18th AIAA Computational Fluid Dynamics Conference
25 - 28 June 2007, Miami, FL

AIAA 2007-4468

An Analysis of the Upwind Moment Scheme and Its Extension to Systems of Nonlinear Hyperbolic-Relaxation Equations

Yoshifumi Suzuki[*] and Bram van Leer[†]

The University of Michigan, Ann Arbor, MI 48109–2140 USA

The goal of this research is developing a unified numerical method for simulating continuum and transitional flow. To achieve our ultimate goal, first, hyperbolic-relaxation equations are introduced, then a new discretization method is developed. The method is based on Huynh's upwind moment scheme, with implicit treatment of the source term. Our previous linear method is generalized to 1-D nonlinear hyperbolic-relaxation equations. First, a Fourier analysis is conducted to uncover the accuracy and stability. Then, the Euler equations with heat transfer, which reduce to the isothermal Euler equations in the equilibrium limit, are adopted as a model equation for the numerical experiment. The analysis and numerical results show the superiority of the proposed method in both accuracy and efficiency over the semi-discrete, method-of-line approach.

I. Introduction

A unified methodology is necessary to achieve high accuracy and efficiency when both continuum and rarefied regimes are in the same computational domain. The methods currently available for the transition from continuum to moderately rarefied flow switch between a continuum and a particle approach; this strategy fails to achieve both accuracy and efficiency at the same time. Thus, our goal is developing a unified numerical method for simulation of a flow in the continuum and transition regimes. Besides our aim, especially in recent years, the need for robust high-order (more than second-order accurate) discretizations for high-fidelity CFD on unstructured grids has been widely recognized. To be attractive, a high-order method must not only be accurate, but efficient as well, thus reducing the total CPU time needed to yield a given accuracy. In this paper we propose a combination of two approaches for robust, accurate, and efficient schemes for advection-dominated flows on unstructured grids, one at the partial differential equation (PDE) level and the other at the discretization level.

The first approach is replacing everybody's favorite Navier-Stokes (NS) equations by a larger set of first-order hyperbolic-relaxation PDEs, which contains the NS equations. (N.B.: here "first-order" refers to the order of the PDEs.) In comparison with second-order PDE models such as the NS equations, first-order PDE models offer many numerical advantages: robustness, accuracy, and efficiency. More specifically, the latter require smaller discrete stencils, reducing communications in parallel processing; the first-order PDE can replace the global stiffness from the diffusion term with the local stiffness from the source term, and yield

[*] Research Assistant, Department of Aerospace Engineering, 2001 François-Xavier Bagnoud Building, Student Member.
[†] Professor, Department of Aerospace Engineering, 3000 François-Xavier Bagnoud Building, Fellow.

the best accuracy on non-smooth, adaptively refined grids. We will therefore consider the one-dimensional hyperbolic-relaxation form:

$$\partial_t \mathbf{u} + \partial_x \mathbf{f}(\mathbf{u}) = \frac{1}{\epsilon}\mathbf{s}(\mathbf{u}), \tag{1}$$

where $\mathbf{u} \in \mathbb{R}^m$ is the vector of conserved variables, $\mathbf{f}(\mathbf{u})\colon \mathbb{R}^m \to \mathbb{R}^m$ is the flux of $\mathbf{u}$, $\mathbf{s}(\mathbf{u})\colon \mathbb{R}^m \to \mathbb{R}^m$ is the source term, and $\epsilon \in \mathbb{R}^+$ is the relaxation time. This is the form of moment closures of the Boltzmann equation, where the source term describes departure from local thermodynamic equilibrium. Hence, the above hyperbolic-relaxation equation is able to describe a continuum and transitional flow, whereas the NS equations are only valid in a flow near local thermodynamic equilibrium. In the near equilibrium ($\epsilon \ll 1$), that is, when the system is stiff, the system formally reduces to a system of second-order equations,

$$\partial_t \mathbf{U} + \partial_x \mathbf{F}(\mathbf{U}) = \partial_x(\mathbf{D}\partial_x \mathbf{U}), \tag{2}$$

where $\mathbf{U} \in \mathbb{R}^n, n < m$, is the vector of conserved variables of the reduced equation, $\mathbf{F}(\mathbf{U})\colon \mathbb{R}^n \to \mathbb{R}^n$ is the flux of $\mathbf{U}$, and $\mathbf{D} \in \mathbb{R}^{n\times n}$ is a tensor of diffusion coefficients, with eigenvalues proportional to ϵ.

In conjunction with the PDE approach, a discretization method that can preserve compactness in high-order discretization is adopted. In standard finite-volume methods, high-order accuracy relies on piecewise-polynomial reconstruction, which requires extended stencils.[1] For instance, stencils for the quadratic reconstruction (third-order accuracy) on tetrahedral grids include 50 to 70 cells.[2] Discontinuous Galerkin (DG) methods overcome the issue of reconstruction by using extra equations for updating the polynomial representation of state variables. For a comprehensive literature review, see Cockburn and Shu.[3] Currently, the most successful DG methods are semi-discretizations combined with a TVD Runge–Kutta (RK) ordinary differential equation (ODE) solver[3,4] denoted as DG(k)–RKs, where k is the degree of the polynomial basis and s is the order of the RK method.

The method proposed here to solve the hyperbolic-relaxation equations is based on the 'upwind moment scheme[5]' recently developed by Huynh for hyperbolic conservation laws. The solution representation is only piecewise linear. The two key characteristics of this method are:

1. cell variables are updated over a half time step without any interactions with neighboring cells (Hancock's observation[6,7]);
2. the gradient of each flow variable evolves by an independent equation (DG representation).

We denote the upwind moment scheme by "DG(1)–Hancock method"; it looks promising in comparison to the popular method-of-line (MOL) approach such as FV/DG–MOL. The MOL decouples the discretization in space and time: first a semi-discrete form in space is solved, then integrated over time by a suitable ODE solver, e.g., RKs. The upwind moment scheme $\big($DG(1)–Hancock$\big)$ is a fully discrete, one step method with one intermediate update step needed for computing the volume integral of the flux. It requires solving a Riemann problem twice at each cell interface. Based on a Fourier analysis, the method achieves third-order accuracy in space and time. By design, the upwind moment scheme for a linear advection equation reduces to Van Leer's 'scheme III',[8] which is a DG spatial discretization with an exact shift operator for the time evolution. It was shown that the method is linearly stable up to Courant number 1 with an upwind flux, whereas DG spatial discretizations combined with MOL typically have a more strict stability condition: for DG(1)–RK2 (second-order) the limit is $\frac{1}{3}$, and for the DG(2)–RK3 (third-order) it is $\frac{1}{5}$.[3] Here we extend the upwind moment method to a one-dimensional *nonlinear* hyperbolic-relaxation system.

II. Extension of the Upwind Moment Scheme to Nonlinear Hyperbolic-Relaxation Equations (DG(1)–Hancock method)

When discretizing hyperbolic-relaxation equations (1), the source term has to be treated implicitly to ensure the stability in the stiff regime ($\epsilon \ll 1$). In contrast, the advection term is treated explicitly due to the complexity of the flux evaluation. These methods are 'semi-implicit.' It is expected that the stability of a method is solely constrained by the explicit discretization, that is, by an advective CFL condition. This can indeed be realized; however, the simple implicit treatment of source terms, e.g., the backward Euler method, does not always guarantee high-order accuracy in the stiff regime, where the source-term effect is important.

The unique feature of the upwind moment scheme is the evaluation of the space-time volume integral in the update equation of the solution gradient. Indeed, this volume integral of the flux makes the method third-order accurate. When applying the upwind moment scheme to the hyperbolic-relaxation equations, a difficulty arises in computing the volume integral of the source term in the update equation of the cell-average. Since the vector of cell-average variables at the next time level, $\bar{\mathbf{u}}^{n+1}$, is still unknown, no simple quadrature rule can be used. Thus, instead of computing the volume integral by a quadrature, we use a stiff ODE solver to solve the equation accurately. In our method, the time integral of the source term is discretized by the two-point Gauss–Radau quadrature, which can be regarded as an L-stable ODE solver if advection terms are omitted. The same quadrature points are used for the volume integral of the flux in the gradient update equation, whereas Huynh's original upwind moment method for conservation laws adopts the three-point Gauss–Lobatto quadrature. Because the source term does not contain spatial derivatives, the method is point-implicit, that is, the implicitness is local, requiring no information from neighboring cells. Previously a method for the case of a linear flux and source was presented;[9] here we extend the method to a nonlinear flux and source.

II.A. DG formulation

For brevity, we only consider a one-dimensional case with a uniform grid. Let $\Delta x = x_{j+1/2} - x_{j-1/2}$ be the cell width and $I_j = [x_{j-1/2}, x_{j+1/2}]$ be the domain of cell j. The general DG method is obtained by converting a differential equation to a weak formulation. Here, since a DG discretization is adopted only in space, a test function $v(x)\colon \mathbb{R} \to \mathbb{R}$ is just a function of space. The Hancock method is adopted for time discretization. Multiplying (1) by a test function, $v(x)$, and integrating over the interval I_j leads to the semi-discrete form of the weak formulation:

$$\frac{\partial}{\partial t}\int\limits_{I_j} \mathbf{u}(x,t)v(x)\,dx = -\int\limits_{I_j} \partial_x \mathbf{f}(\mathbf{u}(x,t))v(x)\,dx + \int\limits_{I_j} \frac{1}{\epsilon}\mathbf{s}(\mathbf{u}(x,t))v(x)\,dx. \tag{3}$$

Applying integration by parts on the flux term transfers the spatial differential operator acting on the flux $\mathbf{f}(\mathbf{u})$ to the test function $v(x)$,

$$\frac{\partial}{\partial t}\int\limits_{I_j} \mathbf{u}(x,t)v(x)\,dx = -\mathbf{f}(\mathbf{u}(x,t))v(x)\Big|_{x_{j-1/2}}^{x_{j+1/2}} + \int\limits_{I_j} \mathbf{f}(\mathbf{u}(x,t))\partial_x v(x)\,dx + \int\limits_{I_j} \frac{1}{\epsilon}\mathbf{s}(\mathbf{u}(x,t))v(x)\,dx. \tag{4}$$

To derive the fully discrete method, integrate again in time over $T^n = [t^n, t^{n+1}]$,

$$\begin{aligned}\underbrace{\int\limits_{I_j} \mathbf{u}(x,t)v(x)\,dx\Big|_{t^n}^{t^{n+1}}}_{\text{Time evolution}} &= \underbrace{-\int\limits_{T^n} \mathbf{f}(\mathbf{u}(x,t))v(x)\Big|_{x_{j-1/2}}^{x_{j+1/2}} dt}_{\text{Boundary integral}} \\ &\quad + \underbrace{\iint\limits_{I_j\times T^n} \mathbf{f}(\mathbf{u}(x,t))\partial_x v(x)\,dxdt}_{\text{Volume integral}} + \underbrace{\iint\limits_{I_j\times T^n} \frac{1}{\epsilon}\mathbf{s}(\mathbf{u}(x,t))v(x)\,dxdt}_{\text{Volume integral}}\,. \end{aligned}\tag{5}$$

Note that (5) is still exact in the weak formulation. To discretize the weak formulation, we now approximate the exact solution $\mathbf{u}(x,t)$ by $\mathbf{u}_h(x,t) \in \mathbb{P}^1$, and the test function $v(x)$ by $v_h(x) \in \mathbb{P}^1$, where the subscript h represents the approximate solution in polynomial space. The Legendre polynomials up to degree 1 are adopted for both basis and test functions, thus

$$\mathbf{u}_h(x,t) = \bar{\mathbf{u}}_j(t)\phi_0(x) + \overline{\Delta\mathbf{u}}_j(t)\phi_1(x), \tag{6a}$$

$$v_h(x) \in \operatorname{span}\{\phi_0(x), \phi_1(x)\}, \tag{6b}$$

where

$$\phi_0(x) = 1, \quad \phi_1(x) = \frac{x - x_j}{\Delta x}. \tag{7}$$

Here, the cell-average and the undivided gradient of $\mathbf{u}(x,t)$ in space are defined by

$$\bar{\mathbf{u}}_j(t) := \frac{1}{\Delta x}\int\limits_{I_j} \mathbf{u}(x,t)\,dx, \tag{8a}$$

$$\overline{\Delta\mathbf{u}}_j(t) := \frac{12}{\Delta x^2}\int\limits_{I_j} (x - x_j)\mathbf{u}(x,t)\,dx. \tag{8b}$$

Note that $\mathbf{u}(x,t) = \mathbf{u}_h(x,t) + O\big(\Delta x^2\big)$ in $x \in I_j$; however, the distributions of the true solution $\mathbf{u}(x,t)$ and the approximated polynomial function $\mathbf{u}_h(x,t)$ over the domain I_j are equivalent in the weak sense due to the orthogonality of the Legendre polynomials,

$$\int\limits_{I_j} \mathbf{u}(x,t)\phi_k(x)\,dx \equiv \int\limits_{I_j} \mathbf{u}_h(x,t)\phi_k(x)\,dx, \quad k = 0, 1. \tag{9}$$

Once the basis and test functions are chosen, approximated governing equations are derived by adopting the basis functions $\phi_0(x)$ and $\phi_1(x)$ as the test function $v_h(x)$. Inserting $\phi_i(x)$ into $v_h(x)$ leads to two independent update equations:

$$\int\limits_{I_j} \mathbf{u}_h(x,t)\,dx\Big|_{t^n}^{t^{n+1}} = -\int\limits_{T^n} \mathbf{f}(\mathbf{u}_h(x,t))\Big|_{x_{j-1/2}}^{x_{j+1/2}} dt + \frac{1}{\epsilon}\iint\limits_{I_j\times T^n} \mathbf{s}(\mathbf{u}_h(x,t))\,dxdt, \tag{10a}$$

$$\begin{aligned}\int\limits_{I_j} \mathbf{u}_h(x,t)\frac{x-x_j}{\Delta x}\,dx\Big|_{t^n}^{t^{n+1}} &= -\int\limits_{T^n} \mathbf{f}(\mathbf{u}_h(x,t))\frac{x-x_j}{\Delta x}\Big|_{x_{j-1/2}}^{x_{j+1/2}} dt \\ &\quad + \frac{1}{\Delta x}\iint\limits_{I_j\times T^n} \mathbf{f}(\mathbf{u}_h(x,t))\,dxdt + \frac{1}{\epsilon}\iint\limits_{I_j\times T^n} \mathbf{s}(\mathbf{u}_h(x,t))\frac{x-x_j}{\Delta x}\,dxdt.\end{aligned} \tag{10b}$$

The time-evolution term and boundary integral can be further simplified by inserting (6a) into $\mathbf{u}_h(x,t)$, then

$$\Delta x\left[\bar{\mathbf{u}}_j^{n+1} - \bar{\mathbf{u}}_j^{n}\right] = \underbrace{-\int\limits_{T^n}\left[\mathbf{f}_{j+1/2}(t) - \mathbf{f}_{j-1/2}(t)\right]dt}_{\text{Boundary integral}} + \underbrace{\frac{1}{\epsilon}\iint\limits_{I_j\times T^n} \mathbf{s}(\mathbf{u}_h(x,t))\,dxdt}_{\text{Volume integral}}, \tag{11a}$$

$$\begin{aligned}\frac{\Delta x}{12}\left[\overline{\Delta \mathbf{u}}_j^{n+1} - \overline{\Delta \mathbf{u}}_j^{n}\right] &= \underbrace{-\frac{1}{2}\int\limits_{T^n}\left[\mathbf{f}_{j+1/2}(t) + \mathbf{f}_{j-1/2}(t)\right]dt}_{\text{Boundary integral}} \\ &\quad \underbrace{+\frac{1}{\Delta x}\iint\limits_{I_j\times T^n} \mathbf{f}(\mathbf{u}_h(x,t))\,dxdt}_{\text{Volume integral}} \underbrace{+\frac{1}{\epsilon}\iint\limits_{I_j\times T^n} \mathbf{s}(\mathbf{u}_h(x,t))\frac{x-x_j}{\Delta x}\,dxdt}_{\text{Volume integral}},\end{aligned} \tag{11b}$$

where $\mathbf{f}_{j\pm1/2}(t) = \mathbf{f}(\mathbf{u}_h(x_{j\pm1/2},t))$. The approximation made so far is in the solution representation, which is polynomial of degree 1, and in using the same polynomial basis for basis and test functions. The next step is approximating the boundary integral, $\int_{T^n}(\cdot)\,dt$, for the interface flux, and the volume integral, $\iint_{I_j\times T^n}(\cdot)\,dxdt$, for both source term and flux by quadrature. Previously, the method for a linear flux and source term was introduced.[9] Also, a Fourier analysis of both semi-discrete HR2 and DG(1) methods was presented.[10] Here, we extend the linear method to nonlinear flux and source terms, and a Fourier analysis of a fully discrete method is presented.

II.B. Boundary Integral of the Flux (Interface-Flux Calculation)

At the cell-interfaces, $x_{j\pm1/2}$, the time integral of a flux is approximated by the midpoint rule. Thus when a flux is integrated over the time interval in $[t^n, t^{n+k}]$, the flux at $t^{n+k/2}$ is considered as the averaged flux:

$$\int\limits_{t^n}^{t^{n+k}} \mathbf{f}_{j\pm1/2}(t)\,dt \approx (k\Delta t)\;\mathbf{f}_{j\pm1/2}^{n+k/2}, \tag{12}$$

where $\Delta t = t^{n+1} - t^n$ and $k \in [0,1]$. Since the approximated solution at the cell-interface, $\mathbf{u}_h(x_{j\pm1/2},t)$, is discontinuous, a Riemann problem is solved exactly or approximately to compute the interface-flux. Let $\hat{\mathbf{f}}$

be the solution of a Riemann solver, then

$$\mathbf{f}^{n+k/2}_{j\pm1/2} \approx \hat{\mathbf{f}}_{j\pm1/2}\big(\mathbf{u}^{n+k/2}_{j\pm1/2,L}, \mathbf{u}^{n+k/2}_{j\pm1/2,R}\big). \tag{13}$$

For a linear flux, $\mathbf{f}(\mathbf{u}) = \mathbf{A}\mathbf{u}$ where $\mathbf{A} \in \mathbb{R}^{m\times m}$, the upwind flux is given by

$$\hat{\mathbf{f}}(\mathbf{u}_L, \mathbf{u}_R) = \mathbf{A}^+\mathbf{u}_L + \mathbf{A}^-\mathbf{u}_R, \tag{14}$$

with $\mathbf{A}^\pm = \mathbf{R}\mathbf{\Lambda}^\pm\mathbf{L}$, where $\mathbf{\Lambda} \in \mathbb{R}^{m\times m}$ is the diagonal matrix of eigenvalues of $\mathbf{A}$. For a nonlinear flux, common approximate Riemann solvers are given by the following form:

$$\hat{\mathbf{f}}(\mathbf{u}_L, \mathbf{u}_R) = \frac{1}{2}\left[\mathbf{f}(\mathbf{u}_R) + \mathbf{f}(\mathbf{u}_L)\right] - \frac{1}{2}\mathbf{Q}\left[\mathbf{u}_R - \mathbf{u}_L\right], \quad \text{with} \quad \mathbf{Q} = \begin{cases} \mathbf{R}|\mathbf{\Lambda}|\mathbf{L} & \text{upwind}^{11}, \\ |\lambda_i|_{\max}\,\mathbf{I} & \text{Rusanov}^{12}, \\ \dfrac{\Delta x}{\Delta t}\,\mathbf{I} & \text{Lax–Friedrichs}^{13}, \end{cases} \tag{15}$$

where $\mathbf{I} \in \mathbb{R}^{m\times m}$ is the identity matrix. The cell-interface values at the half-time, $\mathbf{u}^{n+k/2}_{j+1/2,L/R}$, are obtained by a Taylor-series expansion of $\mathbf{u}(x,t)$ in space and time using the Cauchy–Kovalevskaya (or Lax–Wendroff) procedure: replacing the time derivative by the spatial derivative,

$$\begin{aligned} \mathbf{u}(x,t) &= \mathbf{u}(x_j,t^n) + (x-x_j)\partial_x\mathbf{u}(x_j,t^n) + (t-t^n)\partial_t\mathbf{u}(x_j,t^n) + O\big(\Delta x^2, \Delta t^2, \Delta x\Delta t\big) \\ &\approx \mathbf{u}^n_j + (x-x_j)\partial_x\mathbf{u}^n_j + (t-t^n)\left[-\partial_x\mathbf{f}(\mathbf{u}^n_j) + \frac{1}{\epsilon}\mathbf{s}(\mathbf{u}^n_j)\right] \\ &= \mathbf{u}^n_j + \big[(x-x_j)\mathbf{I} - (t-t^n)\mathbf{A}(\mathbf{u}^n_j)\big]\,\partial_x\mathbf{u}^n_j + \frac{t-t^n}{\epsilon}\mathbf{s}(\mathbf{u}^n_j), \qquad x \in I_j,\ t \in T^n, \end{aligned} \tag{16}$$

where $\mathbf{A}(\mathbf{u}) = \dfrac{\partial\mathbf{f}}{\partial\mathbf{u}}$ with $\mathbf{A}(\mathbf{u})\colon \mathbb{R}^m \to \mathbb{R}^{m\times m}$. Replacing point values, $\mathbf{u}^n_j$ and $\partial_x\mathbf{u}^n_j$, by cell-averages and undivided gradient values preserves the second-order accuracy since $\mathbf{u}^n_j = \bar{\mathbf{u}}^n_j + O\big(\Delta x^2\big)$ and $\Delta x\partial_x\mathbf{u}^n_j = \overline{\Delta\mathbf{u}}^n_j + O\big(\Delta x^3\big)$. Also, the source term is evaluated from the approximated solution, $\mathbf{u}(x,t)$, instead of the known solution, $\bar{\mathbf{u}}^n_j$, to make the source term implicit. This is again does not affect the order of approximation. Finally, the approximation of the state variable $\mathbf{u}(x,t)$ in domain $I_j \times T^n$ is given by

$$\mathbf{u}(x,t) \approx \bar{\mathbf{u}}^n_j + \big[(x-x_j)\mathbf{I} - (t-t^n)\mathbf{A}(\bar{\mathbf{u}}^n_j)\big]\frac{\overline{\Delta\mathbf{u}}^n_j}{\Delta x} + \frac{t-t^n}{\epsilon}\mathbf{s}\big(\mathbf{u}(x,t)\big). \tag{17}$$

Inserting $x = x_j \pm \dfrac{\Delta x}{2}$ and $t = t^n + \dfrac{k\Delta t}{2}$ leads to the cell-interface values for a Riemann solver:

$$\begin{aligned} \mathbf{u}^{n+k/2}_{j+1/2,L} &= \mathbf{u}_j(x_j + \Delta x/2,\ t^n + k\Delta t/2) \\ &= \bar{\mathbf{u}}^n_j + \frac{1}{2}\left[\mathbf{I} - \frac{k\Delta t}{\Delta x}\mathbf{A}(\bar{\mathbf{u}}^n_j)\right]\overline{\Delta\mathbf{u}}^n_j + \frac{k\Delta t}{2\epsilon}\,\mathbf{s}\big(\mathbf{u}^{n+k/2}_{j+1/2,L}\big), \end{aligned} \tag{18a}$$

$$\begin{aligned} \mathbf{u}^{n+k/2}_{j+1/2,R} &= \mathbf{u}_{j+1}(x_{j+1} - \Delta x/2,\ t^n + k\Delta t/2) \\ &= \bar{\mathbf{u}}^n_{j+1} - \frac{1}{2}\left[\mathbf{I} + \frac{k\Delta t}{\Delta x}\mathbf{A}(\bar{\mathbf{u}}^n_{j+1})\right]\overline{\Delta\mathbf{u}}^n_{j+1} + \frac{k\Delta t}{2\epsilon}\,\mathbf{s}\big(\mathbf{u}^{n+k/2}_{j+1/2,R}\big). \end{aligned} \tag{18b}$$

Note that the implicit character is caused by the source term. In practice, for fluid dynamics equations, this predictor step can be simplified by using a different form of governing equations. Typically, the primitive form leads to a linear source term:

$$\mathbf{s}(\mathbf{w}) = \mathbf{Q}_{\mathrm{w}}\mathbf{w}, \tag{19}$$

where $\mathbf{w} \in \mathbb{R}^m$ is the vector of primitive variables, and $\mathbf{Q}_\mathrm{w} \in \mathbb{R}^{m\times m}$. The constant coefficient matrix $\mathbf{Q}_\mathrm{w}$ can be inverted analytically, thus the predictor step is evaluated explicitly as follows:

$$\mathbf{w}(x,t) = \left[\mathbf{I} - \frac{t-t^n}{\epsilon}\mathbf{Q}_\mathrm{w}\right]^{-1}\left[\overline{\mathbf{w}}_j^n + \Big((x-x_j)\,\mathbf{I} - (t-t^n)\,\mathbf{A}(\overline{\mathbf{w}}_j^n)\Big)\frac{\overline{\Delta\mathbf{w}}_j^n}{\Delta x}\right]. \tag{20}$$

Once primitive variables at the half-time step are obtained, these values are used as the input for a Riemann solver (15) to compute the interface flux.

II.C. Volume Integral of the Source Term

When a DG spatial discretization with a piecewise linear solution representation is applied to hyperbolic-relaxation equations, three volume integrals appear in (11): one is in the update equation of $\bar{\mathbf{u}}_j$, and the other two are in the update equation of $\overline{\Delta\mathbf{u}}_j$. The same strategy as in the upwind moment method can be applied to the latter two volume integrals, assuming state variables in all quadrature points in time are already known. This is true as long as the update equation for $\bar{\mathbf{u}}_j$ is solved first. A difficulty arises when a quadrature rule is applied to the volume integral in (11a). Since this is the update equation of the cell-average variables $\bar{\mathbf{u}}_j$, the updated state variables at a quadrature point, $\bar{\mathbf{u}}_j^{n+1}$, are still unknown. Yet, we can update $\bar{\mathbf{u}}_j$ by iterating with a quadrature for the volume integral of the source term; however, the quadrature rule or points have to be chosen carefully when solving systems of stiff ODEs.

Here, we focus on constructing a third-order discretization in time for the source term, while accepting second-order accuracy in space, so as to circumvent the quadrature in space, more specifically, removing the $\overline{\Delta\mathbf{u}}_j$ dependence in the volume integral of $\mathbf{s}(\mathbf{u}_h)$ in (11a). Thus, the following source term expansion in space is adopted:

$$\mathbf{s}(\mathbf{u}_h(x,t)) = \mathbf{s}(\bar{\mathbf{u}}_j(t)) + \frac{x-x_j}{\Delta x}\mathbf{Q}(\bar{\mathbf{u}}_j(t))\,\overline{\Delta\mathbf{u}}_j(t) + O\big(\Delta x^2\big)\,, \tag{21}$$

where $\mathbf{Q}(\mathbf{u}) = \dfrac{\partial\mathbf{s}}{\partial\mathbf{u}}$ with $\mathbf{Q}(\mathbf{u})\colon \mathbb{R}^m \to \mathbb{R}^{m\times m}$. Inserting (21) into the volume integral of the source term in (11a) leads to

$$\iint\limits_{I_j\times T^n} \mathbf{s}(\mathbf{u}_h(x,t))\,dxdt = \Delta x\int\limits_{T^n}\mathbf{s}(\bar{\mathbf{u}}_j(t))\,dt + O\big(\Delta x^3\big)\,. \tag{22}$$

This approximation removes the coupling between (11a) and (11b), allowing independent updates of the two equations.

In order to integrate the above equation, quadrature points have to be chosen carefully due to its stiffness. Also, to ensure stability in the stiff regime ($\epsilon \ll 1$), the time-integration method for the source term needs to be implicit. Previously, the backward Euler method, which is only first-order accurate, was used to integrate the source term and obtain the intermediate stage.[9] Unfortunately, linear analysis shows that the source-term discretization is overall only second-order accuracy due to the first-order temporal discretization in the intermediate step. In order to achieve high-order accuracy, and circumvent the stiffness, a fully-implicit method is preferable. The properties of the classes of implicit Runge–Kutta methods are tabulated in Table 1. Based on these properties, the Radau IA, Radau IIA, and Lobatto IIIC methods, which possess

Table 1. The properties of the classes of implicit Runge–Kutta methods are tabulated.[14] The order p is based on the linear theory, and the stage order $\tilde{p}$ is the lower bound obtained by the nonlinear theory. Thus, in general, the order of a method q satisfies $\tilde{p} \leq q \leq p$.

s-stage RK method	order p	stage order $\tilde{p}$	linear stability	nonlinear stability
Gauss	$2s$	s	A-stability	algebraic stability
Radau IA	$2s-1$	$s-1$	L-stability	algebraic stability
Radau IIA	$2s-1$	s	L-stability	algebraic stability
Lobatto IIIA	$2s-2$	s	A-stability	No algebraic stability
Lobatto IIIB	$2s-2$	$s-2$	A-stability	No algebraic stability
Lobatto IIIC	$2s-2$	$s-1$	L-stability	algebraic stability

both L-stability and nonlinear stability, are candidates for time integration of the source term. In order to achieve third-order accuracy in time, the Radau IA/IIA methods require only two stages ($s = 2, p = 3$), whereas the Lobatto IIIC method requires three stages ($s = 3, p = 4$). To minimize the computational cost of the fully-implicit procedure for the source term in a scheme, we chose the former, particularly the Radau IIA method, for the source-term integral. Hence, the volume integral of the source term is approximated as

$$\iint\limits_{I_j \times T^n} \mathbf{s}(\mathbf{u}_h(x,t))\, dxdt \approx \Delta x \Delta t \left[\frac{3}{4}\mathbf{s}(\bar{\mathbf{u}}_j^{n+1/3}) + \frac{1}{4}\mathbf{s}(\bar{\mathbf{u}}_j^{n+1})\right], \tag{23}$$

where a new intermediate stage at time level $n + \dfrac{1}{3}$ is introduced. The overall update equations are given by

$$\bar{\mathbf{u}}_j^{n+1/3} = \bar{\mathbf{u}}_j^n - \frac{\Delta t}{3}\frac{1}{\Delta x}\underbrace{\left[\hat{\mathbf{f}}_{j+1/2}^{n+1/6} - \hat{\mathbf{f}}_{j-1/2}^{n+1/6}\right]}_{\text{explicit}} + \frac{\Delta t}{3}\frac{1}{\epsilon}\underbrace{\left[\frac{5}{4}\mathbf{s}(\bar{\mathbf{u}}_j^{n+1/3}) - \frac{1}{4}\mathbf{s}(\bar{\mathbf{u}}_j^{n+1})\right]}_{\text{implicit}}, \tag{24a}$$

$$\bar{\mathbf{u}}_j^{n+1} = \bar{\mathbf{u}}_j^n - \frac{\Delta t}{\Delta x}\underbrace{\left[\hat{\mathbf{f}}_{j+1/2}^{n+1/2} - \hat{\mathbf{f}}_{j-1/2}^{n+1/2}\right]}_{\text{explicit}} + \frac{\Delta t}{\epsilon}\underbrace{\left[\frac{3}{4}\mathbf{s}(\bar{\mathbf{u}}_j^{n+1/3}) + \frac{1}{4}\mathbf{s}(\bar{\mathbf{u}}_j^{n+1})\right]}_{\text{implicit}}. \tag{24b}$$

To solve this system numerically, first the interface fluxes are computed explicitly. Then, the problem reduces to finding the solutions of systems of nonlinear algebraic equations of the following form:

$$\mathbf{u}_{\mathrm{A}} = \mathbf{C}_{\mathrm{f}} + \mathbf{C}_{\mathrm{s}}\ \mathbf{s}_{\mathrm{A}}(\mathbf{u}_{\mathrm{A}}), \tag{25}$$

where

$$\mathbf{u}_{\mathrm{A}} = \begin{pmatrix} \bar{\mathbf{u}}_j^{n+1/3} \\ \bar{\mathbf{u}}_j^{n+1} \end{pmatrix}, \qquad \mathbf{s}_{\mathrm{A}}(\mathbf{u}_{\mathrm{A}}) = \begin{pmatrix} \mathbf{s}(\bar{\mathbf{u}}_j^{n+1/3}) \\ \mathbf{s}(\bar{\mathbf{u}}_j^{n+1}) \end{pmatrix}, \tag{26a}$$

$$\mathbf{C}_{\mathrm{f}} = \begin{pmatrix} \bar{\mathbf{u}}_j^n - \dfrac{\Delta t}{3}\dfrac{1}{\Delta x}\left[\hat{\mathbf{f}}_{j+1/2}^{n+1/6} - \hat{\mathbf{f}}_{j-1/2}^{n+1/6}\right] \\ \bar{\mathbf{u}}_j^n - \dfrac{\Delta t}{\Delta x}\left[\hat{\mathbf{f}}_{j+1/2}^{n+1/2} - \hat{\mathbf{f}}_{j-1/2}^{n+1/2}\right] \end{pmatrix}, \qquad \mathbf{C}_{\mathrm{s}} = \frac{\Delta t}{\epsilon}\begin{pmatrix} \dfrac{5}{12}\mathbf{I} & -\dfrac{1}{12}\mathbf{I} \\ \dfrac{3}{4}\mathbf{I} & \dfrac{1}{4}\mathbf{I} \end{pmatrix}, \tag{26b}$$

with $\mathbf{u}_{\mathrm{A}}, \mathbf{C}_{\mathrm{f}} \in \mathbb{R}^{2m}, \mathbf{C}_{\mathrm{s}} \in \mathbb{R}^{2m \times 2m}$, and $\mathbf{s}_{\mathrm{A}}(\mathbf{u}_{\mathrm{A}}) \colon \mathbb{R}^{2m} \to \mathbb{R}^{2m}$. Here, the Newton-Raphson method is adopted to find the solution, thus the iteration process at p-step is given by

$$\mathbf{u}_{\mathrm{A}}^{p+1} = \mathbf{u}_{\mathrm{A}}^p - \left[\mathbf{I}_{\mathrm{A}} - \mathbf{C}_{\mathrm{s}}\mathbf{Q}_{\mathrm{A}}(\mathbf{u}_{\mathrm{A}}^p)\right]^{-1}\left[\mathbf{u}_{\mathrm{A}}^p - \mathbf{C}_{\mathrm{f}} - \mathbf{C}_{\mathrm{s}}\ \mathbf{s}_{\mathrm{A}}(\mathbf{u}_{\mathrm{A}}^p)\right], \quad p = 0, 1, 2\ldots, \tag{27}$$

where

$$\mathbf{Q}_{\mathrm{A}}(\mathbf{u}_{\mathrm{A}}) = \frac{\partial \mathbf{s}_{\mathrm{A}}}{\partial \mathbf{u}_{\mathrm{A}}} = \begin{pmatrix} \mathbf{Q}(\bar{\mathbf{u}}_j^{n+1/3}) & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}(\bar{\mathbf{u}}_j^{n+1}) \end{pmatrix}, \quad \mathbf{Q}_{\mathrm{A}}(\mathbf{u}_{\mathrm{A}}) \colon \mathbb{R}^{2m} \to \mathbb{R}^{2m\times 2m}, \tag{28}$$

and $\mathbf{I}_{\mathrm{A}} \in \mathbb{R}^{2m\times 2m}$. To start up the iteration, the initial guess at the time level n, $\mathbf{u}_{\mathrm{A}}^0 = \left[\bar{\mathbf{u}}_j^n; \bar{\mathbf{u}}_j^n\right]$, is used. The iteration on the system of $2m$ equations can be reduced to $2(m-l)$ equations where $l < n$, since the first l entries of the source term are zero.

In general, when the Newton-Raphson method is implemented, it is more efficient to adopt LU-decomposition to the matrix $[\mathbf{I}_{\mathrm{A}} - \mathbf{C}_{\mathrm{s}}\mathbf{Q}_{\mathrm{A}}(\mathbf{u}_{\mathrm{A}}^p)]$, and solve the system of linear algebraic equations instead of inverting it. However, the structure of the source term is typically simple, and the inverse matrix, $[\mathbf{I}_{\mathrm{A}} - \mathbf{C}_{\mathrm{s}}\mathbf{Q}_{\mathrm{A}}(\mathbf{u}_{\mathrm{A}}^p)]^{-1}$, can be obtained analytically as a function of $\mathbf{u}_{\mathrm{A}}$. The advantage of the choice of hyperbolic-relaxation equations over the NS equations is clear here: since the method is point-implicit due to the source term, the inverse of the matrix is still local, whereas the implicit treatment of the diffusion term in the NS equations makes the domain of dependence global.

II.D. Volume Integral of the Flux

II.D.1. Gauss–Lobatto Points (Original Upwind Moment Scheme)

First, we review Huynh's original upwind moment scheme.[5] The method utilizes the three-point Gauss–Lobatto quadrature for both space and time integration of the flux, thus the volume integral is approximated by

$$\begin{aligned} \iint\limits_{I_j\times T^n} \mathbf{f}(\mathbf{u}_h(x,t))\,dxdt &\approx \Delta t \int\limits_{I_j} \frac{1}{6}\Big[\mathbf{f}(\mathbf{u}(x,t^n)) + 4\mathbf{f}(\mathbf{u}(x,t^{n+1/2})) + \mathbf{f}(\mathbf{u}(x,t^{n+1}))\Big]\,dx \\ &\approx \Delta t \int\limits_{I_j} \mathbf{f}(\hat{\mathbf{u}}(x))\,dx \\ &\approx \Delta t \frac{\Delta x}{6}\left[\mathbf{f}(\hat{\mathbf{u}}(x_{j-1/2})) + 4\mathbf{f}(\hat{\mathbf{u}}(x_j)) + \mathbf{f}(\hat{\mathbf{u}}(x_{j+1/2}))\right], \end{aligned} \tag{29}$$

where

$$\hat{\mathbf{u}}(x) = \hat{\mathbf{u}}_j + \widehat{\overline{\Delta\mathbf{u}}}_j \frac{x-x_j}{\Delta x}, \tag{30a}$$

$$\hat{\mathbf{u}}_j = \frac{1}{6}\left(\bar{\mathbf{u}}_j^n + 4\bar{\mathbf{u}}_j^{n+1/2} + \bar{\mathbf{u}}_j^{n+1}\right), \tag{30b}$$

$$\widehat{\overline{\Delta\mathbf{u}}}_j = \frac{1}{2}\left(\overline{\Delta\mathbf{u}}_j^n + \overline{\Delta\mathbf{u}}_j^{n+1}\right). \tag{30c}$$

Here, $\hat{(\cdot)}$ denotes a time-averaged value. When the discretization of conservation laws, (11) with $\mathbf{s} = 0$, is considered, the volume integral appears only in the second equation (11b). Since the cell-average variables are updated first, $\bar{\mathbf{u}}_j^{n+1}$ is already known when the volume integral is evaluated. Thus $\hat{\mathbf{u}}_j$ is evaluated explicitly whereas $\overline{\Delta\mathbf{u}}_j^{n+1}$ in (30c) is still unknown, and an iteration process is required. To start the iteration process, the slope at the time level n is used as the initial guess; however, Huynh reported that no improvement was observed by iterations, and suggested to replace (30c) by

$$\widehat{\overline{\Delta\mathbf{u}}}_j = \overline{\Delta\mathbf{u}}_j^n. \tag{31}$$

When the flux is linear, $\mathbf{f}(\mathbf{u}) = \mathbf{A}\mathbf{u}$, the volume integral (29) is evaluated exactly in space and approximately in time, thus

$$\begin{aligned}\iint\limits_{I_j \times T^n} \mathbf{f}(\mathbf{u}_h(x,t))\, dxdt &= \Delta x \mathbf{A} \int\limits_{T^n} \bar{\mathbf{u}}_j(t)\, dt \\ &\approx \Delta x \Delta t\ \mathbf{A}\hat{\bar{\mathbf{u}}}_j, \qquad (32)\end{aligned}$$

where $\hat{\bar{\mathbf{u}}}_j$ is given by (30b). In (30b), the new intermediate stage, $\bar{\mathbf{u}}_j^{n+1/2}$, is introduced, and computed in advance by updating over a half time step:

$$\begin{aligned}\bar{\mathbf{u}}_j^{n+1/2} &= \bar{\mathbf{u}}_j^n - \frac{1}{\Delta x} \int\limits_{t^n}^{t^{n+1/2}} \left[\mathbf{f}_{j+1/2}(t) - \mathbf{f}_{j-1/2}(t)\right] dt \\ &= \bar{\mathbf{u}}_j^n - \frac{\Delta t}{2} \frac{1}{\Delta x} \left[\hat{\mathbf{f}}_{j+1/2}^{n+1/4} - \hat{\mathbf{f}}_{j-1/2}^{n+1/4}\right]. \qquad (33)\end{aligned}$$

Note that the method only requires the cell-average value at the half-time step, not the undivided gradient, since the slope at the time level n is used in the entire space-time domain according to (31). Once the intermediate state is obtained, from (11), the final update equations become

$$\bar{\mathbf{u}}_j^{n+1} = \bar{\mathbf{u}}_j^n - \frac{\Delta t}{\Delta x} \left[\hat{\mathbf{f}}_{j+1/2}^{n+1/2} - \hat{\mathbf{f}}_{j-1/2}^{n+1/2}\right], \qquad (34a)$$

$$\overline{\Delta \mathbf{u}}_j^{n+1} = \overline{\Delta \mathbf{u}}_j^n - \frac{\Delta t}{\Delta x} 6 \left[\hat{\mathbf{f}}_{j+1/2}^{n+1/2} + \hat{\mathbf{f}}_{j-1/2}^{n+1/2} - \frac{2}{\Delta x \Delta t} \overline{\mathbf{f}(\hat{\mathbf{u}}_j)}\right], \qquad (34b)$$

where $\overline{\mathbf{f}(\hat{\mathbf{u}}_j)}$ is given by (29). In summary, the original upwind moment scheme is a one-step method with one intermediate stage for the volume integral, requiring two Riemann solvers at each cell-interface.

II.D.2. Gauss–Radau Points

When the hyperbolic-relaxation equations are considered, quadrature points for the flux in (11b) need to be modified based on the quadrature for the source term. Since we adopt the two-point Radau IIA method (23) as the time integrator for the source term, the same Radau points are employed for the volume integral of the flux. Hence, the Gauss–Radau quadrature in time and the Gauss–Lobatto quadrature in space are applied:

$$\begin{aligned}\iint\limits_{I_j \times T^n} \mathbf{f}\big(\mathbf{u}_h(x,t)\big)\, dxdt &= \Delta t \int\limits_{I_j} \left[\frac{3}{4}\mathbf{f}\big(\mathbf{u}(x, t^{n+1/3})\big) + \frac{1}{4}\mathbf{f}\big(\mathbf{u}(x, t^{n+1})\big)\right] dx + O\big(\Delta t^4\big) \\ &\approx \Delta t \int\limits_{I_j} \mathbf{f}\big(\tilde{\mathbf{u}}(x)\big)\, dx \\ &\approx \Delta t \frac{\Delta x}{6} \left[\mathbf{f}\big(\tilde{\mathbf{u}}(x_{j-1/2})\big) + 4\mathbf{f}\big(\tilde{\mathbf{u}}(x_j)\big) + \mathbf{f}\big(\tilde{\mathbf{u}}(x_{j+1/2})\big)\right], \qquad (35)\end{aligned}$$

where

$$\tilde{\mathbf{u}}(x) = \tilde{\mathbf{u}}_j + \overline{\Delta \mathbf{u}}_j^n \frac{x - x_j}{\Delta x} \quad \text{with} \quad \tilde{\mathbf{u}}_j = \frac{1}{4}\left(3\bar{\mathbf{u}}_j^{n+1/3} + \bar{\mathbf{u}}_j^{n+1}\right). \qquad (36)$$

Here, the undivided gradient is frozen at the time level n in order to keep the treatment explicit. In the update equation of the undivided gradient, the intermediate-stage value, $\overline{\Delta \mathbf{u}}_j^{n+1/3}$, is required, and another volume integral of the flux over the time domain $T^{n'} = [t^n, t^{n+1/3}]$ is needed. Since the flux is not a stiff term and the cell-averaged variables at the quadrature points $k = 0, \dfrac{1}{3}$ are known, the trapezoidal rule is applied in time while the Gauss–Lobatto quadrature is applied in space:

$$\begin{aligned}
\iint\limits_{I_j\times T^{n'}} \mathbf{f}\big(\mathbf{u}_h(x,t)\big)\,dxdt &= \frac{\Delta t}{3}\int\limits_{I_j}\frac{1}{2}\Big[\mathbf{f}\big(\mathbf{u}(x,t^n)\big)+\mathbf{f}\big(\mathbf{u}(x,t^{n+1/3})\big)\Big]\,dx + O\big(\Delta t^3\big)\\
&\approx \frac{\Delta t}{3}\int\limits_{I_j}\mathbf{f}\big(\check{\mathbf{u}}(x)\big)\,dx\\
&\approx \frac{\Delta t}{3}\frac{\Delta x}{6}\left[\mathbf{f}\big(\check{\mathbf{u}}(x_{j-1/2})\big)+4\mathbf{f}\big(\check{\mathbf{u}}(x_j)\big)+\mathbf{f}\big(\check{\mathbf{u}}(x_{j+1/2})\big)\right],
\end{aligned}\tag{37}$$

where

$$\check{\mathbf{u}}(x) = \check{\bar{\mathbf{u}}}_j + \overline{\Delta\mathbf{u}}_j^n\frac{x-x_j}{\Delta x} \quad \text{with} \quad \check{\bar{\mathbf{u}}}_j = \frac{1}{2}\left(\bar{\mathbf{u}}_j^n + \bar{\mathbf{u}}_j^{n+1/3}\right). \tag{38}$$

II.E. Integral of the Moment of the Source Term

The second-order approximation of the source term, (21), is inserted into the volume integral of the moment of the source term in (11b), and the Gauss–Radau quadrature is used for time:

$$\begin{aligned}
\iint\limits_{I_j\times T^{n}} \mathbf{s}(\mathbf{u}_h(x,t))\frac{x-x_j}{\Delta x}\,dxdt &= \frac{\Delta x}{12}\int\limits_{T^n}\mathbf{Q}(\bar{\mathbf{u}}_j(t))\overline{\Delta\mathbf{u}}_j(t)\,dt + O\big(\Delta x^3\big)\\
&\approx \frac{\Delta x\Delta t}{12}\left[\frac{3}{4}\mathbf{Q}(\bar{\mathbf{u}}_j^{n+1/3})\overline{\Delta\mathbf{u}}_j^{n+1/3} + \frac{1}{4}\mathbf{Q}(\bar{\mathbf{u}}_j^{n+1})\overline{\Delta\mathbf{u}}_j^{n+1}\right].
\end{aligned}\tag{39}$$

Following the same procedure for the cell-average update, the Radau IIA method is adopted for the source term, and the final update formulas for the undivided gradient are given by

$$\overline{\Delta\mathbf{u}}_j^{n+1/3} = \overline{\Delta\mathbf{u}}_j^{n} - \frac{\Delta t}{3}\frac{6}{\Delta x}\underbrace{\left[\hat{\mathbf{f}}_{j+1/2}^{n+1/6} + \hat{\mathbf{f}}_{j-1/2}^{n+1/6} - \frac{2}{\Delta x\Delta t}\overline{\mathbf{f}(\mathbf{u}_j^{n+1/6})}\right]}_{\text{explicit}} + \frac{\Delta t}{3}\frac{1}{\epsilon}\underbrace{\left[\frac{5}{4}\mathbf{Q}(\bar{\mathbf{u}}_j^{n+1/3})\overline{\Delta\mathbf{u}}_j^{n+1/3} - \frac{1}{4}\mathbf{Q}(\bar{\mathbf{u}}_j^{n+1})\overline{\Delta\mathbf{u}}_j^{n+1}\right]}_{\text{implicit}}, \tag{40a}$$

$$\overline{\Delta\mathbf{u}}_j^{n+1} = \overline{\Delta\mathbf{u}}_j^{n} - \frac{\Delta t}{\Delta x}6\underbrace{\left[\hat{\mathbf{f}}_{j+1/2}^{n+1/2} + \hat{\mathbf{f}}_{j-1/2}^{n+1/2} - \frac{2}{\Delta x\Delta t}\overline{\mathbf{f}(\mathbf{u}_j^{n+1/2})}\right]}_{\text{explicit}} + \frac{\Delta t}{\epsilon}\underbrace{\left[\frac{3}{4}\mathbf{Q}(\bar{\mathbf{u}}_j^{n+1/3})\overline{\Delta\mathbf{u}}_j^{n+1/3} + \frac{1}{4}\mathbf{Q}(\bar{\mathbf{u}}_j^{n+1})\overline{\Delta\mathbf{u}}_j^{n+1}\right]}_{\text{implicit}}, \tag{40b}$$

where $\overline{\mathbf{f}(\mathbf{u}_j^{n+1/6})}$ and $\overline{\mathbf{f}(\mathbf{u}_j^{n+1/2})}$ are obtained by (37) and (35) respectively. Once the interface fluxes and volume integrals of fluxes are computed explicitly, the problem is again reduced to solving a system of linear algebraic equations:

$$\Delta\mathbf{u}_{\mathrm{A}} = \Delta\mathbf{C}_{\mathrm{f}} + \mathbf{C}_{\mathrm{s}}\mathbf{Q}_{\mathrm{A}}(\mathbf{u}_{\mathrm{A}})\Delta\mathbf{u}_{\mathrm{A}}, \tag{41}$$

where

$$\Delta\mathbf{u}_{\mathrm{A}} = \begin{pmatrix}\overline{\Delta\mathbf{u}}_j^{n+1/3}\\ \overline{\Delta\mathbf{u}}_j^{n+1}\end{pmatrix}, \quad \Delta\mathbf{C}_{\mathrm{f}} = \begin{pmatrix}\overline{\Delta\mathbf{u}}_j^{n} - \dfrac{\Delta t}{3}\dfrac{6}{\Delta x}\left[\hat{\mathbf{f}}_{j+1/2}^{n+1/6} + \hat{\mathbf{f}}_{j-1/2}^{n+1/6} - \dfrac{2}{\Delta x\Delta t}\overline{\mathbf{f}(\mathbf{u}_j^{n+1/6})}\right]\\ \overline{\Delta\mathbf{u}}_j^{n} - \dfrac{\Delta t}{\Delta x}6\left[\mathbf{f}_{j+1/2}^{n+1/2} + \mathbf{f}_{j-1/2}^{n+1/2} - \dfrac{2}{\Delta x\Delta t}\overline{\mathbf{f}(\mathbf{u}_j^{n+1/2})}\right]\end{pmatrix}, \tag{42}$$

with $\Delta\mathbf{u}_{\mathrm{A}}, \Delta\mathbf{C}_{\mathrm{f}} \in \mathbb{R}^{2m}$. Since $\mathbf{u}_{\mathrm{A}}$ is already known by (27), no iteration is required, and $\Delta\mathbf{u}_{\mathrm{A}}$ is obtained by

$$\Delta\mathbf{u}_{\mathrm{A}} = \left[\mathbf{I}_{\mathrm{A}} - \mathbf{C}_{\mathrm{s}}\mathbf{Q}_{\mathrm{A}}(\mathbf{u}_{\mathrm{A}})\right]^{-1}\Delta\mathbf{C}_{\mathrm{f}}. \tag{43}$$

As mentioned previously, the structure of the source term is typically simple; the inverse of the above matrix can be obtained analytically.

III. Analysis for 1-D and 2-D Linear Advection Equations

III.A. Discretization Methods

A Fourier analysis is employed to uncover the linear properties of methods for hyperbolic conservation laws without source terms; hyperbolic-relaxation systems will be analyzed in Section IV. The analysis is also called a 'Von Neumann analysis' named after John von Neumann who originally introduced the analysis for parabolic differential equations.[15] The actual applications of the analysis can be found in many textbooks.[16, 17] The analysis shows the order of accuracy, the dominant numerical dissipation/dispersion errors for the low-frequence mode, and the linear stability of a method. Note that the assumptions required for a Fourier analysis are uniform grids and periodic boundary conditions. The dimensionless 1-D and 2-D linear advection equations,

$$\partial_t u + r\partial_x u = 0, \quad |r| \leq 1, \tag{44a}$$

$$\partial_t u + r\partial_x u + s\partial_y u = 0, \quad |r|, |s| \leq 1, \tag{44b}$$

are considered as the model equations. Here, the normalization is rather uncommon. The advection speed is normalized by the larger 'frozen' wave speed (= 1) arising further on in hyperbolic-relaxation equations. The motivation of this normalization will be clear once hyperbolic-relaxation systems are considered.

The Courant number, ν, is defined by the dimensionless frozen wave speed, 1, instead of the advection speed, r, thus

$$\nu := 1\frac{\Delta t}{\Delta x}, \quad \Delta x, \Delta t \in \mathbb{R}^{+}. \tag{45}$$

Again, this definition is rather uncommon. Conventionally, the Courant number for a linear advection equation is defined by

$$\tilde{\nu} := r\frac{\Delta t}{\Delta x}, \tag{46}$$

where the advection speed, r, is normalized by the spatial and temporal scales. To make analysis consistent with the results later presented for the linear hyperbolic-relaxation equations, we adopt ν as the Courant number here. The conventional expression can be recovered by substituting

$$r\nu = \tilde{\nu}. \tag{47}$$

Recall that, for a linear advection equation, both the upwind moment scheme (with Gauss–Lobatto quadrature for the volume integral) and the proposed method (with Gauss–Radau quadrature) are identical to Van Leer's scheme III.[8] The upwind moment method (DG(1)–Hancock) is compared with three other

Table 2. The combinations of space and time discretization methods. First row: semi-discrete methods; second row: the fully discrete methods.

	High-Resolution Godunov (HR)	Discontinuous Galerkin (DG)
Runge-Kutta (RK)	HR2–RK2, HR2–RK3	DG(1)–RK2, DG(1)–RK3
Hancock (Ha)	Hancock (HR2–Ha)	Upwind moment (DG(1)–Ha)

methodologies: a semi-discrete higher-resolution Godunov method (HR) with method-of-lines (MOL) or Hancock time integration, and a DG(1)–MOL method. These methods can be regarded as the combination of a spatial and a temporal discretization, and are tabulated in Table 2. Here, we adopt the notations HRs and RKs, where s is the order of accuracy, and DG(k) where k is the degree of the polynomial basis. A Fourier analysis for the 1-D advection equation shows that the upwind moment method is linearly stable up to the Courant number 1 with an upwind flux, whereas DG spatial discretizations combined with MOL typically have a more strict stability condition: for DG(1)–RK2 (second-order) the limit is $\frac{1}{3}$, and for the DG(2)–RK3 (third-order) it is $\frac{1}{5}$.[3]

III.B. Methodology

III.B.1. Difference Operators in Fourier (Frequency) Space

To investigate and compare the properties of a method, it is useful to write a method in compact form. Let the forward, δ^+, and backward, δ^-, difference operator be

$$\delta^+\mathbf{u}_j = \mathbf{u}_{j+1} - \mathbf{u}_j, \tag{48a}$$

$$\delta^-\mathbf{u}_j = \mathbf{u}_j - \mathbf{u}_{j-1}. \tag{48b}$$

Then, translation over any number of cells can be expressed by applying these difference operators multiple times, e.g.,

$$\mathbf{u}_{j+2} = (I + \delta^+)^2\mathbf{u}_j, \tag{49a}$$

$$\mathbf{u}_{j-2} = (I - \delta^-)^2\mathbf{u}_j, \tag{49b}$$

where I is the identity operator, that is, $\mathbf{u}_j = I\mathbf{u}_j$. Using these difference operators, the simplest fully discrete methods can be expressed as

$$\mathbf{u}_j^{n+1} = \mathbf{G}(\nu, r, q)\mathbf{u}_j^n, \tag{50}$$

where $\mathbf{G}$ is an amplification factor or matrix, and q is the dissipation parameter of the q-flux (15).[18] Since the fully discrete method considered here is a multi-stage one-step method, it can be written as the forward Euler method in time, thus

$$\frac{\mathbf{u}_j^{n+1} - \mathbf{u}_j^n}{\Delta t} = \mathbf{M}(\nu, r, q, \Delta x)\,\mathbf{u}_j^n, \quad \text{or} \quad \mathbf{u}_j^{n+1} = [\mathbf{I} + \Delta t\,\mathbf{M}(\nu, r, q, \Delta x)]\,\mathbf{u}_j^n, \tag{51}$$

where $\mathbf{M}$ is a 'spatial-temporal' difference operator. Comparing to the (50), the amplification matrix, $\mathbf{G}$, can be expressed in terms of $\mathbf{M}$:

$$\text{fully discrete: } \mathbf{G}_{\mathrm{M}} = \mathbf{I} + \Delta t\, \mathbf{M}. \tag{52}$$

Conversely, a semi-discrete method is only expressed in an ODE form:

$$\frac{\partial \mathbf{u}_j(t)}{\partial t} = \mathbf{N}(r, q, \Delta x)\, \mathbf{u}_j(t), \tag{53}$$

where $\mathbf{N}$ is a 'spatial' difference operator. The notation of the two difference operators is made distinct on purpose: to a fully discrete method is assigned $\mathbf{M}$, and to a semi-discrete method $\mathbf{N}$. Note that since a semi-discrete method is only discretized in space, $\mathbf{N}$ is not a function of ν, thus an ODE solver is in charge of the temporal discretization. Here, two RK methods are adopted for the time integration in semi-discrete methods. The two- and three-stage RK methods are second- and third-order accurate in time respectively. For a system of linear equations, a RK method simply generates the series expansion of the matrix exponential $e^{\mathbf{N}\Delta t}$ up to a certain order, thus

$$\text{RK2: } \mathbf{G}_{\mathrm{N}} = \mathbf{I} + \Delta t\, \mathbf{N} + \frac{\Delta t^2}{2}\mathbf{N}^2, \tag{54a}$$

$$\text{RK3: } \mathbf{G}_{\mathrm{N}} = \mathbf{I} + \Delta t\, \mathbf{N} + \frac{\Delta t^2}{2}\mathbf{N}^2 + \frac{\Delta t^3}{6}\mathbf{N}^3. \tag{54b}$$

In a Fourier analysis, the investigation of the amplification factor $\mathbf{G}$ is the main interest. When a finite-volume method is applied to a scalar linear equation, this factor is a scalar, thus the derivation is straightforward. However, when a DG method is applied, even though the target equation is a scalar equation, an amplification factor becomes a matrix due to the introduction of extra variables in each cell. To extract the behavior of a method, eigenvalues of $\mathbf{G}$ must be obtained by solving a characteristic equation,

$$\det(\mathbf{G} - g\mathbf{I}) = 0, \tag{55}$$

where g is an eigenvalue of $\mathbf{G}$.

For a single Fourier mode, the solution at cell j can be represented as follows:

$$\begin{aligned} \mathbf{u}_j &= \hat{\mathbf{u}}_0 \exp\left(i\frac{2\pi j}{N}\right) \\ &= \hat{\mathbf{u}}_0 \exp(i\beta j), \quad \beta \in [-\pi, \pi], \end{aligned} \tag{56}$$

where β is the spatial frequency of the wave. Here, $\beta = 0$ corresponds to the low-frequency limit, and $\beta = \pi$ is the high-frequency limit. Using a Fourier representation, the difference operators are now replaced by exponential functions,

$$\delta^+ = e^{i\beta} - 1, \tag{57a}$$

$$\delta^- = 1 - e^{-i\beta}. \tag{57b}$$

Inserting these relations into a matrix operator, $\mathbf{M}$ or $\mathbf{N}$, removes the difference operators in an amplification matrix.

III.B.2. Exact Solution

An exact solution is critical to examining the order of accuracy of a method. The exact solution of (44a) in the harmonic mode is given by

$$\mathbf{u}(x,t) = \hat{\mathbf{u}}_0 e^{ik(x-rt)}, \tag{58}$$

where k is the spatial wave number. The exact amplification factor is obtained by expressing $\mathbf{u}(x,t+\Delta t)$ in terms of $\mathbf{u}(x,t)$,

$$\begin{aligned}\mathbf{u}(x,t+\Delta t) &= \hat{\mathbf{u}}_0 e^{ik(x-rt)} e^{-irk\Delta t} \\ &= e^{-irk\Delta t}\mathbf{u}(x,t).\end{aligned} \tag{59}$$

It shows that the exact amplification factor for the time step Δt, and the exact eigenvalue of the spatial discretization operator in (53) are given by

$$g_{\text{exact}} = e^{-irk\Delta t}, \quad \lambda_{\text{exact}} = -irk. \tag{60}$$

In a Fourier mode, the wave number k is related to the frequency of a wave β by

$$k = \frac{\beta}{\Delta x}, \tag{61}$$

thus the amplification factor and spatial eigenvalue become

$$g_{\text{exact}}(\tilde{\nu},\beta) = e^{-ir\nu\beta} = e^{-i\tilde{\nu}\beta}, \quad \lambda_{\text{exact}} = -\frac{ir}{\Delta x}\beta. \tag{62}$$

III.C. Difference Operators and Their Properties in 1-D

In this section, a Fourier analysis is employed to uncover the dominant truncation errors in the low-frequency limit, the order of accuracy, and the stability conditions of various discretization methods.

III.C.1. HR–MOL Method

A semi-discrete high-resolution Godunov method combined with the method-of-lines (HR–MOL) for the 1-D linear advection equation (44a) has the following form:

$$\frac{\partial \bar{u}_j(t)}{\partial t} = -\frac{1}{\Delta x}\left(\hat{f}_{j+1/2}(t) - \hat{f}_{j-1/2}(t)\right), \tag{63}$$

where the linear flux, $f(u(t)) = ru(t)$, is evaluated at each cell interface. The interface flux $\hat{f}_{j\pm1/2}(t)$ must be evaluated at various time levels, depending on the time discretization method (ODE solver) chosen. The interface flux is obtained as in the q-flux (15). For the linear advection equation, it becomes

$$\hat{f}_{j+1/2}(t) = \frac{r}{2}\big(u_{j+1/2,L}(t) + u_{j+1/2,R}(t)\big) - \frac{q}{2}\big(u_{j+1/2,R}(t) - u_{j+1/2,L}(t)\big), \quad q > 0, \tag{64}$$

where the input values of the flux function are the linearly reconstructed values at the cell interface $j+1/2$:

$$u_{j+1/2,L}(t) = \bar{u}_j + \frac{\Delta x}{2}\left(\frac{\bar{u}_{j+1} - \bar{u}_{j-1}}{2\Delta x}\right), \tag{65a}$$

$$u_{j+1/2,R}(t) = \bar{u}_{j+1} - \frac{\Delta x}{2}\left(\frac{\bar{u}_{j+2} - \bar{u}_j}{2\Delta x}\right). \tag{65b}$$

The linear reconstruction of the original piecewise constant values leads an HR–MOL method to second-order accuracy in space whereas the original Godunov method, which uses the piecewise constant data, is first-order accurate. Here, we denote the second-order semi-discrete Godunov method as HR2–MOL.

After inserting the cell interface fluxes into the original semi-discrete form (63), and some algebra, the spatial difference operator, $\mathrm{N}_{\mathrm{HR2}}$, in the semi-discrete form of (53) is given by

$$\mathrm{N}_{\mathrm{HR2}} = -\frac{1}{8\Delta x}\left[(q-r)(\delta^+)^2 - 2(q-2r)\delta^+ + 2(q+2r)\delta^- + (q+r)(\delta^-)^2\right], \tag{66a}$$

or, for a Fourier mode, using (57),

$$\mathrm{N}_{\mathrm{HR2}} = -\frac{1}{8\Delta x}\left[(q-r)e^{2i\beta} + 2(3r-2q)e^{i\beta} + 6q - 2(2q+3r)e^{-i\beta} + (q+r)e^{-2i\beta}\right]. \tag{66b}$$

The identical result can be obtained by setting $\nu = 0$ in the spatial-temporal operator for the Hancock method (86), derived further below.

Accuracy Taking the low-frequency limit of the difference spatial operator, $\mathrm{N}_{\mathrm{HR2}}$, leads to the asymptotic eigenvalue:

$$\lambda_{\mathrm{HR2}} = -\frac{ir}{\Delta x}\beta - \frac{ir}{12\Delta x}\beta^3 - \frac{q}{8\Delta x}\beta^4 + O(\beta^5). \tag{67}$$

The order of accuracy in space is obtained by replacing β by the wave number k, then

$$\lambda_{\mathrm{HR2}} - \lambda_{\mathrm{exact}} = -\frac{ir}{12}\boxed{\Delta x^2}\,k^3 - \frac{q}{8}\Delta x^3\,k^4 + O(k^5); \tag{68}$$

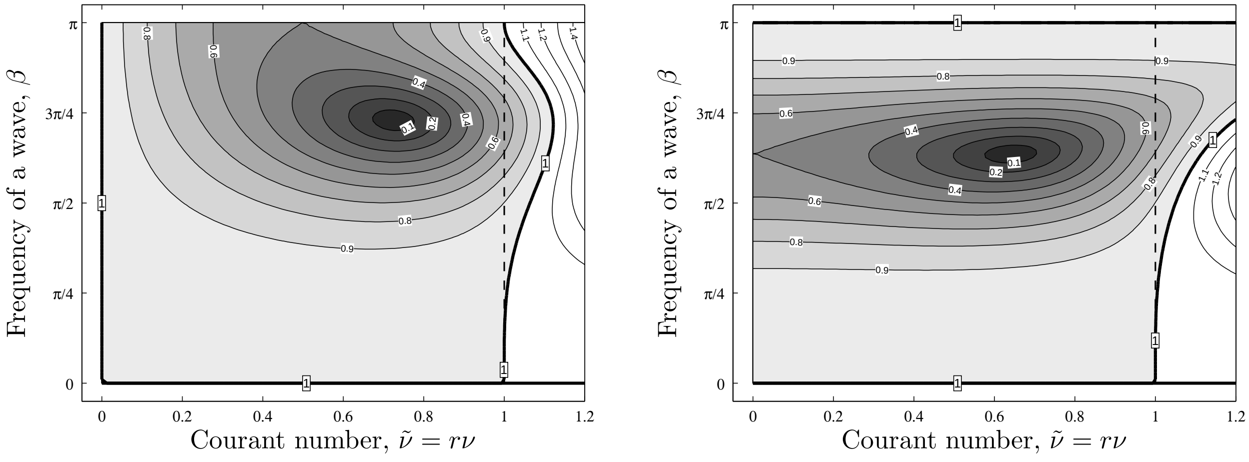
the method appears to be second-order accurate in space. To examine the overall order of accuracy, the RK2 and RK3 method (54) are employed for the time integration; their local truncation errors have the following forms:

$$\mathrm{LTE}_{\mathrm{HR2RK2}} = -\frac{ir}{12}\left(\boxed{\Delta x^2} + 2r^2\boxed{\Delta t^2}\right)k^3 - \frac{r}{8}\left[\frac{q}{r} - (r\nu)^3\right]\Delta x^3 k^4 + O(k^5), \tag{69a}$$

$$\mathrm{LTE}_{\mathrm{HR2RK3}} = -\frac{ir}{12}\boxed{\Delta x^2}\,k^3 - \frac{r}{8}\left(\frac{q}{r}\Delta x^3 + \frac{1}{3}r^3\boxed{\Delta t^3}\right)k^4 + O(k^5). \tag{69b}$$

Thus, the HR2–RK2 method is second-order accurate in space and time, and the HR2–RK3 is second-order in space and third-order in time. It clearly shows that the third-order time integration method (RK3) eliminates the Δt^2–term in the HR2–RK2 method, making the method third-order accurate in time. However, since a semi-discrete method decouples the space and time discretizations, a higher-order time integration can not eliminate the second-order spatial discretization error. Thus, the HR2–RK3 method is still second-order in space.

Stability We adopt a numerical approach to investigate the linear stability condition. The modulus of the amplification factor, $|g_{\mathrm{HR2RK2}}(\tilde{\nu},\beta)|$, evaluated with the upwind and Lax–Friedrichs fluxes, is shown in Figures 1(a) and 1(b) respectively. The shaded area indicates the stability region, where $|g_{\mathrm{HR2RK2}}(\tilde{\nu},\beta)| \leq 1$. These two figures show that the HR2–RK2 method is linearly stable for $\tilde{\nu} \leq 1$ with both the upwind and the Lax–Friedrichs fluxes.



(a) HR2–RK2 method with the upwind flux

(b) HR2–RK2 method with the Lax–Friedrichs flux

Figure 1. Contour plots of the modulus of the amplification factor, $|g_{\mathbf{HR2RK2}}(\tilde{\nu}, \beta)|$, computed with the upwind (left) and Lax–Friedrichs (right) fluxes. These show that the HR2–RK2 method is stable for $\tilde{\nu} \leq 1$.

III.C.2. DG–MOL Method

A semi-discrete DG method combined with the method-of-lines (DG–MOL) for the 1-D linear advection equation (44a) has update formulas for both the cell-average and undivided gradient,

$$\frac{\partial \bar{u}_j(t)}{\partial t} = -\frac{1}{\Delta x}\left(\hat{f}_{j+1/2}(t) - \hat{f}_{j-1/2}(t)\right), \tag{70a}$$

$$\frac{\partial \overline{\Delta u}_j(t)}{\partial t} = -\frac{1}{\Delta x}6\left(\hat{f}_{j+1/2}(t) + \hat{f}_{j-1/2}(t) - 2r\bar{u}_j(t)\right), \tag{70b}$$

where the volume integral of the flux in the second equation simplifies owing to the linearity. The q-flux (64) is adopted for the cell interface fluxes with linearly interpolated values:

$$u_{j+1/2,L} = \bar{u}_j + \frac{1}{2}\overline{\Delta u}_j, \tag{71a}$$

$$u_{j+1/2,R} = \bar{u}_{j+1} - \frac{1}{2}\overline{\Delta u}_{j+1}. \tag{71b}$$

Note that a DG method does not need to approximate the slope $\overline{\Delta u}_j(t)$ by using data from the neighboring cells since the slope is also stored as a variable in each cell. After inserting the difference form of fluxes, and some algebra, the spatial difference operator has the following form:

$$\mathbf{N}_{\mathrm{DG}(1)} = \boldsymbol{\mathcal{A}}^{+}\mathbf{D}^{+} + \boldsymbol{\mathcal{C}} + \boldsymbol{\mathcal{A}}^{-}\mathbf{D}^{-}, \tag{72}$$

where

$$\boldsymbol{\mathcal{A}}^{+} = \frac{q-r}{2\Delta x}\begin{pmatrix} 1 & -\dfrac{1}{2} \\ 6 & -3 \end{pmatrix}, \quad \boldsymbol{\mathcal{A}}^{-} = \frac{q+r}{2\Delta x}\begin{pmatrix} -1 & -\dfrac{1}{2} \\ 6 & 3 \end{pmatrix}, \quad \boldsymbol{\mathcal{C}} = \frac{r}{\Delta x}\begin{pmatrix} 0 & 0 \\ 0 & -\dfrac{6q}{r} \end{pmatrix}, \quad \mathbf{D}^{\pm} = \delta^{\pm}\mathbf{I}, \tag{73}$$

or, for a Fourier mode,

$$\mathbf{N}_{\mathrm{DG}(1)} = \boldsymbol{\mathcal{A}}^{+}e^{i\beta\mathbf{I}} + \boldsymbol{\mathcal{C}}' - \boldsymbol{\mathcal{A}}^{-}e^{i\beta\mathbf{I}} \quad \text{where} \quad \boldsymbol{\mathcal{C}}' = -\frac{r}{\Delta x}\begin{pmatrix} \dfrac{q}{r} & \dfrac{1}{2} \\ -6 & \dfrac{3q}{r} \end{pmatrix}. \tag{74}$$

Here, the notation "DG(1)" stands for a method representing the solution as piecewise polynomial of degree 1. The identical result can be obtained by setting $\nu = 0$ in the spatial-temporal operator for a DG–Hancock method (96)–(98). In order to obtain the eigenvalues of the spatial operator, the characteristic equation of $\mathbf{N}_{\mathrm{DG(1)}}$, given by

$$(\Delta x\lambda)^2 + \left[(q-r)\delta^+ - (q+r)\delta^- + 6q\right]\Delta x\lambda + 3q\left[(q-r)\delta^+ - (q+r)\delta^-\right] - 3(q^2-r^2)\delta^+\delta^- = 0, \qquad (75)$$

is solved for λ. Because of the lengthy expression of the roots in the general form, we present only the result for the upwind flux, $q = r$, as an example:

$$\lambda^{(1),(2)}_{\mathrm{DG(1),\ upwind}} = \frac{r}{\Delta x}\left(-3+\delta^- \pm \sqrt{9-12\delta^- + (\delta^-)^2}\right) \qquad (76a)$$

$$= \frac{r}{\Delta x}\left(-2-e^{-i\beta} \pm \sqrt{-2+10e^{-i\beta}+e^{-2i\beta}}\right). \qquad (76b)$$

Accuracy The asymptotic eigenvalues in the low-frequency limit are given by

$$\lambda^{(1)}_{\mathrm{DG(1)}} = -\frac{ir}{\Delta x}\beta - \frac{r^2}{72q\Delta x}\beta^4 + O\big(\beta^5\big), \qquad (77a)$$

$$\lambda^{(2)}_{\mathrm{DG(1)}} = -\frac{6q}{\Delta x} + \frac{3ir}{\Delta x}\beta + O\big(\beta^2\big), \qquad (77b)$$

where the former is the principal root and the latter is the extraneous one. Since a temporal discretization has not been considered yet, all errors appearing here are attributed solely to the spatial discretization. The order of accuracy in space is obtained by replacing β by the wave number k, then

$$\lambda^{(1)}_{\mathrm{DG(1)}} - \lambda_{\mathrm{exact}} = -\frac{r^2}{72q}\boxed{\Delta x^3}\,k^4 + O\big(k^5\big), \qquad (78a)$$

$$\lambda^{(2)}_{\mathrm{DG(1)}} - \lambda_{\mathrm{exact}} = -\frac{6q}{\Delta x} + O(k), \qquad (78b)$$

thus the principal root is third-order accurate in space, and the extraneous root is zeroth-order. Fortunately, the extraneous root damps quickly since the leading error, $-\frac{6q}{\Delta x}$, is a large negative real value.

To examine the overall accuracy, the time integration methods RK2 and RK3 (54) are employed. Then, the local truncation errors become

$$\mathrm{LTE}_{\mathrm{DG(1)RK2}} = -\frac{ir^3}{6}\boxed{\Delta t^2}\,k^3 - \frac{r}{72}\left(\frac{r}{q}\boxed{\Delta x^3} - 9r^3\Delta t^3\right)k^4 + O\big(k^5\big), \qquad (79a)$$

$$\mathrm{LTE}_{\mathrm{DG(1)RK3}} = -\frac{r}{72}\left(\frac{r}{q}\boxed{\Delta x^3} + 3r^3\boxed{\Delta t^3}\right)k^4 + O\big(k^5\big). \qquad (79b)$$

The first equation shows that the temporal discretization RK2 introduces a second-order error in the DG(1)–RK2 method, hence the DG(1)–RK2 method is third-order in space, yet second-order in time. Since the RK3 method is third-order accurate, the DG(1)–RK3 method is third-order in both space and in time.

Stability The stability domain of the DG(1)–RK2 method was first presented by Cockburn and Shu,[19] and they referred to the stability proof for a simpler case by Chavent and Cockburn.[20, 21] Here, we present the stability limit by plotting the modulus of the two amplification factors independently. The modulus of the accurate and inaccurate amplification factors, $|g^{(1),(2)}_{\mathrm{DG(1)RK2}}|$, computed with the upwind flux are shown in Figures 2(a) and 2(b) respectively. The figures show that the accurate amplification factor possesses larger stability domain ($\tilde{\nu}_{\max} = 0.468$) than the inaccurate amplification factor ($\tilde{\nu}_{\max} = 1/3$). Overall, the stability is constrained by the inaccurate amplification factor, so DG(1)–RK2 with the upwind flux is stable for $\tilde{\nu} \leq 1/3$.

Counterintuitive results are shown in Figures 2(c) and 2(d), where the Lax–Friedrichs flux is employed. The contour plots of the modulus show that neither the accurate nor inaccurate amplification factor is stable for any Courant number, even when $\tilde{\nu} = 0$. Thus, the DG(1)–RK2 with the Lax–Friedrichs flux is unconditionally unstable. This was originally found by Rider and Lowrie.[22] The same result is obtained for the DG(1)–RK3 method. This is somewhat surprising since the Lax–Friedrichs flux adopts the largest possible dissipation coefficient, $\dfrac{\Delta x}{\Delta t}$, among all q-fluxes to stabilize the method. A reason for the destabilizing result produced by this most dissipative flux function can be found by comparing the dominant numerical dissipation in (79) to that in (69). For a DG method, the dissipation parameter q appears in the denominator, whereas an HR method contains in the numerator. Thus, for a DG method, as the numerical dissipation in the flux increases, the method actually becomes less dissipative, at least at low frequencies. This is completely opposite to the behavior of an HR method. Hence, the most dissipative flux leads to the least low-frequency dissipation, resulting in an unconditionally unstable DG method. More specifically, the instability originates in the extraneous root, $\lambda^{(2)}_{\mathrm{DG}(1)}$. The leading error in the extraneous root, multiplied by the time step Δt (this product appears when a time integration method is applied), evaluated with the upwind and Lax–Friedrichs fluxes, reads:

$$\Delta t\, \lambda^{(2),\mathrm{LxF}}_{\mathrm{DG}(1)} = -\Delta t \frac{6q}{\Delta x} = -6, \tag{80a}$$

$$\Delta t\, \lambda^{(2),\mathrm{upwind}}_{\mathrm{DG}(1)} = -\Delta t \frac{6r}{\Delta x} = -6\tilde{\nu}. \tag{80b}$$

Assume, for instance, that the RK2 method is used for time integration, then the above eigenvalues should satisfy a necessary condition, $\Delta t\, \lambda \in [-2, 0]$, for stability in the low-frequency limit. The second equation, for the upwind flux, satisfies this stability condition as long as $\tilde{\nu} \leq \dfrac{1}{3}$. Conversely, the first equation never satisfies the stability condition, no matter how small the time step is; thus, DG(1) together with the Lax–Friedrichs flux is unconditionally unstable.

To remedy the instability of the Lax–Friedrichs flux, Rider and Lowrie propose the following modified Lax–Friedrichs flux:[22]

$$f^{\mathrm{mLxF}}_{j+1/2}(u_L, u_R) = \frac{r}{2}(u_L + u_R) - \frac{z}{2}\frac{\Delta x}{\Delta x}(u_R - u_L), \tag{81}$$

where $z = \dfrac{1}{3}$ for DG(1), and $z = \dfrac{1}{5}$ for DG(2). These constants are chosen such that the maximum stable Courant number is the same as for the DG method combined with the upwind flux. The motivation of the choice of constant becomes clear when the leading error is again considered:

$$\Delta t \lambda^{(2),\mathrm{mLxF}}_{\mathrm{DG}(1)} = -\Delta t \frac{6\, q_{\mathrm{mLxF}}}{\Delta x} = -6z, \tag{82}$$

thus as long as $z \leq \dfrac{1}{3}$, the leading error satisfies the stability condition, $\Delta t\, \lambda \in [-2, 0]$. Since the condition is merely necessary and not sufficient, the full stability domains based on the modified Lax–Friedrichs flux are obtained numerically and shown in Figures 2(e) and 2(f). For this flux function, both accurate and inaccurate eigenmodes possess the same stability limit, $\tilde{\nu} \leq 0.424$. This is less restrictive than DG(1)–RK2 with the upwind flux; however, it can be observed that the modified Lax–Friedrichs flux is more dissipative than the upwind flux, especially for high frequency modes.

III.C.3. HR–Hancock Method

The original Hancock method is a fully discrete one-step method.[6,7] Here, we denote the method as "HR–Hancock" or "HR–Ha." The update formula is slightly different from an HR–MOL method, and given by

$$\bar{u}_j^{n+1} = \bar{u}_j^n - \frac{\Delta t}{\Delta x}\left(\hat{f}_{j+1/2}^{n+1/2} - \hat{f}_{j-1/2}^{n+1/2}\right), \tag{83}$$

where the time level of the flux evaluation is already specified: $t = t^{n+1/2}$. Again, the q-flux (64) is adopted; however, the input values of the flux function are given by a Taylor series expansion in space and time, thus

$$u_{j+1/2,L}^{n+1/2} = \bar{u}_j^n + \frac{1}{2}\left(1 - r\frac{\Delta t}{\Delta x}\right)\overline{\Delta u}_j^n, \tag{84a}$$

$$u_{j+1/2,R}^{n+1/2} = \bar{u}_{j+1}^n - \frac{1}{2}\left(1 + r\frac{\Delta t}{\Delta x}\right)\overline{\Delta u}_{j+1}^n. \tag{84b}$$

As in the HR–MOL method, the slope $\overline{\Delta u_j}$ is obtained by the average of two slopes over cells $(j+1, j, j-1)$, hence

$$\frac{\overline{\Delta u}_j^n}{\Delta x} = \frac{1}{2}\left(\frac{\bar{u}_{j+1}^n - \bar{u}_j^n}{\Delta x} + \frac{\bar{u}_j^n - \bar{u}_{j-1}^n}{\Delta x}\right) = \frac{\bar{u}_{j+1}^n - \bar{u}_{j-1}^n}{2\Delta x}. \tag{85}$$

After inserting the difference form of fluxes, and some algebra, the spatial-temporal difference operator is given by

$$\begin{aligned} \mathrm{M}_{\mathrm{HR2Ha}} = &-\frac{1}{8\Delta x}\left[(q-r)(1+r\nu)(\delta^+)^2 + (q+r)(1-r\nu)(\delta^-)^2\right] \\ &+ \frac{1}{4\Delta x}\left[(q-2r+r^2\nu)\delta^+ - (q+2r+r^2\nu)\delta^-\right], \end{aligned} \tag{86a}$$

or, for a Fourier mode, using (57),

$$\begin{aligned} \mathrm{M}_{\mathrm{HR2Ha}} = &-\frac{1}{8\Delta x}\left[(q-r)(1+r\nu)e^{2i\beta} + (q+r)(1-r\nu)e^{-2i\beta}\right] \\ &- \frac{1}{4\Delta x}\left[(3r-2q-rq\nu)e^{i\beta} + (3q+r^2\nu) - (3r+2q-rq\nu)e^{-\beta}\right]. \end{aligned} \tag{86b}$$

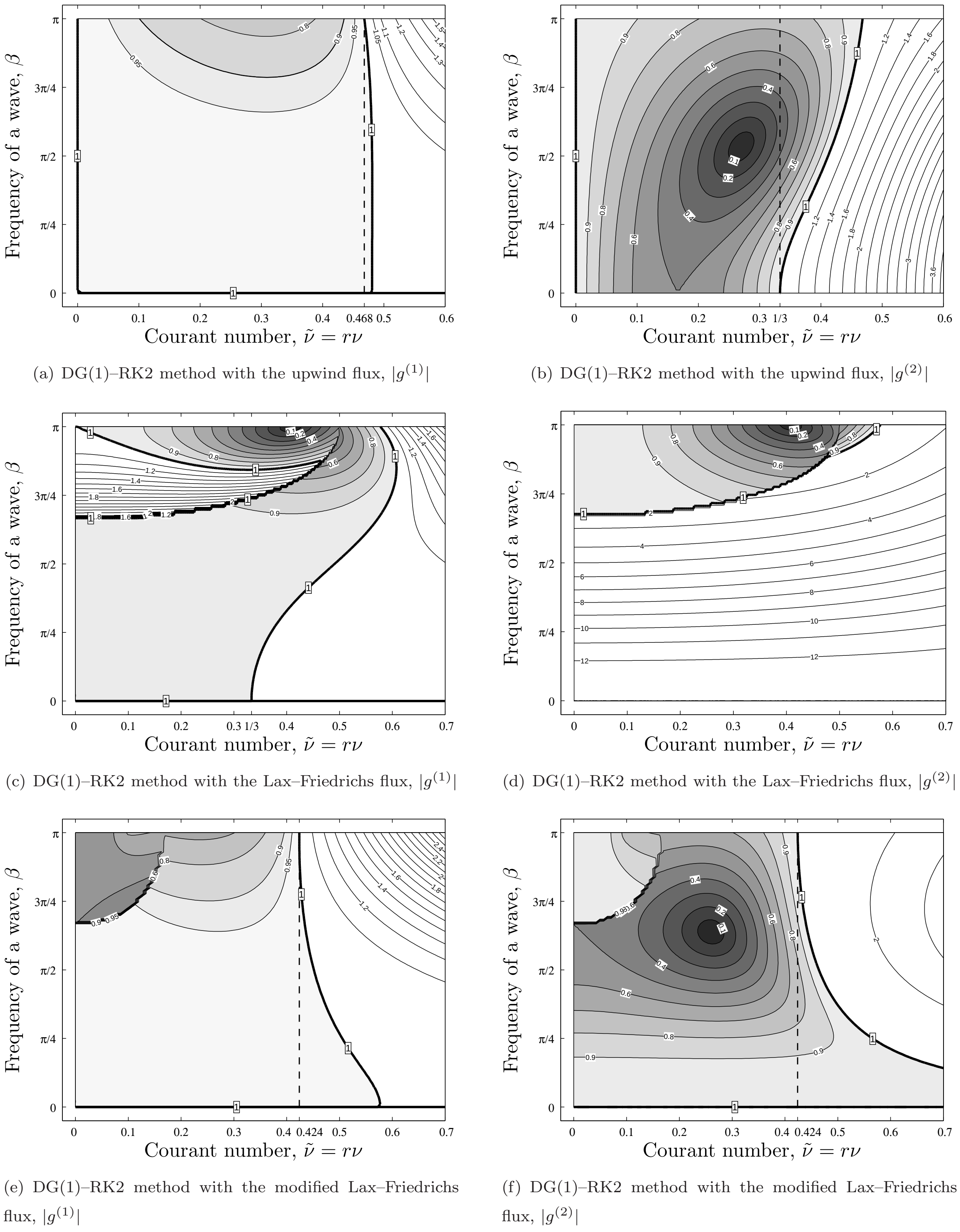
When the upwind flux, $q = r$, is used, and we set the Courant number equal to 1 ($r\nu = \tilde{\nu} = 1$), then the above operator reduces to

$$\Delta t\,\mathrm{M}_{\mathrm{HR2Ha}} = -\delta^-, \tag{87}$$

which is the exact upwind difference operator. Inserting the above equation into the original update scheme (83) leads to

$$\begin{aligned} \bar{u}_j^{n+1} &= \bar{u}_j^n - \delta^-\bar{u}_j^n \\ &= \bar{u}_{j-1}^n. \end{aligned} \tag{88}$$

Thus, the HR2–Hancock method produces the exact shift.



(a) DG(1)–RK2 method with the upwind flux, $|g^{(1)}|$

(b) DG(1)–RK2 method with the upwind flux, $|g^{(2)}|$

(c) DG(1)–RK2 method with the Lax–Friedrichs flux, $|g^{(1)}|$

(d) DG(1)–RK2 method with the Lax–Friedrichs flux, $|g^{(2)}|$

(e) DG(1)–RK2 method with the modified Lax–Friedrichs flux, $|g^{(1)}|$

(f) DG(1)–RK2 method with the modified Lax–Friedrichs flux, $|g^{(2)}|$

Figure 2. Contour plots of the modulus of the amplification factor, $|g_{\mathbf{DG(1)RK2}}(\tilde{\nu},\beta)|$, computed with the upwind (top), Lax–Friedrichs (middle), and modified Lax–Friedrichs (bottom) fluxes. The upwind flux leads to the stable method for $\tilde{\nu} \le 1/3$, whereas the Lax–Friedrichs flux results in the unconditionally unstable method. The modified Lax–Friedrichs flux remedies the instability, and stable for $\tilde{\nu} \le 0.424$.

Accuracy Replacing the difference operators by their Fourier symbols (57) and taking the low-frequency limit leads to the asymptotic eigenvalue

$$\lambda_{\mathrm{HR2Ha}} = -\frac{ir}{\Delta x}\beta - \frac{r^2\nu}{2\Delta x}\beta^2 - \left(\frac{ir}{12\Delta x} - \frac{iqr\nu}{4\Delta x}\right)\beta^3 - \left(\frac{q}{8\Delta x} - \frac{r^2\nu}{6\Delta x}\right)\beta^4 + O\big(\beta^5\big)\,. \tag{89}$$

The order of accuracy in space is obtained by letting $\nu \to 0$, and replacing β by the wave number k,

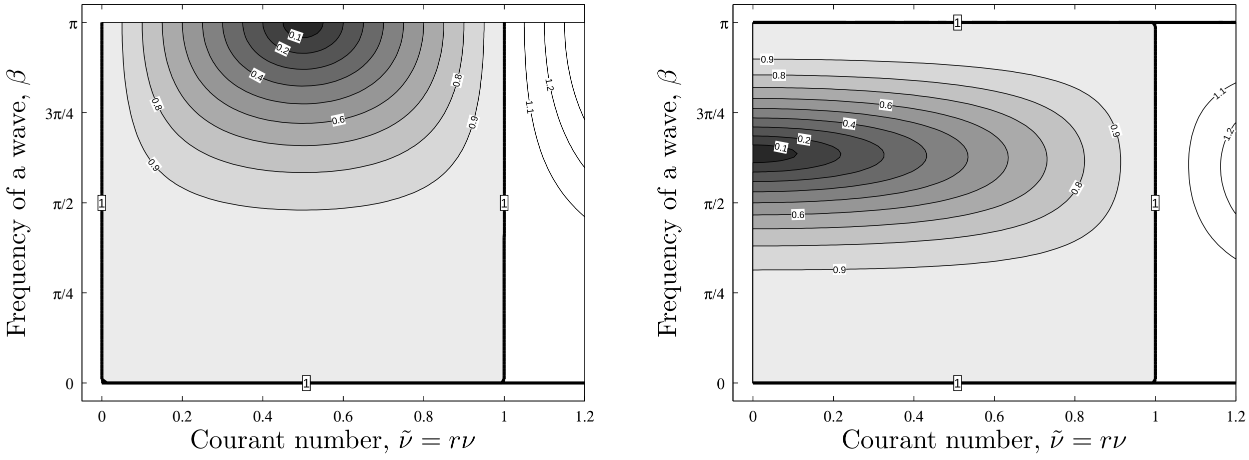
$$\lambda_{\mathrm{HR2Ha}} - \lambda_{\mathrm{exact}} = -\frac{ir}{12}\boxed{\Delta x^2}\,k^3 + O\big(k^4\big)\,, \tag{90}$$

thus the method is second-order accurate in space. To examine the overall order of accuracy, the amplification factor, $g_{\mathrm{HR2Ha}} = 1 + \Delta t\,\mathrm{M}_{\mathrm{HR2Ha}}$, is considered; the local truncation error is given by

$$\mathrm{LTE}_{\mathrm{HR2Ha}} = -\frac{ir}{12}\left(\boxed{\Delta x^2} - 3q\,\boxed{\Delta x\Delta t} + 2r^2\,\boxed{\Delta t^2}\right)k^3 - \frac{r}{8}\left[\frac{q}{r} - (r\nu)^3 + 2r\nu(q\nu-1)\right]\Delta x^3k^4 + O\big(k^5\big)\,. \tag{91}$$

Here, a new expression, $\Delta x\Delta t$, appears in the leading error term. Since the time step scales as the grid size, $\Delta t \propto \Delta x$, based on the CFL condition, this term is second-order error. Thus, the HR2–Hancock method is second-order in space and time.

Stability The modulus of the amplification factor, $|g_{\mathrm{HR2Ha}}(\tilde{\nu},\beta)|$, evaluated with the upwind and Lax–Friedrichs fluxes are shown in Figures 3(a) and 3(b) respectively. The shaded area indicates the stability region, where $|g_{\mathrm{HR2Ha}}(\tilde{\nu},\beta)| \le 1$. These two figures show that the HR2–Hancock method combined with both upwind and Lax–Friedrichs fluxes is linearly stable for $\tilde{\nu} \le 1$. Compared to the HR2–RK2 method shown in Figures 1(a) and 1(b), the HR2–Hancock is less dissipative, and also possesses the shift condition: $|g_{\mathrm{HR2Ha}}(1,\beta)| = 1$ for any β.



(a) HR–Hancock method with the upwind flux

(b) HR–Hancock method with the Lax–Friedrichs flux

Figure 3. Contour plots of the modulus of the amplification factor, $|g_{\mathrm{HR2Ha}}(\tilde{\nu},\beta)|$, computed with the upwind (left) and Lax–Friedrichs (right) fluxes. These show that the HR2–Hancock method is stable for $\tilde{\nu} \le 1$.

III.C.4. DG–Hancock Method

A DG–Hancock method is also fully discrete, and introduces two variables for a scalar equation, yielding a 2×2 amplification matrix. The update formulas for cell-average and undivided gradient have the following form:

$$\bar{u}_j^{n+1} = \bar{u}_j^n - \frac{\Delta t}{\Delta x}\left(\hat{f}_{j+1/2}^{n+1/2} - \hat{f}_{j-1/2}^{n+1/2}\right), \tag{92a}$$

$$\overline{\Delta u}_j^{n+1} = \overline{\Delta u}_j^n - \frac{\Delta t}{\Delta x} 6\left(\hat{f}_{j+1/2}^{n+1/2} + \hat{f}_{j-1/2}^{n+1/2} - 2r\check{u}_j\right), \tag{92b}$$

where the interface fluxes are evaluated at the time level, $t^{n+1/2}$, by the q-flux (64). The input values for the flux function at the time level $t^{n+k/2}$ are given by

$$u_{j+1/2,L}^{n+k/2} = \bar{u}_j^n + \frac{1}{2}\left(1 - r\frac{k\Delta t}{\Delta x}\right)\overline{\Delta u}_j^n, \tag{93a}$$

$$u_{j+1/2,R}^{n+k/2} = \bar{u}_{j+1}^n - \frac{1}{2}\left(1 + r\frac{k\Delta t}{\Delta x}\right)\overline{\Delta u}_{j+1}^n. \tag{93b}$$

The volume integral of the flux over the domain $[x_{j-1/2}, x_{j+1/2}] \times [t^n, t^{n+1}]$ simplifies owing to the linear flux. Hence, the spatial integration is done exactly, and a quadrature is only required in time. Two techniques are employed: 3-point Gauss–Lobatto and 2-point Gauss–Radau quadratures. These quadratures, with their necessary intermediate update equation, are as follows:

3-point Gauss–Lobatto

$$\check{u}_{j,\mathrm{GL}} = \frac{1}{6}(\bar{u}_j^n + 4\bar{u}_j^{n+1/2} + \bar{u}_j^{n+1}), \quad \text{where} \quad \bar{u}_j^{n+1/2} = \bar{u}_j^n - \frac{\Delta t}{2}\frac{1}{\Delta x}\left(\hat{f}_{j+1/2}^{n+1/4} - \hat{f}_{j-1/2}^{n+1/4}\right), \tag{94}$$

2-point Gauss–Radau

$$\check{u}_{j,\mathrm{GR}} = \frac{1}{4}(3\bar{u}_j^{n+1/3} + \bar{u}_j^{n+1}), \quad \text{where} \quad \bar{u}_j^{n+1/3} = \bar{u}_j^n - \frac{\Delta t}{3}\frac{1}{\Delta x}\left(\hat{f}_{j+1/2}^{n+1/6} - \hat{f}_{j-1/2}^{n+1/6}\right). \tag{95}$$

Here, the cell-interface fluxes, $\hat{f}_{j\pm1/2}^{n+1/6}$ and $\hat{f}_{j\pm1/2}^{n+1/4}$, are again obtained by the q-flux, with the input values (93) with $k = \frac{1}{6}, \frac{1}{4}$ respectively. Even though the two quadratures use different points, owing to the linear flux both Gauss–Lobatto and Gauss–Radau lead to the identical volume integral of the flux, hence $\hat{u}_{j,\mathrm{GL}} \equiv \hat{u}_{j,\mathrm{GR}}$.

After inserting the flux formula into the update scheme, and some algebra, a space-time difference operator in the form of (51) results; using the notation $\mathbf{u}_j^n = [\bar{u}_j^n, \overline{\Delta u}_j^n]^T$ it can be written in the form

$$\mathbf{M}_{\mathrm{DG(1)Ha}} = \boldsymbol{\mathcal{A}}^+\mathbf{D}^+ + \boldsymbol{\mathcal{C}} + \boldsymbol{\mathcal{A}}^-\mathbf{D}^-, \tag{96}$$

where

$$\boldsymbol{\mathcal{A}}^+ = \frac{q-r}{2\Delta x}\begin{pmatrix} 1 & -\frac{1}{2}(1+r\nu) \\ 6(1+r\nu) & -3-6r\nu-2(r\nu)^2 \end{pmatrix}, \quad \boldsymbol{\mathcal{A}}^- = \frac{q+r}{2\Delta x}\begin{pmatrix} -1 & -\frac{1}{2}(1-r\nu) \\ 6(1-r\nu) & 3-6r\nu+2(r\nu)^2 \end{pmatrix}, \tag{97a}$$

$$\boldsymbol{\mathcal{C}} = \frac{r}{\Delta x}\begin{pmatrix} 0 & 0 \\ 0 & -6\left(\frac{q}{r} - r\nu\right) \end{pmatrix}, \qquad \mathbf{D}^{\pm} = \delta^{\pm}\mathbf{I}, \tag{97b}$$

or, when applied to a Fourier mode,

$$\mathbf{M}_{\mathrm{DG(1)Ha}} = \boldsymbol{\mathcal{A}}^{+} e^{i\beta\mathbf{I}} + \boldsymbol{\mathcal{C}}' - \boldsymbol{\mathcal{A}}^{-} e^{-i\beta\mathbf{I}} \quad \text{where} \quad \boldsymbol{\mathcal{C}}' = -\frac{r}{\Delta x}\begin{pmatrix} \dfrac{q}{r} & \dfrac{1}{2}(1-q\nu) \\ -6(1-q\nu) & \dfrac{3q}{r} - 2rq\nu^2 \end{pmatrix}. \tag{98}$$

When the upwind flux $q = r$ is used, and we set the Courant number equal to 1 ($r\nu = \tilde{\nu} = 1$), the above operator reduces to

$$\Delta t\, \mathbf{M}_{\mathrm{DG(1)Ha}} = -\delta^{-}\mathbf{I}, \tag{99}$$

which is again the exact upwind difference operator. Combined with the forward Euler time integrator (52), the method reduces to the exact solution $\mathbf{u}_j^{n+1} = \mathbf{u}_{j-1}^{n}$. Thus the DG(1)–Hancock method produces the exact shift.

As mentioned earlier, even though a scalar equation is considered, a DG method produces a difference operator in matrix form. Thus, the characteristic equation of $\mathbf{M}_{\mathrm{DG(1)Ha}}$ is a quadratic form:

$$\begin{aligned}(\Delta x\lambda)^2 + \big[(q-r)\big((r\nu)^2 + 3r\nu + 1\big)\delta^{+} - (q+r)\big((r\nu)^2 - 3r\nu + 1\big)\delta^{-} + 6(q - r^2\nu)\big]\,\Delta x\lambda \\ - \frac{1}{4}(r\nu)^2\left[(q-r)^2(\delta^{+})^2 + (q+r)^2(\delta^{-})^2\right] + 3(q - r^2\nu)\left[(q-r)\delta^{+} - (q+r)\delta^{-}\right] \\ - \frac{1}{2}(q^2 - r^2)\left[6 - (r\nu)^2\right]\delta^{+}\delta^{-} = 0,\end{aligned} \tag{100}$$

which provides two eigenvalues; principal and extraneous. Since the general forms of eigenvalues are lengthy, only the eigenvalues for the upwind flux, $q = r$, are presented here as an example:

$$\begin{aligned}\lambda^{(1,2)}_{\mathrm{DG(1)Ha,\ upwind}} = \frac{r}{\Delta x}\left[1 - 3r\nu + (r\nu)^2\right]\delta^{-} \\ - \frac{r}{\Delta x}(1 - r\nu)\left[3 \mp \sqrt{9 - 6(2 - r\nu)\delta^{-} + \big(1 - 4r\nu + (r\nu)^2\big)(\delta^{-})^2}\right].\end{aligned} \tag{101}$$

The asymptotic analysis that follows, though, is based on the general q-flux.

Accuracy Replacing the difference operators by their Fourier symbols (57), and taking the low-frequency limit, leads to

$$\lambda^{(1)}_{\mathrm{DG(1)Ha}} = -\frac{ir}{\Delta x}\beta - \frac{r^2\nu}{2\Delta x}\beta^2 + \frac{ir^3\nu^2}{6\Delta x}\beta^3 - \frac{r}{72\Delta x}\left[\frac{r}{q}\frac{\big(1 - (r\nu)^2\big)^2}{1 - r^2\nu/q} - 3r\nu\big(1 - q\nu + (r\nu)^2\big)\right]\beta^4 + O\big(\beta^5\big), \tag{102a}$$

$$\lambda^{(2)}_{\mathrm{DG(1)Ha}} = -\frac{6r}{\Delta x}\left(\frac{q}{r} - r\nu\right) + \frac{ir\left[3 - 6q\nu + 2(r\nu)^2\right]}{\Delta x}\beta + O\big(\beta^2\big). \tag{102b}$$

When comparing with the exact eigenvalue (62), it is clear that $\lambda^{(1)}_{\mathrm{DG(1)Ha}}$ is the principal root and $\lambda^{(2)}_{\mathrm{DG(1)Ha}}$ is the extraneous one. Based on the range of the dissipation parameter, $r \le q \le \dfrac{\Delta x}{\Delta t}$, it is easily shown that $\dfrac{q}{r} - r\nu \ge 0$. Thus, the leading term independent of β in $\lambda^{(2)}_{\mathrm{DG(1)Ha}}$ is a negative real value, and corresponding extraneous wave is damped quickly. The order of accuracy in space is obtained by letting $\nu \longrightarrow 0$,

$$\lambda^{(1)}_{\mathrm{DG(1)Ha}} - \lambda_{\mathrm{exact}} = -\frac{r^2}{72q}\boxed{\Delta x^3}\,k^4 + O\big(k^5\big), \tag{103a}$$

$$\lambda^{(2)}_{\mathrm{DG(1)Ha}} - \lambda_{\mathrm{exact}} = -\frac{6q}{\Delta x} + O(k), \tag{103b}$$

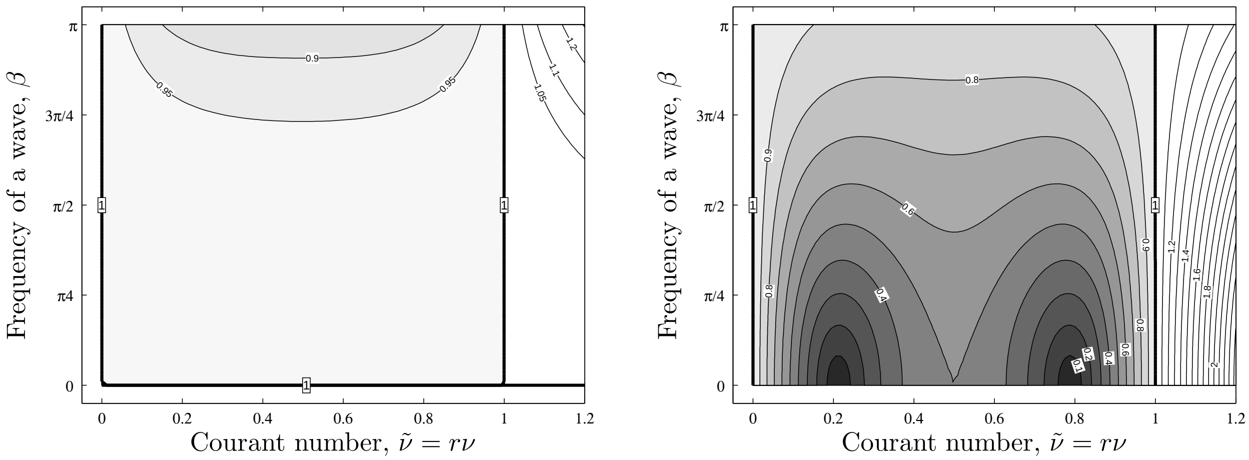
thus the principal root is third-order accurate in space and the extraneous root is zeroth-order.

The overall accuracy is derived by the eigenvalues of the amplification matrix of a fully discrete form, $\mathbf{G}_{\mathrm{DG(1)Ha}} = \mathbf{I} + \Delta t\, \mathbf{M}_{\mathrm{DG(1)Ha}}$. Following the same procedure as before, the local truncation error of an accurate mode is given by

$$\mathrm{LTE}^{(1)}_{\mathrm{DG(1)Ha}} = -\frac{r}{72}\left[\frac{r}{q}\frac{\left(1-(r\nu)^2\right)^2}{1-r^2\nu/q} - 3r\nu(1-q\nu)\right]\boxed{\Delta x^3}\; k^4 + O\big(k^5\big)\,. \tag{104}$$

Therefore, the DG(1)–Hancock method is third-order in space and time.

Stability The modulus of the accurate and inaccurate amplification factors, $|g^{(1),(2)}_{\mathrm{DG(1)Ha}}|$, computed with the upwind fluxes are shown in Figure 4. Compared to the DG(1)–RK2 method illustrated in Figures 2(a) and 2(b), the DG(1)–Hancock method possesses a wider stability region, $\tilde{\nu} \leq 1$, and also is less dissipative at high frequencies. When the Lax–Friedrichs flux is employed, as for DG(1)–RK2, the DG(1)–Hancock method becomes unconditionally unstable.



(a) DG(1)–Hancock method with the upwind flux

(b) DG(1)–Hancock method with the upwind flux

Figure 4. Contour plots of the modulus of the amplification factor, $|g_{\mathbf{DG(1)Ha}}(\tilde{\nu},\beta)|$, computed with the upwind flux. These show that the DG(1)–Hancock method with the upwind flux is stable for $\tilde{\nu} \leq 1$.

III.C.5. Dominant Dispersion/Dissipation Error in 1-D

The results of a Fourier analysis for each method are listed below for comparison. Even though our main focus in this section is the analysis and comparison of the DG–Hancock method to the HR/DG–MOL methods, we also present results for two newly developed methods: the spectral finite volume (SV) method[23] and the arbitrary high order schemes using derivatives (DG–ADER method).[24] The details of the analysis for these methods are omitted here. The local truncation errors show the dominant dispersion error, $O\big(k^3\big)$-term, and the dissipation error, $O\big(k^4\big)$-term. Moreover, owing to the equivalence of the Fourier analysis and the modified-equation analysis, the leading error term indicates the order of accuracy.

semi-discrete methods:

$$\mathrm{LTE}_{\mathrm{HR2RK2}} = c_3 \left[1 + 2(r\nu)^2\right] k^3 \quad + \quad c_4 \left[\frac{q}{r} - (r\nu)^3\right] k^4, \tag{105a}$$

$$\mathrm{LTE}_{\mathrm{HR2RK3}} = c_3\ k^3 \quad + \quad c_4 \left[\frac{q}{r} + \frac{1}{3}(r\nu)^3\right] k^4, \tag{105b}$$

$$\mathrm{LTE}_{\mathrm{HR3RK3}} = \quad + \frac{2}{3} c_4 \left[\frac{q}{r} + \frac{1}{2}(r\nu)^3\right] k^4, \tag{105c}$$

$$\mathrm{LTE}^{(1)}_{\mathrm{DG(1)RK2}} = c_3 \left[2(r\nu)^2\right] k^3 \quad + \frac{1}{9} c_4 \left[\frac{r}{q} - 9(r\nu)^3\right] k^4, \tag{105d}$$

$$\mathrm{LTE}^{(1)}_{\mathrm{DG(1)RK3}} = \quad \frac{1}{9} c_4 \left[\frac{r}{q} + 3(r\nu)^3\right] k^4, \tag{105e}$$

$$\mathrm{LTE}^{(1),\mathrm{upwind}}_{\mathrm{DG(2)RK3}} = \quad \frac{1}{9} c_4 \left[\quad 3(r\nu)^3\right] k^4, \tag{105f}$$

$$\mathrm{LTE}^{(1)}_{\mathrm{SV2RK2}} = -\frac{1}{2} c_3 \left[1 - 4(r\nu)^3\right] k^3 \quad + \frac{2}{9} c_4 \left[\frac{r}{q} - 4(r\nu)^3\right] k^4, \tag{105g}$$

fully discrete methods:

$$\mathrm{LTE}_{\mathrm{HR2Ha}} = c_3 \left[1 - 3q\nu + 2(r\nu)^2\right] k^3 + \quad c_4 \left[\frac{q}{r} - (r\nu)^3 + 2r\nu(q\nu - 1)\right] k^4, \tag{105h}$$

$$\mathrm{LTE}^{(1)}_{\mathrm{DG(1)Ha}} = \quad \frac{1}{9} c_4 \left[\frac{r}{q} \frac{\left(1 - (r\nu)^2\right)^2}{1 - r^2\nu/q} - 3r\nu(1 - q\nu)\right] k^4, \tag{105i}$$

$$\mathrm{LTE}^{(1)}_{\mathrm{DG(1)ADER}} = c_3 \left[r\nu \left(\frac{r}{q} - r\nu\right)\right] k^3 \quad + \frac{1}{9} c_4 \left[\frac{r}{q} \left(1 - 4\frac{r}{q}(r\nu) + 3(r\nu)^2\right)\right] k^4, \tag{105j}$$

$$\mathrm{LTE}^{(1),\mathrm{upwind}}_{\mathrm{DG(2)ADER}} = \quad \frac{1}{15} c_4 \left[r\nu(1 - r\nu)\right] k^4, \tag{105k}$$

where

$$c_3 = -\frac{ir}{12} \boxed{\Delta x^2}, \quad c_4 = -\frac{r}{8} \boxed{\Delta x^3}. \tag{106}$$

The above equations show that the leading errors of the HR3–RK3, DG(1)–RK3, DG(2)–RK3, DG(1)–Hancock, and DG(2)–ADER methods are $O(\Delta x^3)$, whereas in the rest of methods they are $O(\Delta x^2)$. Interestingly, the DG(1) spatial discretization can yield a third-order method, at least for a linear equation discretized on a uniform grid, if a proper time integration method is adopted. The Hancock and RK3 method lead to a third-order method; note that, DG(1)–RK3 requires three flux calculations at each cell-interface whereas DG(1)–Hancock requires two to achieve the same order. The DG(2) spatial discretization together with the same (third) order of temporal discretization provides a third-order method: DG(2)–RK3 and DG(2)–ADER. In these methods, the leading error can be attributed to the temporal discretization, as can be seen by letting $\tilde{\nu} = r\nu \to 0:$ the $O(k^4)$-term disappears. The analysis also shows the lower dissipative of DG discretizations: the leading-error coefficient of a DG(1) method is $\frac{1}{9}$ of the value for HR2, $\frac{1}{6}$ of the value for HR3, and $\frac{1}{2}$ of the value for SV2.

The stability limit of the methods when combined with the upwind and Lax–Friedrichs fluxes are shown in Table 3. In an HR (finite-volume) method, the linear stability limit increases as the order of method increases due to the inclusion of wider stencils. Conversely, DG and SV methods reduce their stability domain while increasing the order of accuracy, because increasing the number of unknowns per cell is equivalent to a grid refinement. Among the DG(1)–Hancock, DG(1)–RK3, and DG(2)–RK3, all third-order accurate, yet DG(1)–Hancock possesses the largest stability domain, $\tilde{\nu}_{\max} = 1.0$. Nevertheless, DG(1)–Hancock requires only two Riemann solvers per cell-interface per time-step, whereas both DG(1)–RK3 and DG(2)–RK3 need three Riemann solvers.

Table 3. The maximum stable Courant number, $\tilde{\nu}_{\max} := r\dfrac{\Delta t}{\Delta x}$, of a method applied to the 1-D linear advection equation is tabulated. The DG(1)–Hancock method is seen to possess the largest stability domain among all DG discretizations listed here.

	method	order p	maximum Courant number, $\tilde{\nu}_{\max}$	
			upwind: $q = r$	Lax–Friedrichs: $q = \Delta x/\Delta t$
semi-discrete	HR1–RK1	1	1.0	1.0
	HR2–RK2	2	1.0	1.0
	HR2–RK3	2	1.175	1.499
	HR3–RK3	3	1.625	1.681
	DG(1)–RK2	2	0.333	unstable
	DG(1)–RK3	3	0.409	unstable
	DG(2)–RK3	3	0.209	unstable
	SV2–RK2	2	0.500	unstable
	SV3–RK3	3	0.333	(unconfirmed)
fully discrete	HR2–Hancock	2	1.0	1.0
	DG(1)–Hancock	3	1.0	unstable
	DG(1)–ADER	2	0.333	unstable
	DG(2)–ADER	3	0.170	unstable

III.D. Difference Operators and Their Properties in 2-D

In this section, a several discretization methods are applied to the two-dimensional linear advection equation (44b), and a Fourier analysis is employed to uncover their dominant dissipation and dispersion error, order of accuracy, and domain of linear stability. Four methods, HR2–RK2, HR–Hancock, DG(1)–RK2, and DG(1)–RK3, are compared to the DG(1)–Hancock method. For brevity, the detailed analysis is only presented for the DG(1)–Hancock method. The exact amplification factor and exact eigenvalue of 2-D spatial differentiation can be expressed as

$$g_{\text{exact}}(\tilde{\nu}_x, \tilde{\nu}_y, \alpha, \beta) = e^{-i(r\nu_x\alpha + s\nu_y\beta)} = e^{-i(\tilde{\nu}_x\alpha + \tilde{\nu}_y\beta)}, \tag{107a}$$

$$\lambda_{\text{exact}} = -i\left(\frac{r}{\Delta x}\alpha + \frac{s}{\Delta y}\beta\right), \tag{107b}$$

where α and β are spatial frequencies in the x- and y-directions, respectively.

III.D.1. DG–Hancock Method

The DG–Hancock method on rectangular grids has the following update formulas:

$$\begin{aligned}
\bar{u}_{i,j}^{n+1} &= \bar{u}_{i,j}^{n} - \frac{\Delta t}{\Delta x}\left(\hat{f}_{i+1/2,j}^{n+1/2} - \hat{f}_{i-1/2,j}^{n+1/2}\right) - \frac{\Delta t}{\Delta y}\left(\hat{g}_{i+1/2,j}^{n+1/2} - \hat{g}_{i-1/2,j}^{n+1/2}\right), && \text{(108a)}\\
\overline{\Delta_x u}_{i,j}^{n+1} &= \overline{\Delta_x u}_{i,j}^{n} - \frac{\Delta t}{\Delta x}6\left(\hat{f}_{i+1/2,j}^{n+1/2} + \hat{f}_{i-1/2,j}^{n+1/2} - 2r\check{u}_{i,j}\right)\\
&\quad - \frac{\Delta t}{\Delta y}12\left[\int_0^1\left(\xi-\frac{1}{2}\right)\hat{g}_{i,j+1/2}^{n+1/2}(\xi)\;d\xi - \int_0^1\left(\xi-\frac{1}{2}\right)\hat{g}_{i,j-1/2}^{n+1/2}(\xi)\;d\xi\right], && \text{(108b)}\\
\overline{\Delta_y u}_{i,j}^{n+1} &= \overline{\Delta_y u}_{i,j}^{n} - \frac{\Delta t}{\Delta y}6\left(\hat{g}_{i,j+1/2}^{n+1/2} + \hat{g}_{i,j-1/2}^{n+1/2} - 2s\check{u}_{i,j}\right)\\
&\quad - \frac{\Delta t}{\Delta x}12\left[\int_0^1\left(\eta-\frac{1}{2}\right)\hat{f}_{i+1/2,j}^{n+1/2}(\eta)\;d\eta - \int_0^1\left(\eta-\frac{1}{2}\right)\hat{f}_{i-1/2,j}^{n+1/2}(\eta)\;d\eta\right], && \text{(108c)}
\end{aligned}$$

where the fluxes in both coordinate directions are given by the q-flux (64). The volume integral of the flux simplifies owing to flux linearity, and quadrature is only required in time. Again, both Gauss–Lobatto and Gauss–Radau quadratures in time for $\check{u}_{i,j}$ lead to identical final update formulas. After inserting the difference form of the volume integral of the fluxes and the q-flux, the difference operator has the form

$$\mathbf{u}_j^{n+1} = \left(\mathbf{I} + \Delta t\,\mathbf{M}_{\mathrm{DG(1)Ha}}\right)\mathbf{u}_j^{n}, \tag{109}$$

where

$$\mathbf{M}_{\mathrm{DG(1)Ha}} = \boldsymbol{\mathcal{A}}^{+}\mathbf{D}_x^{+} + \boldsymbol{\mathcal{A}}^{-}\mathbf{D}_x^{-} + \boldsymbol{\mathcal{B}}^{+}\mathbf{D}_y^{+} + \boldsymbol{\mathcal{B}}^{-}\mathbf{D}_y^{-} + \boldsymbol{\mathcal{C}} \tag{110}$$

with

$$\mathbf{D}_x^{\pm} = \delta_x^{\pm}\mathbf{I}, \quad \mathbf{D}_y^{\pm} = \delta_y^{\pm}\mathbf{I}, \tag{111}$$

and the coefficient matrices are given by

$$\boldsymbol{\mathcal{A}}^{+} = \frac{q_x - r}{2\Delta x}\begin{pmatrix} 1 & -\frac{1}{2}(1+r\nu_x) & -\frac{s\nu_y}{2} \\ 6(1+r\nu_x) & -3-6r\nu_x-2(r\nu_x)^2 & -(3+2r\nu_x)s\nu_y \\ 6s\nu_y & -(3+2r\nu_x)s\nu_y & 1-2(s\nu_y)^2 \end{pmatrix}, \tag{112a}$$

$$\boldsymbol{\mathcal{A}}^{-} = \frac{q_x + r}{2\Delta x}\begin{pmatrix} -1 & -\frac{1}{2}(1-r\nu_x) & \frac{s\nu_y}{2} \\ 6(1-r\nu_x) & 3-6r\nu_x+2(r\nu_x)^2 & -(3-2r\nu_x)s\nu_y \\ -6s\nu_y & -(3-2r\nu_x)s\nu_y & -1+2(s\nu_y)^2 \end{pmatrix}, \tag{112b}$$

$$\boldsymbol{\mathcal{B}}^{+} = \frac{q_y - s}{2\Delta y}\begin{pmatrix} 1 & -\frac{r\nu_x}{2} & -\frac{1}{2}(1+s\nu_y) \\ 6r\nu_x & 1-2(r\nu_x)^2 & -(3+2s\nu_y)r\nu_x \\ 6(1+s\nu_y) & -(3+2s\nu_y)r\nu_x & -3-6s\nu_y-2(s\nu_y)^2 \end{pmatrix}, \tag{112c}$$

$$\boldsymbol{\mathcal{B}}^{-} = \frac{q_y + s}{2\Delta y}\begin{pmatrix} -1 & \frac{r\nu_x}{2} & -\frac{1}{2}(1-s\nu_y) \\ -6r\nu_x & -1+2(r\nu_x)^2 & -(3-2s\nu_y)r\nu_x \\ 6(1-s\nu_y) & -(3-2s\nu_y)r\nu_x & 3-6s\nu_y+2(s\nu_y)^2 \end{pmatrix}, \tag{112d}$$

$$\boldsymbol{\mathcal{C}} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & -\dfrac{6r}{\Delta x}\left(\dfrac{q_x}{r} - r\nu_x\right) & \dfrac{6s}{\Delta y} r\nu_x \\ 0 & \dfrac{6r}{\Delta x} s\nu_y & -\dfrac{6s}{\Delta y}\left(\dfrac{q_y}{s} - s\nu_y\right) \end{pmatrix}. \tag{112e}$$

Accuracy In view of the lengthy formula, let us assume the wave frequencies in the x- and y-directions are the same, thus $\alpha = \beta$. Furthermore, in the $O(\beta^4)$-term, a square mesh, $\Delta x = \Delta y = \Delta h$, is assumed. Under these assumptions, the asymptotic eigenvalues based on the upwind flux, $(q_x, q_y) = (r, s)$, become

$$\begin{aligned} \lambda^{(1)}_{\mathrm{DG(1)Ha}} &= -i\left(\frac{r}{\Delta x} + \frac{s}{\Delta y}\right)\beta - \frac{1}{2}\left(\frac{r}{\Delta x} + \frac{s}{\Delta y}\right)^2 \Delta t \beta^2 + \frac{i}{6}\left(\frac{r}{\Delta x} + \frac{s}{\Delta y}\right)^3 \Delta t^2 \beta^3 \\ &\quad - \frac{r+s}{72\Delta h}\left[4 - 5(r+s)\frac{\Delta t}{\Delta h} + 2(r+s)^2\left(\frac{\Delta t}{\Delta h}\right)^2 - 4(r+s)^3\left(\frac{\Delta t}{\Delta h}\right)^3\right]\beta^4 + O(\beta^5)\,, \end{aligned} \tag{113a}$$

$$\lambda^{(2),(3)}_{\mathrm{DG(1)Ha}} = -\frac{3}{\Delta h}\left[(r+s) - \nu(r^2+s^2) \pm \sqrt{(r-s)^2\big(1 - 2(r+s)\nu\big) + (r^2+s^2)^2\nu^2}\right] + O(\beta)\,. \tag{113b}$$

Replacing the wave frequency by the wave number, $k = \dfrac{\beta}{\Delta h}$, and letting $\Delta t \to 0$ brings out the spatial order of accuracy:

$$\lambda^{(1),\mathrm{upwind}}_{\mathrm{DG(1)Ha}} - \lambda_{\mathrm{exact}} = -\frac{r+s}{18}\boxed{\Delta h^3}\,k^4 + O(k^5)\,, \tag{114a}$$

$$\lambda^{(2),\mathrm{upwind}}_{\mathrm{DG(1)Ha}} - \lambda_{\mathrm{exact}} = -\frac{6r}{\Delta h} + O(k)\,, \tag{114b}$$

$$\lambda^{(3),\mathrm{upwind}}_{\mathrm{DG(1)Ha}} - \lambda_{\mathrm{exact}} = -\frac{6s}{\Delta h} + O(k)\,, \tag{114c}$$

thus, the spatial discretization is third-order accurate. Comparing the dominant dissipation error (114a) to the error obtained in the one-dimensional case, (103a), the multi-dimensionality increases the dissipation by a factor 4. This multi-dimensional error originates with the line integral of the flux along the cell interfaces, the last term in (108b) and (108c). When the Lax–Friedrichs flux $\left(q_x = q_y = q = \dfrac{\Delta h}{\Delta t} \text{ in the } O(\beta^4) \text{ term}\right)$ is adopted, the asymptotic eigenvalues become

$$\begin{aligned} \lambda^{(1),\mathrm{LxF}}_{\mathrm{DG(1)}} &= -i\left(\frac{r}{\Delta x} + \frac{s}{\Delta y}\right)\beta - \frac{1}{2}\left(\frac{r}{\Delta x} + \frac{s}{\Delta y}\right)^2 \Delta t\beta^2 + \frac{i}{6}\left(\frac{r}{\Delta x} + \frac{s}{\Delta y}\right)^3 \Delta t^2\beta^3 \\ &\quad - \frac{r+s}{72[(r^2+s^2)\Delta t - q\Delta h]}\left[(6q^2 + r^2 + s^2) - 12q(r^2 + rs + s^2)\left(\frac{\Delta t}{\Delta h}\right)\right. \\ &\quad + 2(r+s)^2\big(3q^2 + 2(r^2+s^2)\big)\left(\frac{\Delta t}{\Delta h}\right)^2 - 3q(r+s)^2(3r^2 + 2rs + 3s^2)\left(\frac{\Delta t}{\Delta h}\right)^3 \\ &\quad \left. + 4(r+s)^4(r^2+s^2)\left(\frac{\Delta t}{\Delta h}\right)^4\right]\beta^4 + O(\beta^5)\,, \end{aligned} \tag{115a}$$

$$\lambda^{(2),\mathrm{LxF}}_{\mathrm{DG(1)Ha}} = -\frac{6q}{\Delta h} + O(\beta)\,, \tag{115b}$$

$$\lambda^{(3),\mathrm{LxF}}_{\mathrm{DG(1)Ha}} = -\frac{6}{\Delta h}\left[q - (r^2+s^2)\nu\right] + O(\beta)\,. \tag{115c}$$

As in the one-dimensional case, the Lax–Friedrichs flux, $q = \dfrac{\Delta h}{\Delta t}$, leads to the constant leading error $-\dfrac{6q}{\Delta h} = -6$ in $\lambda^{(2),\mathrm{LxF}}_{\mathrm{DG(1)Ha}}$, thus the method becomes unconditionally unstable.

The overall order of accuracy for the scheme with the upwind flux becomes

$$\mathrm{LTE}^{(1),\mathrm{upwind}}_{\mathrm{DG(1)Ha}} = -\frac{r+s}{72}\left[4-5(r+s)\nu+2(r+s)^2\nu^2-(r+s)^3\nu^3\right]\boxed{\Delta h^3}\,k^4+O\big(k^5\big)\,, \tag{116}$$

thus the accurate eigenmode of the two-dimensional DG(1)–Hancock method is third-order in space and time.

Stability The unity contour of the amplification-factor modulus $|g_{\mathrm{method}}|$ is shown in Figure 5 for various upwind schemes. The shaded area indicates the region where $|g_{\mathrm{method}}(\tilde{\nu}_x,\tilde{\nu}_y)| \le 1$ for any $\alpha,\beta\in[0,\pi]$. The numerical result shows that a sufficient condition for the two-dimensional DG(1)–Hancock to be stable is

$$\tilde{\nu}^{\mathrm{2D}} := (\tilde{\nu}_x+\tilde{\nu}_y) \le 0.664, \tag{117}$$

more restrictive than in one dimensional, and also more restrictive than for the two-dimensional HR2–RK2/Hancock method ($\tilde{\nu}^{\mathrm{2D}} \le 1.0$), but still 50% less restrictive than for the two-dimensional DG(1)–RK2 method ($\tilde{\nu}^{\mathrm{2D}} \le 0.333$). The stability limits for the upwind and Lax–Friedrichs fluxes are also summarized in Table 4.

III.D.2. Dominant Dissipation/Dispersion Error in 2-D

The results of a Fourier analysis for each method are listed below for comparison.

semi-discrete methods:

$$\mathrm{LTE}_{\mathrm{HR2RK2}} = c_3\left[1+2(r+s)^2\nu^2\right]k^3 \quad + \quad c_4\left[1-(r+s)^3\nu^3\right]k^4, \tag{118a}$$

$$\mathrm{LTE}^{(1)}_{\mathrm{DG(1)RK2}} = c_3\left[2(r+s)^2\nu^2\right]k^3 \quad +\frac{4}{9}c_4\left[1-\frac{9}{4}(r+s)^3\nu^3\right]k^4, \tag{118b}$$

$$\mathrm{LTE}^{(1)}_{\mathrm{DG(1)RK3}} = \quad \frac{4}{9}c_4\left[1+\frac{3}{4}(r+s)^3\nu^3\right]k^4 \tag{118c}$$

fully discrete methods:

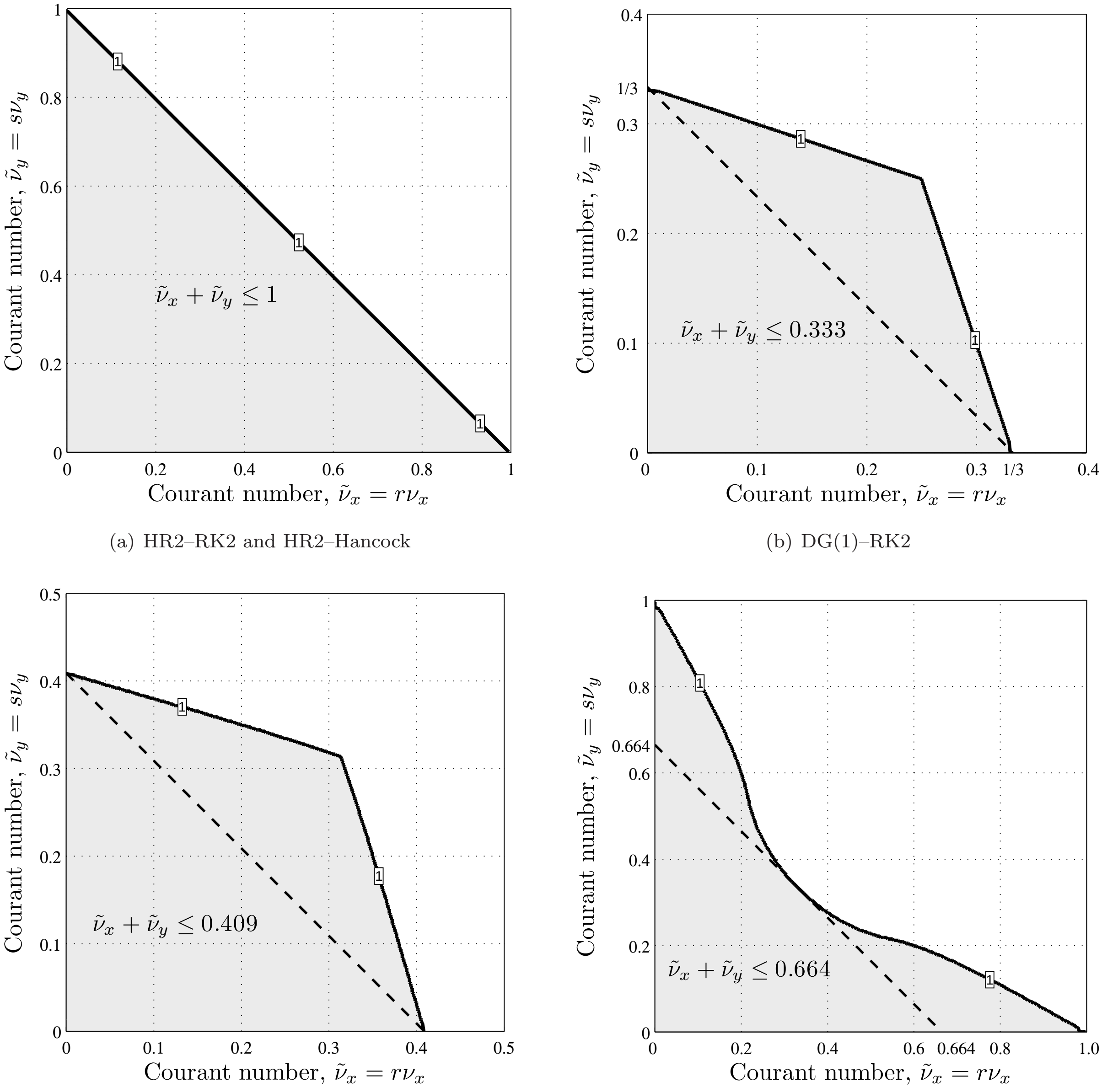
$$\mathrm{LTE}_{\mathrm{HR2Ha}} = c_3\left[1-3(r+s)\nu+2(r+s)^2\nu^2\right]k^3+\quad c_4\left[1-2(r+s)\nu+2(r+s)^2\nu^2-(r+s)^3\nu^3\right]k^4, \tag{118d}$$

$$\mathrm{LTE}^{(1)}_{\mathrm{DG(1)Ha}} = \quad \frac{4}{9}c_4\left[1-\frac{5}{4}(r+s)\nu+\frac{1}{2}(r+s)^2\nu^2-\frac{1}{4}(r+s)^3\nu^3\right]k^4, \tag{118e}$$

where

$$c_3 = -\frac{i(r+s)}{12}\boxed{\Delta h^2}, \quad c_4 = -\frac{(r+s)}{8}\boxed{\Delta h^3}. \tag{119}$$

The local truncation errors show the dominant dispersion error $\big(O\big(k^3\big)\text{-term}\big)$, and the dissipation error, $\big(O\big(k^4\big)\text{-term}\big)$. Compared to the one-dimensional results (105), the leading dissipation error of a two-dimensional DG(1) method increases by a factor 4 due to the multi-dimensionality. In contrast, an HR2 method possesses the same amount of dispersion and dissipation for both one- and two-dimensional discretizations. The DG(1)–Hancock and DG(1)–RK3 methods are superior with a leading error $O\big(\Delta h^3\big)$; the rest of the methods have error $O\big(\Delta h^2\big)$.



(a) HR2–RK2 and HR2–Hancock

(b) DG(1)–RK2

(c) DG(1)–RK3

(d) DG(1)–Hancock

Figure 5. Stability domain of various upwind methods applied to the 2-D linear advection equation. The shaded area indicates the region where $|g_{\mathbf{method}}(\tilde{\nu}_x, \tilde{\nu}_y)| \leq 1$ for $\alpha, \beta \in [0, \pi]$. The Hancock time-integration yields the largest stability domain even in two dimension.

Table 4. Maximum 2-D Courant number, $\tilde{\nu}_{\max}^{2D} := (\tilde{\nu}_x + \tilde{\nu}_y)_{\max}$, for various methods applied to the 2-D linear advection equation. The stability domain of DG(1)–Hancock reduces to $\tilde{\nu}_{\max}^{2D} = 0.664$ in two dimensions, yet greater than for DG(1)–RK2/RK3.

	method	order p	maximum Courant number, $\tilde{\nu}_{\max}^{2D} := (\tilde{\nu}_x + \tilde{\nu}_y)_{\max}$	
			upwind: $(q_x, q_y) = (r, s)$	Lax–Friedrichs: $(q_x, q_y) = \left(\dfrac{\Delta x}{\Delta t}, \dfrac{\Delta y}{\Delta t}\right)$
semi-discrete	HR2–RK2	2	1.0	1.0
	DG(1)–RK2	2	0.333	unstable
	DG(1)–RK3	3	0.409	unstable
fully discrete	HR2–Hancock	2	1.0	1.0
	DG(1)–Hancock	3	0.664	unstable

IV. Analysis for 1-D Linear Hyperbolic-Relaxation Equations

In the previous section, discretization methods for hyperbolic conservation laws were analyzed by adopting the 1-D and 2-D linear advection equations as the model equations. In order to extend the analysis to a method for hyperbolic-relaxation equations, we now consider a dimensionless 2×2 linear model system,[25–27]

$$\partial_t u + \partial_x v = 0, \tag{120a}$$

$$\partial_t v + \partial_x u = -\frac{1}{\epsilon}(v - ru). \tag{120b}$$

Here, u is the conserved variable, v is the flux of u, and $\epsilon > 0$ is a dimensionless relaxation time. In vector form, $\mathbf{u} = [u, v]^T, \mathbf{f} = [v, u]^T$, and $\mathbf{s} = [0, ru - v]^T$ in (1). This system has ‘frozen’ wave speeds ±1 when relaxation is weak ($\epsilon \gg 1$); when relaxation dominates ($\epsilon \ll 1$), it reduces to the advection-diffusion equation,

$$\partial_t u + r\partial_x u = \epsilon(1 - r^2)\partial_{xx} u + O\big(\epsilon^2\big)\,, \tag{121}$$

with an ‘equilibrium’ wave speed r. For stability, $|r| \leq 1$.

Previously, a Fourier analysis of the semi-discrete HR2 and DG(1) methods showed that the upwind flux based on the frozen wave speeds ±1 applied to the model equations (120) reduces to the Rusanov flux ($q = 1$) when $\epsilon \to 0$.[10] Hence, at the discretization level, solving (120) by using the upwind flux (also with $q = 1$ for this system) is identical to directly solving (121) with the Rusanov flux ($q = 1$) when $\epsilon \to 0$. Note that the upwind flux for the advection-diffusion equation (121) is obtained by $q = r$. This means that the semi-discrete HR2 and DG(1) methods for (120) are not strictly upwind in the equilibrium limit ($\epsilon \to 0$). It was further shown that the dominant dispersion/dissipation errors of semi-discrete methods in the low-frequency limit are given by (68) for HR2 and (78) for DG(1) with $q = 1$, while the exact solution in this case is

$$\lambda_{\text{exact}} = -irk - \epsilon(1 - r^2)k^2 - 2i\epsilon^2 r(1 - r^2)k^3 + O\big(\epsilon^3\big)\,. \tag{122}$$

In this paper, we extend the analysis to a fully discrete method.

At first, to demonstrate an extra difficulty arising due to the stiff source term, operator splitting is adopted in the time integrator. This splitting decouples the time evolution of the flux and source terms, allowing us to compute these independently. The great advantage of this method, particularly for hyperbolic-relaxation equations, is that the source term, which yields exponential damping, can be integrated exactly. In order to isolate the error introduced by the operator splitting, we eliminate the spatial discretization error by taking the flux derivative from the exact solution. Thus, the operator-split update operator for (120) taken the form

$$\begin{aligned}\mathbf{u}^{(1)} &= e^{\frac{\Delta t}{2}\mathbf{Q}}\mathbf{u}^{n},\\ \mathbf{u}^{(2)} &= e^{-ik\Delta t\mathbf{M}}\mathbf{u}^{(1)},\\ \mathbf{u}^{n+1} &= e^{\frac{\Delta t}{2}\mathbf{Q}}\mathbf{u}^{(2)},\end{aligned} \tag{123}$$

where

$$\mathbf{M} = \frac{\partial \mathbf{f}}{\partial \mathbf{u}} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad \text{and} \quad \mathbf{Q} = \frac{1}{\epsilon}\begin{pmatrix} 0 & 0 \\ r & -1 \end{pmatrix}. \tag{124}$$

Following the same procedure as in the previous section, the local truncation error in the low-frequency limit is found to be

$$\begin{aligned}\mathrm{LTE}_{\text{splitting}} &= \left[\frac{(1+e^{\Delta t/\epsilon})(1-r^2)\Delta t}{2(1-e^{\Delta t/\epsilon})} + \epsilon(1-r^2)\right]k^2 + O\big(k^3\big)\\ &\simeq -\frac{1-r^2}{12}\frac{\boxed{\Delta t^2}}{\epsilon}k^2 = -\frac{(1-r^2)\nu^2}{12}\frac{\boxed{\Delta x^2}}{\epsilon}k^2,\end{aligned} \tag{125}$$

where the Courant number is defined by (45). The above equation shows that the splitting is second-order in space and time. However, since the above error is in the $O\big(k^2\big)$-term, the numerical dissipation adds to the physical dissipation $-\epsilon(1-r^2)k^2$ in (122), leading to an incorrect diffusion coefficient. Furthermore, in the near-equilibrium limit ($O(\epsilon) \ll 1$), the time step and grid size must be severely restricted,

$$O(\Delta t) = O(\nu\Delta x) = O(\epsilon)\,, \tag{126}$$

otherwise the excessive numerical dissipation damps all waves in the domain.

The above example shows that straightforward decoupling of flux and source term leads to accuracy in the near-equilibrium limit only when (126) is satisfied. In order to overcome this severe restriction, coupling between flux and source term is necessary; for instance, an MOL with several stages, or a fully discrete method in which the flux is affected by the source term. Below, we examine a Fourier analysis of four methods applied to the linear hyperbolic-relaxation equations (120). The local truncation error of DG(1)–Hancock is compared to HR2–MOL, DG(1)–MOL, and HR2–Hancock. As the time integrator for HR2–MOL and DG(1)–MOL, we adopt a second-order implicit-explicit (IMEX) Runge–Kutta method: IMEX–SSP2(3,3,2).[28] This time integrator requires three stages for both explicit and implicit terms to achieve second-order accuracy. The update formulas are given by

$$
\begin{aligned}
\mathbf{u}^{(1)} &= \mathbf{u}^n + \frac{\Delta t}{4\epsilon}\mathbf{s}(\mathbf{u}^{(1)}), \\
\mathbf{u}^{(2)} &= \mathbf{u}^n - \frac{\Delta t}{2}\partial_x \mathbf{f}(\mathbf{u}^{(1)}) + \frac{\Delta t}{4\epsilon}\mathbf{s}(\mathbf{u}^{(2)}), \\
\mathbf{u}^{(3)} &= \mathbf{u}^n - \frac{\Delta t}{2}\left[\partial_x \mathbf{f}(\mathbf{u}^{(1)}) + \partial_x \mathbf{f}(\mathbf{u}^{(2)})\right] + \frac{\Delta t}{3\epsilon}\left[\mathbf{s}(\mathbf{u}^{(1)}) + \mathbf{s}(\mathbf{u}^{(2)}) + \mathbf{s}(\mathbf{u}^{(3)})\right], \\
\mathbf{u}^{n+1} &= \mathbf{u}^n - \frac{\Delta t}{3}\left[\partial_x \mathbf{f}(\mathbf{u}^{(1)}) + \partial_x \mathbf{f}(\mathbf{u}^{(2)}) + \partial_x \mathbf{f}(\mathbf{u}^{(3)})\right] + \frac{\Delta t}{3\epsilon}\left[\mathbf{s}(\mathbf{u}^{(1)}) + \mathbf{s}(\mathbf{u}^{(2)}) + \mathbf{s}(\mathbf{u}^{(3)})\right].
\end{aligned}
\tag{127}
$$

The local truncation error of each method is obtained after some algebra, yielding

semi-discrete methods:

$$
\mathrm{LTE}_{\mathrm{HR2MOL}} = \left[c_3\left(1 + (r\nu)^2\right) + \frac{1}{6}\tilde{c}_3\nu\right]k^3 + \left[c_4\left(\frac{1}{r} - \frac{1}{3}(r\nu)^3\right) + \frac{1}{6}\tilde{c}_4\right]k^4, \tag{128a}
$$

$$
\mathrm{LTE}^{(1)}_{\mathrm{DG(1)MOL}} = \left[c_3(r\nu)^2 + \frac{1}{6}\tilde{c}_3\nu\right]k^3 + \left[\frac{1}{9}c_4\left(r - 3(r\nu)^3\right) + \frac{1}{12}\tilde{c}_4\right]k^4, \tag{128b}
$$

fully discrete methods:

$$
\mathrm{LTE}_{\mathrm{HR2Ha}} = \left[c_3\left(1 - 3\nu + 2(r\nu)^2\right) + \frac{1}{2}\tilde{c}_3\nu\right]k^3 + \left[c_4\left(\frac{1}{r} - (r\nu)^3 + 2r\nu(\nu - 1)\right) + \frac{1}{3}\tilde{c}_4\left(1 + \frac{3}{4}\nu\right)\right]k^4, \tag{128c}
$$

$$
\begin{aligned}
\mathrm{LTE}^{(1)}_{\mathrm{DG(1)Ha}} = &\left[\frac{1}{9}c_4\left(\frac{r\left(1 - (r\nu)^2\right)^2}{1 - r^2\nu} - 3r\nu(1 - \nu)\right)\right. \\
&\left. + \frac{\tilde{c}_4}{12(1 - r^2\nu)^2}\left(1 + \frac{1}{3}r^2 + \frac{\nu^2}{6}\left(2r^2(r^2 - 9) + 3r^4\nu + r^4(9 - 7r^2)\nu^2 + 3r^6\nu^3\right)\right)\right]k^4,
\end{aligned}
\tag{128d}
$$

where the coefficients of errors attributed to the flux (explicit) discretization are

$$
c_3 = -\frac{ir}{12}\boxed{\Delta x^2}, \quad c_4 = -\frac{r}{8}\boxed{\Delta x^3}, \tag{129}
$$

and the source term (implicit) errors have coefficients

$$
\tilde{c}_3 = -i\epsilon r(1 - r^2)\boxed{\Delta x}, \quad \tilde{c}_4 = -\epsilon(1 - r^2)\boxed{\Delta x^2}. \tag{130}
$$

The above equations show that HR–MOL, DG(1)–MOL, and HR–Hancock have a *first-order* error $\sim k^3$, as $\tilde{c}_3 = O(\epsilon\Delta x)$. However, numerically, this error term is not pronounced in the near-equilibrium limit since $\epsilon \ll 1$, thus the dominant error of the methods stem from $c_3 = O\left(\Delta x^2\right)$. Similarly, the dominant error of DG(1)–Hancock is $c_4 = O\left(\Delta x^3\right)$.

V. Numerical Experiments

V.A. Linear Advection Equation

To confirm the previous analysis and demonstrate the efficiency of the DG(1)–Hancock method, the 1-D linear advection equation,

$$
\partial_t u + \partial_x u = 0 \quad \text{with} \quad r = \frac{1}{2}, \quad u(x, 0) = \cos(2\pi x), \tag{131}
$$

is solved over the domain $x \in [0,1]$ with periodic boundary conditions. The numerical solution at $t_{\text{end}} = 300$, after the harmonic wave has propagated 150 times across the domain, is compared to the exact solution. The upwind flux is used to compute a cell-interface flux, and we set the Courant number for each method equal to 90% of the method's linear stability limit listed in Table 3;

$$\nu_{\text{HR2–RK2}} = \nu_{\text{HR2–Hancock}} = \nu_{\text{DG(1)–Hancock}} = 0.9, \tag{132a}$$

$$\nu_{\text{DG(1)–RK2}} = 0.3, \quad \nu_{\text{DG(1)–RK3}} = 0.37, \quad \nu_{\text{DG(2)–RK3}} = 0.19. \tag{132b}$$

At first, to assess the dispersion and dissipation errors of methods qualitatively, the numerical solution by DG(1)–Hancock at t_{end} is plotted together with solutions by three other methods: HR2–RK2, HR2–Hancock, and DG(1)–RK2. The numerical results superimposed on the exact solution are shown in Figure 6. The DG(1)–Hancock method produces the least dispersive/dissipative result among the four methods. As shown in the local truncation errors (105), the leading error of the HR2–RK2, DG(1)–RK2, and HR2–Hancock methods is dispersive, caused by the $O\big(k^3\big)$-term. Thus, a traveling wave solution suffers especially with HR2–RK2 and DG(1)–RK2. However, HR2–Hancock has surprisingly little dispersive/dissipative error, which can be understood from the shift condition (88) which the Hancock method would satisfy at $\tilde{\nu} = 1$.

Secondly, a grid convergence study is conducted, with results are shown in Figure 7. The L_2-norms of solution errors are plotted against the number of degrees of freedom (solution parameters). It is seen that DG(1)–Hancock converges with third-order accuracy, while its error levels are almost comparable to those of DG(2)–RK3.

Even though Figure 7 provides the accuracy of a method, it does not show its efficiency. Therefor, L_2-norms of solution errors are plotted against CPU time in Figure 8. This figure reveals that the DG(1)–Hancock is actually more efficient than DG(2)–RK3: the former method has only two unknowns $\big($DG(1)$\big)$ and a two-stage update formula (Hancock), whereas the latter method has three unknowns $\big($DG(2)$\big)$ and a three-stage update formula $\big($RK(3)$\big)$.

Finally, CPU time normalized by the CPU time of the DG(1)–RK2 method for a specific error level is shown in Figure 9. Remarkably, the DG(1)–Hancock method is almost two orders of magnitude more efficient than the DG(1)–RK2 method. However, this could be a flattered result since the model equation we are solving is merely the 1-D linear advection equation. We expect that, when a nonlinear problem is considered, the efficiency of DG(1)–Hancock will be degraded; in fact, the numerical results in the next section still show an order of magnitude difference with DG(1)–MOL for a 1-D nonlinear hyperbolic-relaxation system.

V.B. Hyperbolic-Relaxation Equations

V.B.1. Model Equations and Initial Conditions

To demonstrate the accuracy of the DG(1)–Hancock method when applied to a nonlinear hyperbolic-relaxation system, the Euler equations with heat transfer, which reduce to the isothermal Euler equations

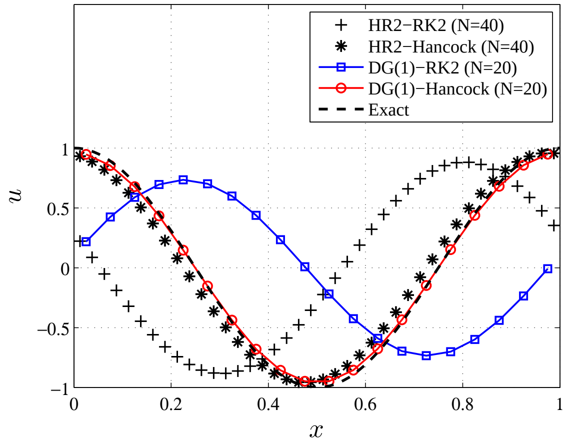


Figure 6. Numerical results of four methods at $t_{\text{end}} = 300$ in problem (131). The DG(1)–Hancock method appears to be the least dissipative and dispersive.

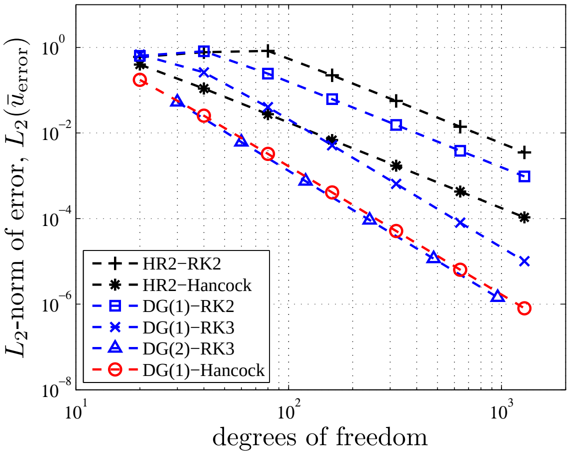


Figure 7. L_2-norms of error plotted against number of degrees of freedom. DG(1)–Hancock is almost comparable to DG(2)–RK3.

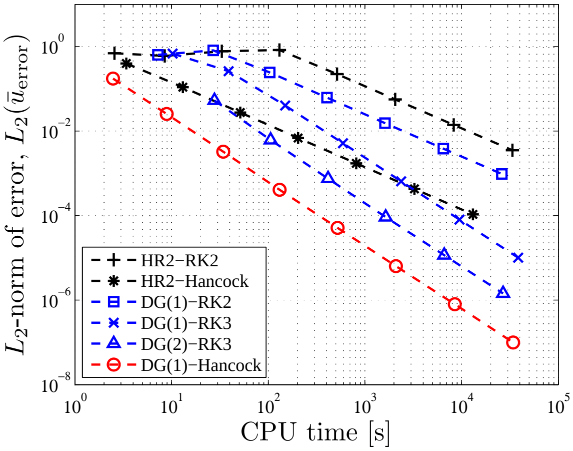


Figure 8. L_2-norms of error plotted against CPU time. DG(1)–Hancock is the most efficient method.

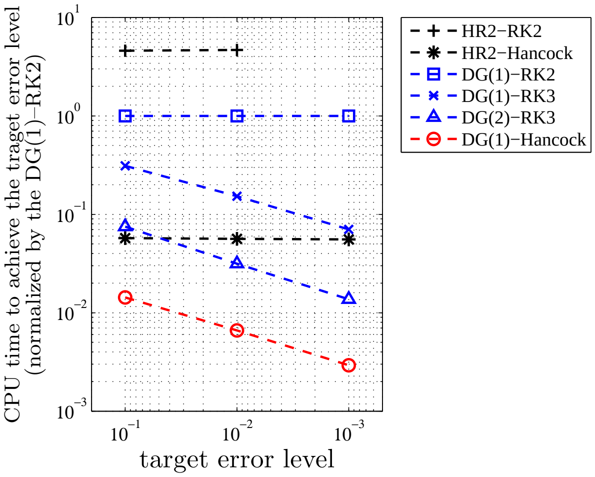


Figure 9. CPU time required to achieve the target error level, normalized by the DG(1)–RK2 result. The high efficiency of DG(1)–Hancock is evident.

in the equilibrium limit, are adopted as a model equation:[29]

$$\frac{\partial}{\partial t}\begin{pmatrix}\rho\\ \rho u\\ \rho E\end{pmatrix}+\frac{\partial}{\partial x}\begin{pmatrix}\rho u\\ \rho u^2+p\\ \rho u H\end{pmatrix}=-\frac{1}{\epsilon}\begin{pmatrix}0\\ 0\\ \rho(T-T_0)\end{pmatrix}, \tag{133}$$

where the pressure is given by the ideal gas law, $p := (\gamma-1)\rho e = \rho R T$. The frozen characteristic speeds are $u \pm a, u$, where the speed of sound is given by $a := \sqrt{\gamma p/\rho}$. In the equilibrium limit ($\epsilon \to 0$), the nonequilibrium temperature T converges to the constant equilibrium temperature T_0, instantaneously. As a result, the above equations tend asymptotically to the following isothermal Euler equations:

$$\frac{\partial}{\partial t}\begin{pmatrix}\rho\\ \rho u\end{pmatrix}+\frac{\partial}{\partial x}\begin{pmatrix}\rho u\\ \rho u^2+p^*\end{pmatrix}=\begin{pmatrix}0\\ 0\end{pmatrix}, \tag{134}$$

where the gas becomes polytropic and the pressure is given by $p^*(\rho) := \rho R T_0$. The equilibrium characteristic speeds are $u \pm a^*$, where the constant speed of sound is $a^* := \sqrt{p^*/\rho} = \sqrt{RT_0}$.

Consider an initial-value problem with the following $\mathbb{C}^\infty$ initial distributions:

$$\rho_0(x) = \exp\left(\frac{u_0(x)}{a^*}\right), \tag{135a}$$

$$u_0(x) = \begin{cases} -a^*, & x < -5, \\ a^* \tanh\left[-\dfrac{10x}{(x+5)(x-5)}\right], & x \in [-5,5], \\ a^*, & x > 5, \end{cases} \tag{135b}$$

$$p_0(x) = (a^*)^2 \rho_0(x), \tag{135c}$$

plotted in Figure 10. The initial conditions are chosen such that the analytical solution of the isothermal Euler equations becomes a simple wave solution; one of the Riemann invariants remains constant: $J^-_{\text{iso}}(x,t) = \ln\rho_0 - \dfrac{u_0}{a^*} \equiv 0$. Also, the flow properties are non-equilibrium ($T \neq T_0$) only within the domain $x \in [-5,5]$. The equilibrium speed of sound, equilibrium temperature, and the ratio of specific heats are set as follows:

$$a^* = \sqrt{0.4}, \quad T_0 = 1, \quad \gamma = 1.4. \tag{136}$$

The computational domain is confined to $x \in [-16, 16]$ with uniform grids, and the solution at the time $t_{\text{end}} = 5.0$ is used for checking grid convergence and comparison among methods.

V.B.2. Richardson Extrapolation for Grid Convergence

In order to determine the order of accuracy of the various schemes, we need to know the exact solution of (133) at the time t_{end}. When the frozen limit ($\epsilon \to \infty$) is considered, the exact solution must be obtained with the regular Euler equations. Conversely, at the equilibrium limit ($\epsilon \to 0$), the exact solution is derived from the isothermal Euler equations, (134). Simple-wave solutions are available for these two conservation

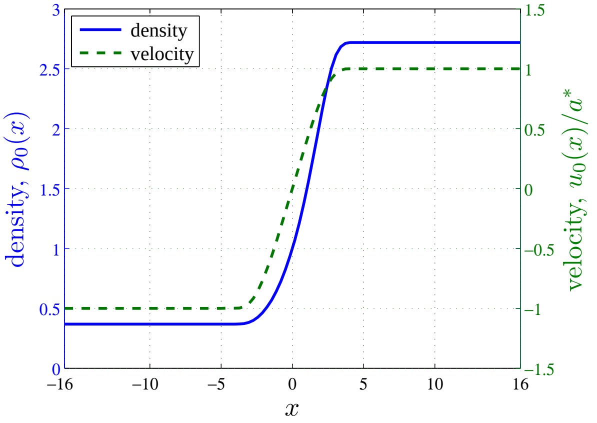


Figure 10. The distribution of the initial conditions for density, $\rho_0(x)$, and normalized velocity, $u_0(x)/a^*$.

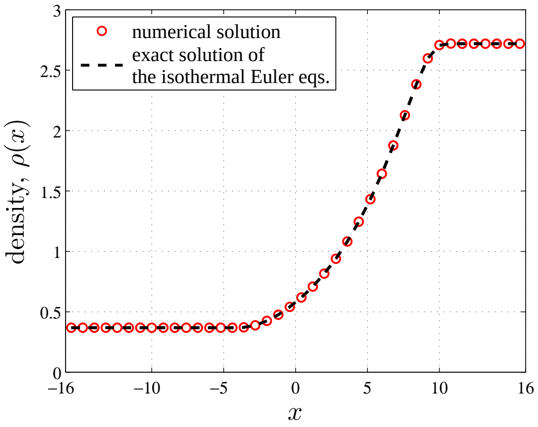


Figure 11. The density distribution at $t_{\text{end}} = 5.0$, computed by the DG(1)–Hancock method, is superposed to the exact solution of the isothermal Euler equations.

laws; however, the resulting exact solutions are not strictly the exact solution of (133). For instance, when $O(\epsilon) \ll 1$, an asymptotic expansion shows that a series of $O(\epsilon^k)$-term appears on the right-hand side of the conservation laws,

$$\partial_t \mathbf{U} + \partial_x \mathbf{F}(\mathbf{U}) = O(\epsilon)\, \partial_{xx} \mathbf{U} + \dots . \tag{137}$$

Thus, the actual exact solution should be derived from (137), which contains an infinite series in terms of ϵ. This could be possible if a simple system is considered; however, the derivation can be cumbersome.

To overcome this difficulty, Richardson extrapolation, which does not require knowledge of the exact solution, is adopted for the grid-convergence study. In brief, successive grid solutions provide an estimated exact solution, $\bar{u}_{i,\text{exact}}$, and the coefficients of the local truncation error, c_j, in the following form:

$$\bar{\mathbf{u}}_i = \bar{\mathbf{u}}_{i,\text{exact}} + \mathbf{c}_1 \Delta x + \mathbf{c}_2 \Delta x^2 + \mathbf{c}_3 \Delta x^3 + \dots . \tag{138}$$

Thus, once the right-hand side of the above equation is computed, the error at the cell i is given by

$$\begin{aligned} \text{error}_i(\mathbf{u}) &:= \bar{\mathbf{u}}_i - \bar{\mathbf{u}}_{i,\text{exact}} \\ &= (\mathbf{c}_1)_i \Delta x + (\mathbf{c}_2)_i \Delta x^2 + (\mathbf{c}_3)_i \Delta x^3 + \dots , \end{aligned} \tag{139}$$

after which, the L_p-norm on the uniform grid is obtained by

$$L_p(\mathbf{u}) := \left[\frac{1}{N} \sum_{i=1}^{N} |\text{error}_i(\mathbf{u})|^p \right]^{1/p} . \tag{140}$$

V.B.3. Numerical Results

The DG(1)–Hancock method is compared to two semi-discrete methods: HR2–MOL and DG(1)–MOL. As to the time integrator, we adopt the IMEX–SSP2(3,3,2) (127). In order to verify the accuracy of a method in the stiff regime ($\epsilon \ll O(1)$), the relaxation time is taken as

$$\epsilon = 10^{-8}. \tag{141}$$

Due to the implicit treatment of the source term, the time step is solely constrained by the maximum acoustic wave speed, thus

$$\Delta t = \nu_{\text{method}} \frac{\Delta x}{|u| + a}, \tag{142}$$

where ν_{method} is the Courant number of the method used. Here, we set a Courant number as 90% of a method's linear stability limit:

$$\begin{aligned} \nu_{\text{HR2–MOL}} &= 0.9, \\ \nu_{\text{DG(1)–MOL}} &= 0.3, \\ \nu_{\text{DG(1)–Hancock}} &= 0.9. \end{aligned} \tag{143}$$

The density distribution at $t_{\text{end}} = 5.0$, superposed to the exact solution of the isothermal Euler equations, is shown in Figure 11. Even though the exact solution of the isothermal Euler equations does not contain the $O(\epsilon)$-term, the numerical result is in good agreement with the exact solution. This is because the relaxation time, ϵ, is so small ($\epsilon = 10^{-8}$), thus the $O(\epsilon)$-term is negligible, at least in the eyeball norm.

To disclose the order of accuracy of each method, Richardson extrapolation is adopted for the grid-convergence study. The L_1-norm of density error, $L_1(\rho)$, is shown in Figure 12. The plot shows that the all three methods are second-order accurate, yet the DG(1)–Hancock method has an error nearly an order of magnitude lower than the HR2/DG(1)–MOL method. Note that previously the linear analysis predicts third-order convergence of the DG(1)–Hancock method (128d); however, due to the linearization of the source term in space (21), the method reduces to second-order accuracy for the nonlinear source.

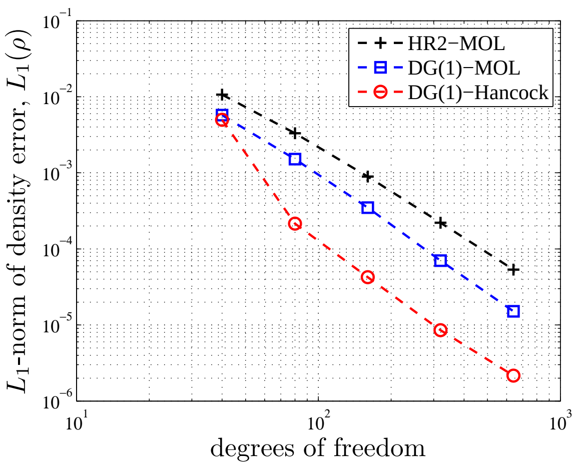


Figure 12. L_1-norms of density error, $L_1(\rho)$, for three methods are compared, showing the high accuracy of the DG(1)–Hancock method.

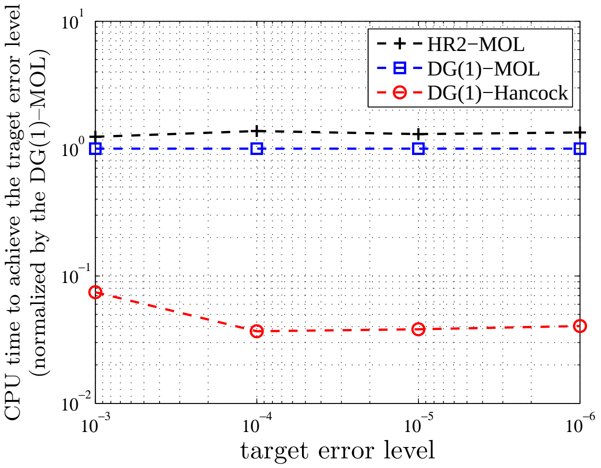


Figure 13. Three methods are compared regarding their overall efficiency. The CPU time required to achieve the target error level is normalized by the DG(1)–MOL method. The high efficiency of the DG(1)–Hancock method is observed.

In Figure 12 it is shown that the DG(1)–Hancock method is superior to the other two methods in terms of the accuracy; however, the method would not be attractive if it required enormous CPU time to achieve the high accuracy. Thus, we examine the overall efficiency of each method. Here, we define the efficiency based on the total CPU time to achieve a target error level. CPU time normalized by the CPU time of the DG(1)–MOL method for a specific error level is shown in Figure 13. It clearly shows the high efficiency of DG(1)–Hancock compared to HR2/DG(1)–MOL. Such a high efficiency is achieved by the combination of the accurate computation due to the DG spatial discretization and the wide stability domain owing to the Hancock temporal discretization.

VI. Conclusions

A discretization method, DG(1)–Hancock, for systems of 1-D nonlinear hyperbolic-relaxation equations is introduced. The method is based on Huynh's upwind moment scheme, with implicit treatment of the source term. For the source term integration, we adopt the fully-implicit two-point Radau IIA method. A Fourier analysis of the DG(1)–Hancock method applied to the 1-D and 2-D linear advection equations shows that the method is third-order accurate, and stable for $\tilde{\nu}^{1\mathrm{D}} \leq 1$ and $\tilde{\nu}^{2\mathrm{D}} \leq 0.664$ respectively. In addition, the local truncation error is compare among various methods; for 1-D, the dominant dissipation error of a DG method is $\frac{1}{9}$ of the error of a second-order Godunov-type method; for 2-D, it increases to $\frac{4}{9}$ due to the multidimensional discretization.

The method is then applied to a linear hyperbolic-relaxation systems for a Fourier analysis. The results obtained with upwind fluxes show that the dominant dispersion/dissipation errors in the near-equilibrium limit are identical to the local truncation error obtained for the linear advection equation, approximated with the Rusanov flux. The Euler equations with heat transfer, which reduce to the isothermal Euler equations in the equilibrium limit, are adopted as a model equation. The numerical results confirm the analysis, and show the superiority of the DG(1)–Hancock method over Method-Of-Line-based semi-discrete schemes. A comparison of the overall efficiencies of the methods shows that the DG(1)–Hancock method is not only producing highly accurate results, but the method is computationally more efficient in achieving a specific error level.

Acknowledgments

The authors would like to thank Hung T. Huynh for his insightful discussions of the upwind moment scheme.

References

[1] Barth, T. J., "Recent developments in high order k-exact reconstruction on unstructured meshes," *31st AIAA Aerospace Sciences Meeting and Exhibit*, Reno, Nevada; USA, 1993, AIAA Paper 1993-0668.

[2] Delanaye, M. and Liu, Y., "Quadratic reconstruction finite volume schemes on 3D arbitrary unstructured polyhedral

grids," *14th AIAA Computational Fluid Dynamics Conference*, Norfolk, Virginia; USA, 1999, AIAA Paper 1999-3259.

[3] Cockburn, B. and Shu, C. W., "Runge–Kutta discontinuous Galerkin methods for convection-dominated problems," *Journal of Scientific Computing*, Vol. 16, No. 3, 2001, pp. 173–261.

[4] Shu, C. W. and Osher, S., "Efficient implementation of essentially non-oscillatory shock-capturing schemes," *Journal of Computational Physics*, Vol. 77, No. 2, 1988, pp. 439–471.

[5] Huynh, H. T., "An upwind moment scheme for conservation laws," *Computational Fluid Dynamics 2004: Proceedings of the Third International Conference on Computational Fluid Dynamics, ICCFD3*, edited by C. Groth and D. W. Zingg, Springer, Berlin, 2006, pp. 761–766.

[6] Van Albada, G. D., Van Leer, B., and Roberts, J., W. W., "A comparative study of computational methods in cosmic gas dynamics," *Astronomy and Astrophysics*, Vol. 108, No. 1, 1982, pp. 76–84.

[7] Van Leer, B., "Upwind and high-resolution methods for compressible flow: From donor cell to residual-distribution schemes," *Communications in Computational Physics*, Vol. 1, No. 2, 2006, pp. 192–205.

[8] Van Leer, B., "Towards the ultimate conservative difference scheme. IV. A new approach to numerical convection," *Journal of Computational Physics*, Vol. 23, No. 3, 1977, pp. 276–299.

[9] Suzuki, Y. and Van Leer, B., "A discontinuous Galerkin method with Hancock-type time integration for hyperbolic systems with stiff relaxation source terms," *The Fourth International Conference on Computational Fluid Dynamics, ICCFD4*, Ghent; Belgium, 2006.

[10] Hittinger, J. A. F., Suzuki, Y., and Van Leer, B., "Investigation of the discontinuous Galerkin method for first-order PDE approaches to CFD," *17th AIAA Computational Flow Dynamics Conference*, Toronto, Ontario; Canada, 2005, AIAA Paper 2005-4989.

[11] Van Leer, B., "Towards the ultimate conservative difference scheme III. Upstream-centered finite-difference schemes for ideal compressible flow," *Journal of Computational Physics*, Vol. 23, No. 3, 1977, pp. 263–275.

[12] Rusanov, V. V., "The calculation of the interaction of non-stationary shock waves and obstacles," *USSR Computational Mathematics and Mathematical Physics*, Vol. 1, No. 2, 1962, pp. 304–320.

[13] Lax, P. D., "Weak solutions of nonlinear hyperbolic equations and their numerical computation," *Communications on Pure and Applied Mathematics*, Vol. 7, No. 1, 1954, pp. 159–193.

[14] Lambert, J. D., *Numerical Methods for Ordinary Differential Systems: The Initial Value Problem*, John Wiley & Sons, New York, 1991.

[15] Von Neumann, J. and Richtmyer, R. D., "On the numerical solutions of partial differential equations of parabolic type," U.S. Government Document LA-657, Los Alamos, December 25 1947.

[16] Richtmyer, R. D. and Morton, K. W., *Difference Methods for Initial-Value Problems*, Interscience Publishers, New York, 2nd ed., 1967.

[17] Tannehill, J. C., Anderson, D. A., and Pletcher, R. H., *Computational Fluid Mechanics and Heat Transfer*, Series in Computational and Physical Processes in Mechanics and Thermal Sciences, Taylor & Francis, Washington, D.C., 2nd ed., 1997.

[18] Van Leer, B., "Stabilization of difference schemes for the equations of inviscid compressible flow by artificial diffusion," *Journal of Computational Physics*, Vol. 3, No. 4, 1969, pp. 473–485.

[19] Cockburn, B. and Shu, C. W., "The Runge–Kutta local projection P^1-discontinuous-Galerkin finite element method for scalar conservation laws," *Rairo-Mahematical Modelling and Numerical Analysis-Modelisation Mathematique Et Analyse Numerique*, Vol. 25, No. 3, 1991, pp. 337–361.

[20] Chavent, G. and Cockburn, B., "Consistance et Stabilite des Schemas LRG pour les Lois de Conservation Scalaires," Tech. Rep. RR-0710, INRIA, 1987.

[21] Chavent, G. and Cockburn, B., "The local projection P^0-P^1-discontinuous-Galerkin finite element method for scalar conservation laws," *Rairo-Mathematical Modelling and Numerical Analysis-Modelisation Mathematique Et Analyse Numerique*, Vol. 23, No. 4, 1989, pp. 565–592.

[22] Rider, W. J. and Lowrie, R. B., "The use of classical Lax–Friedrichs Riemann solvers with discontinuous Galerkin methods," *International Journal for Numerical Methods in Fluids*, Vol. 40, No. 3-4, 2002, pp. 479–486.

[23] Wang, Z. J., "Spectral (finite) volume method for conservation laws on unstructured grids. Basic formulation," *Journal of Computational Physics*, Vol. 178, No. 1, 2002, pp. 210–251.

[24] Dumbser, M. and Munz, C. D., "Building blocks for arbitrary high order discontinuous Galerkin schemes," *Journal of Scientific Computing*, Vol. 27, No. 1-3, 2006, pp. 215–230.

[25] Jin, S. and Levermore, C. D., "Numerical schemes for hyperbolic conservation laws with stiff relaxation terms," *Journal of Computational Physics*, Vol. 126, No. 2, 1996, pp. 449–467.

[26] Hittinger, J. A. F., *Foundations for the Generalization of the Godunov Method to Hyperbolic Systems with Stiff Relaxation Source Terms*, Ph.D. thesis, The University of Michigan, 2000.

[27] Lowrie, R. B. and Morel, J. E., "Methods for hyperbolic systems with stiff relaxation," *International Journal for Numerical Methods in Fluids*, Vol. 40, No. 3-4, 2002, pp. 413–423.

[28] Pareschi, L. and Russo, G., "Implicit-explicit Runge–Kutta schemes and applications to hyperbolic systems with relaxation," *Journal of Scientific Computing*, Vol. 25, No. 1, 2005, pp. 129–155.

[29] Pember, R. B., "Numerical methods for hyperbolic conservation laws with stiff relaxation II. Higher-order Godunov methods," *SIAM Journal on Scientific Computing*, Vol. 14, No. 4, 1993, pp. 824–859.