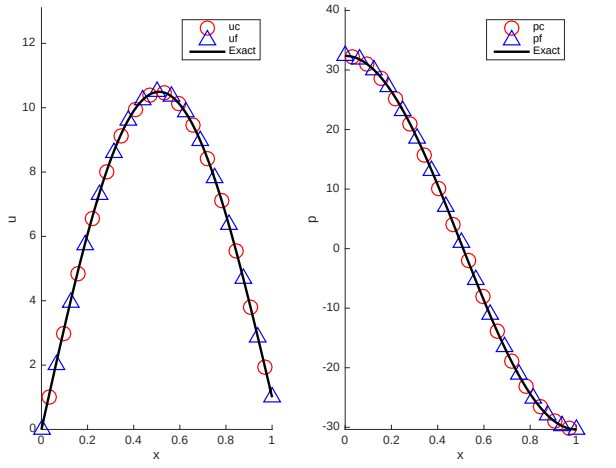

Figure 5: $a/\nu = 0.1$: Steady solution on the 16-cell grid. (uc, pc) are the cell-averaged values, and (uf, pf) are the point values at the face.

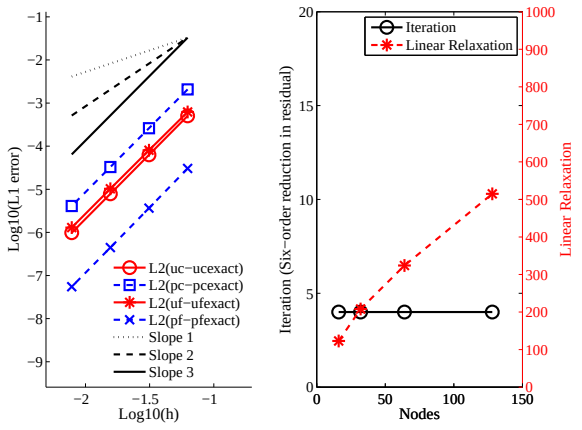

Figure 6: $a/\nu = 0.1$: L_1 Error and iterative convergence results for a steady problem. On the right, Iteration refers to the number of Newton iterations, and Linear Relaxation refers to the number of Gauss-Seidel linear relaxations applied to the linearized system (67).

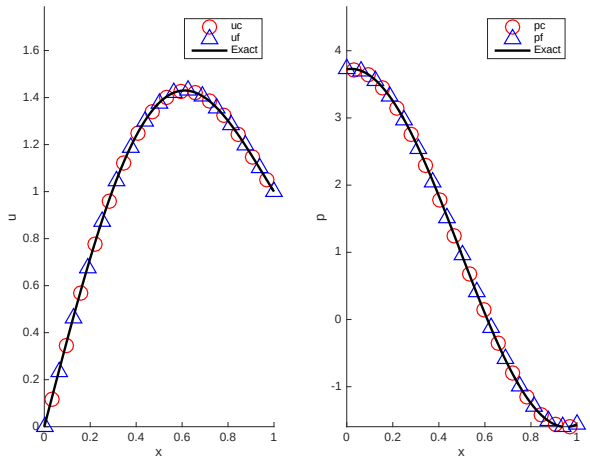

Figure 7: $a/\nu = 1.0$: Steady solution on the 16-cell grid. (uc, pc) are the cell-averaged values, and (uf, pf) are the point values at the face.

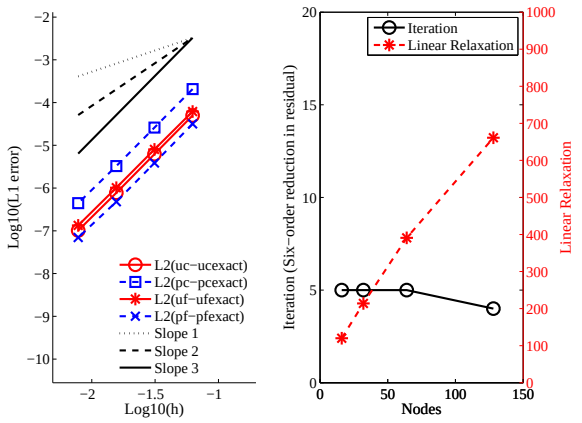

Figure 8: $a/\nu = 1.0$: L_1 Error and iterative convergence results for a steady problem. On the right, Iteration refers to the number of Newton iterations, and Linear Relaxation refers to the number of Gauss-Seidel linear relaxations applied to the linearized system (67).

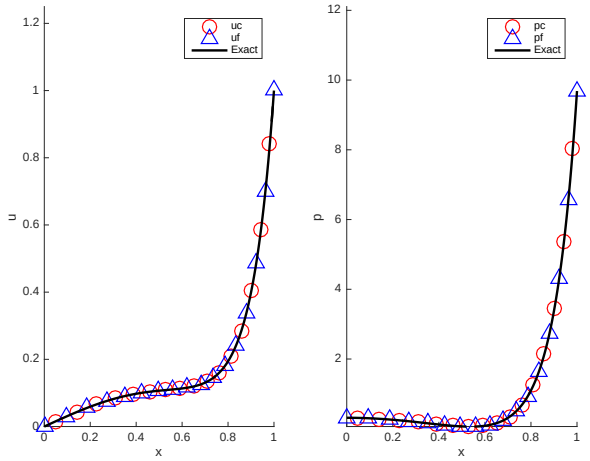

Figure 9: $a/\nu = 10$: Steady solution on the 16-cell grid. (uc, pc) are the cell-averaged values, and (uf, pf) are the point values at the face.

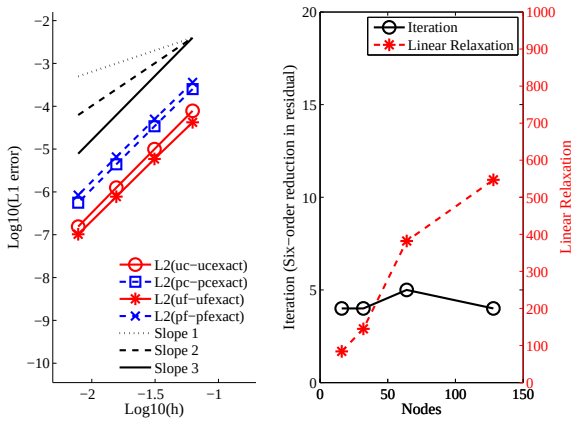

Figure 10: $a/\nu = 10$: L_1 Error and iterative convergence results for a steady problem. On the right, Iteration refers to the number of Newton iterations, and Linear Relaxation refers to the number of Gauss-Seidel linear relaxations applied to the linearized system (67).

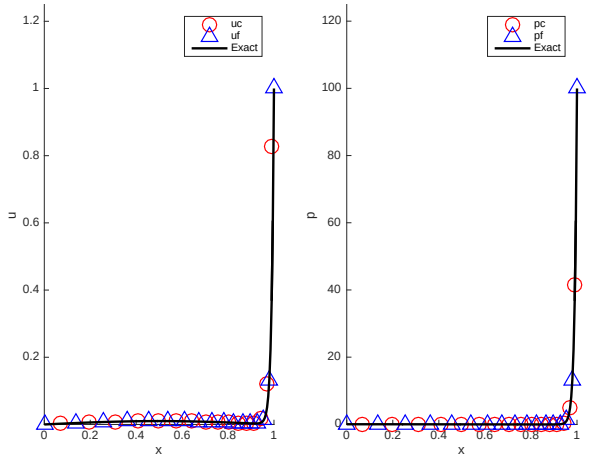

Figure 11: $a/\nu = 100$: Steady solution on the 16-cell grid. (uc, pc) are the cell-averaged values, and (uf, pf) are the point values at the face.

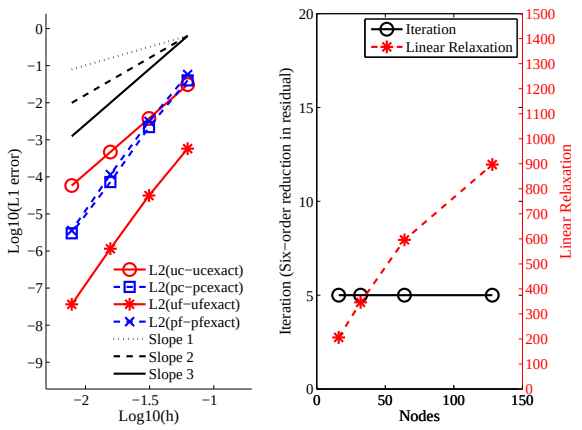

Figure 12: $a/\nu = 100$: L_1 Error and iterative convergence results for a steady problem. On the right, Iteration refers to the number of Newton iterations, and Linear Relaxation refers to the number of Gauss-Seidel linear relaxations applied to the linearized system (67).

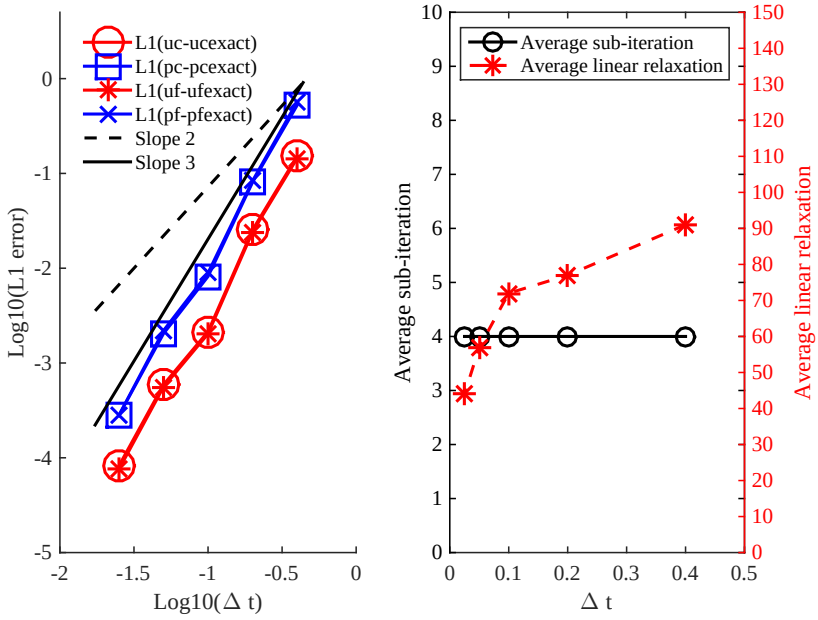

Figure 13: Time accuracy verification for an unsteady problem with $\Delta t = 0.4/2^m$, where $m = 0, 1, 2, 3, 4$, the final time $t = 6.0$, and 20 cells. Time accuracy on the left and the numbers of sub-iterations and linear relaxations, averaged over the entire computation, on the right.

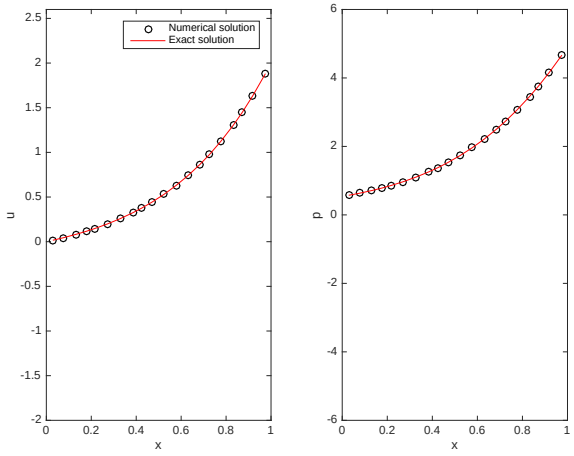

Figure 14: Unsteady solution at $t = 5.0$. $\Delta t = 0.4$ and 20 cells.

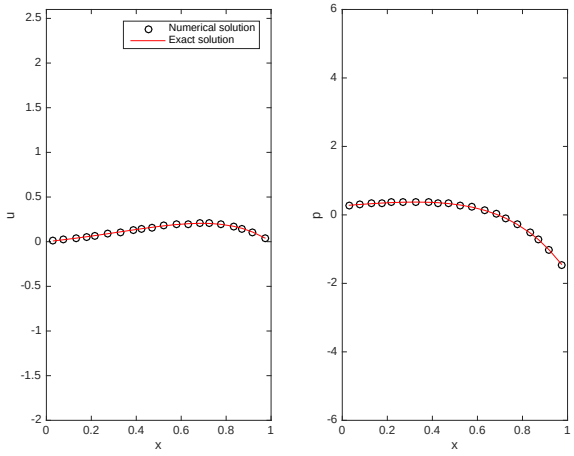

Figure 15: Unsteady solution at $t = 5.25$. $\Delta t = 0.4$ and 20 cells.

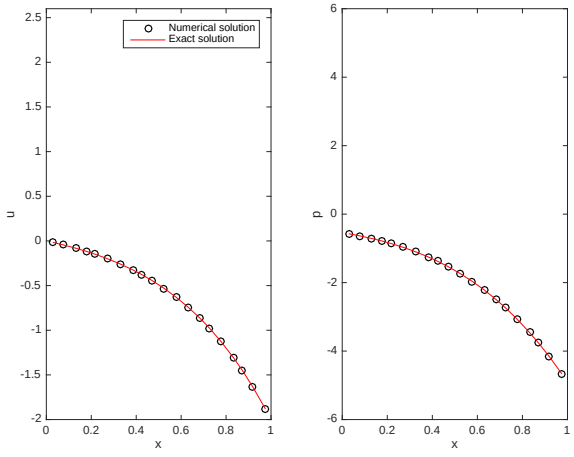

Figure 16: Unsteady solution at $t = 5.5$. $\Delta t = 0.4$ and 20 cells.

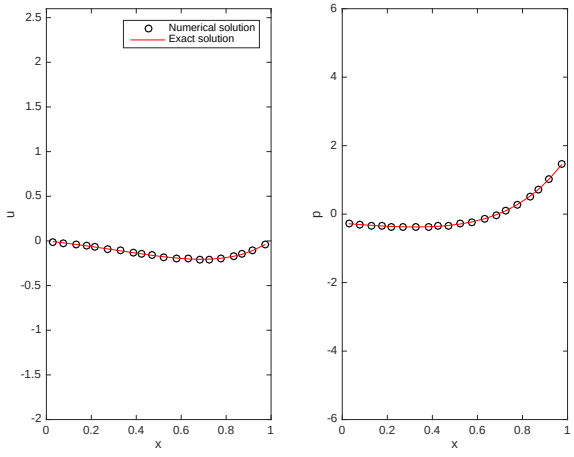

Figure 17: Unsteady solution at $t = 5.75$. $\Delta t = 0.4$ and 20 cells.

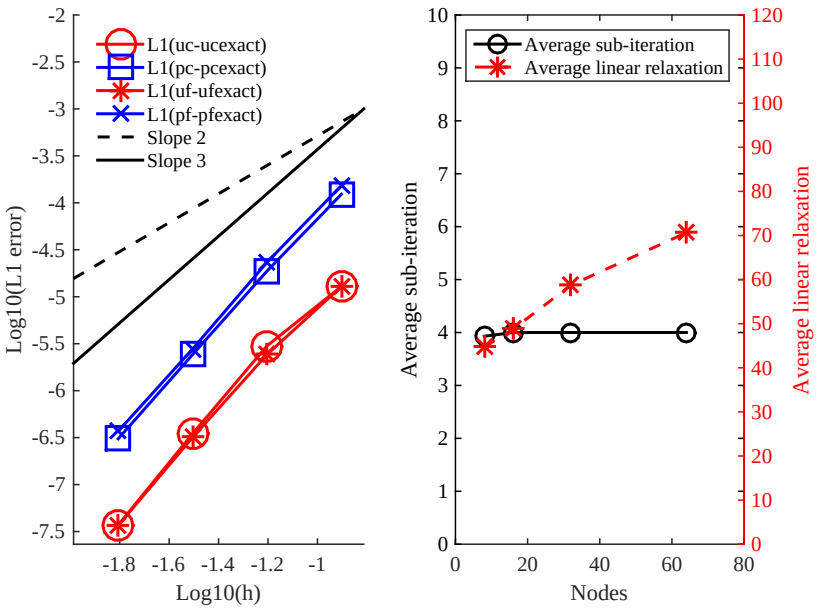

Figure 18: Spatial accuracy verification for an unsteady problem with $\Delta t = 0.0001$, the final time $t = 0.1$, and grids of 8, 16, 32, 64 cells. Spatial accuracy on the left and the numbers of sub-iterations and linear relaxations, averaged over the entire computation, on the right.