Computers and Fluids 250 (2023) 105702

Contents lists available at ScienceDirect

Computers and Fluids

journal homepage: www.elsevier.com/locate/compfluid

A Kinetic Flux Difference Splitting method for compressible flows

K.S. Shrinath [a,b], N.H. Maruthi [c], S.V. Raghurama Rao [d,*], Veeredhi Vasudeva Rao [e]

[a] Research Scholar, Department of Aerospace Engineering, Indian Institute of Science, Bangalore, India
[b] Hindustan Aeronautics Limited (HAL), Bangalore, India
[c] SankhyaSutra Labs, Manyata Embassy Business Park, Bangalore, India
[d] Department of Aerospace Engineering, Indian Institute of Science, Bangalore, India
[e] Department of Mechanical and Industrial Engineering, University of South Africa (UNISA), Johannesburg, South Africa

ARTICLE INFO

Keywords:
Discrete velocity Boltzmann scheme
Kinetic Flux Difference Splitting
Exact discontinuity capturing
Relative entropy (Kullback–Liebler divergence)

ABSTRACT

A low diffusive flux difference splitting based kinetic scheme is developed based on a discrete velocity Boltzmann equation, with a novel three velocity model. While two discrete velocities are used for upwinding, the third discrete velocity is utilized to introduce appropriate additional numerical diffusion only in the expansion regions, identified using relative entropy (Kullback–Liebler divergence) at the cell-interface, along with the estimation of physical entropy. This strategy provides an interesting alternative to entropy fix, which is typically needed for low diffusive schemes. Grid-aligned steady discontinuities are captured exactly by fixing the primary numerical diffusion such that flux equivalence leads to zero numerical diffusion across discontinuities. Results for bench-mark test problems are presented for inviscid and viscous compressible flows.

1. Introduction

Kinetic or Boltzmann schemes have been interesting alternatives to the traditional Riemann solvers for the numerical simulation of fluid flows and this topic has been the focus of research for the past several decades. The basic framework for this class of numerical methods is the connection between the Boltzmann equation of the kinetic theory of gases and the macroscopic equations of gas dynamics, which can be obtained through the well-known *moment method strategy*. The advantage of this strategy is the linearity of the convection terms in the Boltzmann equation, though the collision term is nonlinear. The linearity of the convection terms makes the application of upwinding easier. Thus, this strategy avoids the complications of Riemann solvers, field-by-field decompositions and the strong dependence on eigenstructure. In this paper, a new upwind Boltzmann scheme named as *Kinetic Flux Difference Splitting* (KFDS) method is presented, utilizing the framework of a *Discrete Velocity Boltzmann Equation* (DVBE). The discrete velocity Boltzmann equation is first introduced based on the classical Boltzmann equation with a BGK model for the collision term, by replacing the Maxwellian distribution function by a set of Dirac delta functions. This formulation leads to simpler expressions for the equilibrium distribution and the moment relations. Applying upwinding to the discrete velocity Boltzmann equation leads to a macroscopic flux difference splitting method with simple expressions for the split fluxes. The discrete velocities are chosen based on physical and numerical considerations and one of the resulting schemes leads to exact capturing of steady grid-aligned discontinuities, a feature shared by a few of the well-known macroscopic schemes. Many of the existing kinetic schemes, like the flux vector splitting methods, suffer from high numerical diffusion and hence are not accurate enough compared to the Riemann solvers. The scheme presented in this paper overcomes this drawback, captures steady grid-aligned discontinuities exactly and is less diffusive otherwise. An interesting alternative to entropy fix, which is typically needed for less diffusive schemes, is introduced in this paper based on switching over to a three velocity model instead of a two velocity model in one dimension. The multi-dimensional extension is based on the standard finite volume method. The component of the discrete distribution corresponding to the additional discrete velocity is chosen based on the variation of relative entropy (also known as Kullback–Liebler divergence, directed divergence or D^2-distance) and entropy.

The first Boltzmann scheme was presented by Chu [1] in which a finite difference method is applied to the Boltzmann equation with the BGK model. Sanders & Prendergast [2] presented the *Beam scheme* in which the Maxwellian is replaced by a set of weighted Dirac delta functions (called beams) and the propagation of beams across cells is accounted for, in an algorithm which ensures conservation. van Albada et al. [3] demonstrated the upwinding property of Beam scheme.

* Corresponding author.
E-mail addresses: shrinath.k.s@gmail.com (K.S. Shrinath), maruthinh@gmail.com (N.H. Maruthi), raghu@iisc.ac.in (S.V. Raghurama Rao), vasudvr@unisa.ac.za (V. Vasudeva Rao).

https://doi.org/10.1016/j.compfluid.2022.105702
Received 13 April 2022; Received in revised form 3 September 2022; Accepted 11 October 2022
Available online 23 October 2022

Pullin [4] introduced an *Equilibrium Flux Method* (EFM) in which the fluxes across cells are calculated based on half-Maxwellians, which is equivalent to an upwind scheme. Reitz [5] proposed a Boltzmann scheme in which operator-splitting is used together with an appropriate velocity discretization for solving the Boltzmann equation. Harten, Lax and van Leer [6] gave a general description of kinetic schemes and suggested the equilibrium distribution function as a weighted sum of Dirac delta functions, connecting them to the eigen-structure of macroscopic Euler equations, thus generalizing the approach of Sanders and Prendergast. Deshpande [7] presented a second order accurate kinetic numerical method based on utilizing the Chapman–Enskog distribution to provide an anti-diffusive contribution to an otherwise inviscid scheme. Deshpande [8] and Mandal & Deshpande [9] introduced the Kinetic Flux Vector Splitting (KFVS) method which introduces upwinding directly to the convection terms of the Boltzmann equation, together with operator splitting and instantaneous relaxation to equilibrium, and this leads to a macroscopic upwind scheme with identical split flux expressions as in EFM, though the approach is different. Boltzmann schemes in which the Maxwellian is replaced by a compactly supported hat functions were introduced by Kaniel [10] and Perthame [11]. Prendergast and Kun Xu [12] developed a Boltzmann scheme without operator splitting, by using the solution of the BGK equation with appropriate approximations and a pressure sensor to detect the shocks. Raghurama Rao & Deshpande [13,14] introduced a peculiar velocity based upwind method which leads to a convection-pressure splitting at the macroscopic level, with simpler split flux expressions avoiding error functions and exponentials. It is worth noting that the macroscopic Flux Vector Splitting (FVS) method of Steger & Warming [15] can be recovered from the Beam scheme for a particular value of γ and similarly van Leer's FVS method [16] can be recovered from Perthame's kinetic scheme. While there are several other developments based on the above strategies, a slightly different approach is introduced with the framework of *discrete velocity based kinetic schemes*, by Natalini [17] and later by Aregba-Driollet & Natalini [18]. They introduce efficient kinetic schemes based discrete velocity Boltzmann equation and also establish the connection between the *relaxation systems* of Jin & Xin [19] and *Discrete Velocity Boltzmann systems*. While there is an extensive volume of research based on discrete velocity based kinetic schemes, some of the schemes which have the flavour of the traditional continuous molecular velocity based kinetic schemes are due to Bouchut et al. [20], Raghurama Rao & Balakrishna [21], Raghurama Rao & Subba Rao [22], Bajpayi and Raghurama Rao [23], Arun et al. [24–26], Raghavendra [27], Raghavendra and Raghurama Rao [28], Abgrall & Torlo [29]. An interesting connection of this discrete velocity Boltzmann framework to Lattice Boltzmann Method is introduced by Raghurama Rao et al. [30] and Deshmukh [31] by developing a Lattice Boltzmann Relaxation Scheme for compressible flows.

The discrete velocity based kinetic schemes are simpler than the continuous molecular velocity based kinetic schemes, as the integrals are replaced by summations and the Maxwellians are simple algebraic functions of conserved variables and fluxes. While some of the discrete velocity kinetic schemes are quite efficient, none of the above mentioned kinetic or discrete kinetic schemes can capture steady grid-aligned discontinuities exactly, a feat achieved by some of the macroscopic schemes. In this paper, an exact discontinuity capturing kinetic scheme is presented based on discrete velocity formulation, by choosing the discrete velocity magnitudes in such a way that steady grid-aligned discontinuities are captured without any numerical diffusion. While the design of this kinetic scheme requires just two discrete velocities, the three-velocity model is utilized in a novel way such that the third component is adjusted to avoid any entropy condition violation, which is typical of low diffusive algorithms, by utilizing the relative entropy. The relative entropy is also known as Kullback–Liebler divergence, the directed divergence or D^2-distance (see Kullback [32]). It represents the information theoretic distance between two bivariate normal distributions and its moment is an efficient indicator to distinguish among different nonlinear waves. It is also closely connected to the concept of Mahalanobis distance, introduced in 1936 [33]. Raghavendra et al. [34] utilized the D^2-Distance successfully as a tool for mesh adaptation. In the next section, the basics of moment method strategy are introduced and in the following section the discrete velocity Boltzmann equation and its moments to recover the macroscopic equations are introduced.

2. Moment method strategy for kinetic schemes

The kinetic or Boltzmann schemes are based on the fact that Euler equations of gas dynamics can be obtained as moments of the classical Boltzmann equation. Thus, the Euler equations can be written in the form

$$\left\langle \Psi \left(\underbrace{\frac{\partial f}{\partial t} + \vec{v} \cdot \frac{\partial f}{\partial \vec{x}} = J(f,f)}_{\text{Boltzmann equation}} \right) \right\rangle \tag{1}$$

where the moments are defined by

$$\langle \Psi f \rangle = \int_0^\infty \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \Psi f \; dv_1 dv_2 dv_3 \; dI \tag{2}$$

Here, f is the molecular velocity distribution function, v is the molecular velocity, I is the internal energy variable corresponding to non-translational degrees of freedom and $J(f,f)$ is the collision term. Ψ is the moment function vector, representing the mass, momenta and energy of the molecules (which are conserved during collisions).

$$\Psi = \begin{bmatrix} 1 \\ v_1 \\ v_2 \\ v_3 \\ I + \frac{1}{2}\left(v_1^2 + v_2^2 + v_3^2\right) \end{bmatrix} \tag{3}$$

The Euler equations (1), after completing the moments, are given in their macroscopic form by

$$\frac{\partial U}{\partial t} + \sum_{i=1}^{3} \frac{\partial G_i}{\partial x_i} = 0 \tag{4}$$

where U is the conserved variable vector and G_i are the inviscid flux vectors with standard definitions representing conservation of mass, momenta and energy.

$$U = \begin{bmatrix} \rho \\ \rho u_1 \\ \rho u_2 \\ \rho u_3 \\ \rho E \end{bmatrix}; \; G_1 = \begin{bmatrix} \rho u_1 \\ p + \rho u_1^2 \\ \rho u_1 u_2 \\ \rho u_1 u_3 \\ p u_1 + \rho u_1 E \end{bmatrix};$$

$$G_2 = \begin{bmatrix} \rho u_2 \\ \rho u_2 u_1 \\ p + \rho u_2^2 \\ \rho u_2 u_3 \\ p u_2 + \rho u_2 E \end{bmatrix}; \; G_3 = \begin{bmatrix} \rho u_3 \\ \rho u_3 u_1 \\ \rho u_3 u_2 \\ p + \rho u_3^2 \\ p u_3 + \rho u_3 E \end{bmatrix} \tag{5}$$

Here, the total energy is the sum of internal and kinetic energies.

$$E = \frac{p}{\rho(\gamma - 1)} + \frac{1}{2}\left(u_1^2 + u_2^2 + u_3^2\right) \tag{6}$$

γ is the ratio of specific heats. A simpler model for the collision term ($J(f,f)$) in the Boltzmann equation is given by the popular B–G–K model [35], which reduces the otherwise integro-differential equation to a PDE.

$$J(f,f) = -\frac{1}{\epsilon}\left[f - f^{eq}\right] \tag{7}$$

Here, ϵ is the relaxation time and f^{eq} is the distribution representing local thermodynamic equilibrium, given by a Maxwellian as

$$f^{eq} = \frac{\rho}{I_0}\left(\frac{\beta}{\pi}\right)^{\frac{3}{2}} e^{-\beta(v_1-u_1)^2-\beta(v_2-u_2)^2-\beta(v_3-u_3)^2} e^{-\frac{I}{I_0}} \tag{8}$$

where

$$\beta = \frac{1}{2RT} \text{ and } I_0 = \frac{(2+D)-\gamma D}{2(\gamma-1)}RT \tag{9}$$

with D being the degrees of freedom.

2.1. Strategy of kinetic schemes

2.1.1. Kinetic schemes for Euler equations

Utilizing the B–G–K model and operator splitting, the solution of the Boltzmann equation can be split into two steps as

$$\textbf{Convection Step}: \frac{\partial f}{\partial t} + \frac{\partial \vec{h}}{\partial \vec{x}} = 0 \tag{10}$$

$$\textbf{Collision Step}: \frac{df}{dt} = -\frac{1}{\epsilon}\left[f - f^{eq}\right] \tag{11}$$

where $\vec{h} = \vec{v}f$ is the flux, with the Boltzmann equation being written in conservation form. Choosing an *instantaneous relaxation to equilibrium* ($\epsilon = 0$), the collision step becomes a simple relaxation step as $f = f^{eq}$. Thus, the Euler equations can be written in an intriguing form as

$$\left\langle \Psi \left(\underbrace{\frac{\partial f}{\partial t} + \frac{\partial \vec{h}}{\partial \vec{x}} = 0}_{\text{convection}}, \underbrace{f = f^{eq}}_{\text{relaxation}} \right) \right\rangle \tag{12}$$

The essential advantage of using this representation for macroscopic Euler equations is the linearity of the convection terms, which makes introducing upwinding easier in a kinetic or Boltzmann scheme, automatically leading to an upwind scheme for Euler equations. Note that the nonlinearity is present only in the last step of taking moments. This strategy avoids dependence on the eigen-structure of the nonlinear Euler equations and hence is a good alternative to the Riemann solvers.

2.1.2. Kinetic schemes for Navier–Stokes equations

In the above formulation, f^{eq} represents local thermodynamic equilibrium and hence is insufficient to obtain Navier–Stokes equations. For this purpose, Chapman–Enskog distribution, f^{CE}, (derived using Chapman–Enskog expansion) can be used instead of f^{eq}. Then, we can write the Navier–Stokes equations in the moment form as

$$\left\langle \Psi \left(\frac{\partial f}{\partial t} + \frac{\partial \vec{h}}{\partial \vec{x}} = 0,\ f = f^{CE} \right) \right\rangle \tag{13}$$

where the instantaneous relaxation in the collision step is to Chapman–Enskog distribution (see Junk and Raghurama Rao [36]). The Chapman–Enskog distribution, for the general case of poly-atomic gases, is derived by Deshpande [7] for 1-D case and by Mano Kumar et al. [37] for multi-dimensions (see also Chou & Baganoff [38] and Deshpande et al. [39]). The numerical schemes in this work begin with the above distributions for introducing the discrete velocity versions.

2.1.3. Discrete velocity Boltzmann schemes

Here, we utilize the above formulations and introduce new versions of the equilibrium and Chapman–Enskog distribution functions based on discrete velocities. These discrete distributions are then utilized to introduce novel kinetic schemes for solving Euler and Navier–Stokes equations. Compared to traditional kinetic schemes, the expressions here are much simpler, as integrations are replaced by summations and Maxwellians are replaced by simpler algebraic expressions. Unlike in the Lattice Boltzmann Method, further Chapman–Enskog expansion (to fit the coefficients of viscosity and thermal conductivity) is not needed here as this formulation is based instantaneous relaxation to Maxwellian or Chapman–Enskog distributions in the collision step. One disadvantage with the B–G–K model is the limitation of unit Prandtl number, which is overcome here by employing a special strategy which introduces a correction, in a later section. It is worth noting here that, unlike in the Lattice Boltzmann Method, the numerical schemes developed in this work begin with the conservation form of the Boltzmann equation. Further, the framework of finite volume method and the flux difference splitting form of numerical diffusion ensure the preservation of the conservation form. These frameworks ensure conservation even if we make the discrete velocities functions of conserved variables. Such a modification only makes the numerical diffusion part in the flux difference splitting framework nonlinear but yet preserves conservation. Further, the essential numerical diffusion is fixed based on *flux equivalence across discontinuities*, which amounts to enforcing Rankine–Hugoniot (R–H) conditions, and R–H conditions represent the quintessential conservation of fluxes across discontinuities. This is similar to the modification of diffusion terms in expressions for the cell-interface fluxes in the macroscopic flux difference splitting methods, without the loss of conservation. In the next section, the new discrete velocity Boltzmann equation is introduced.

3. Equilibrium distribution function and its representations

The discrete velocity model is first introduced in 1-D in this section. Consider 1-D Euler equations given by

$$\frac{\partial U}{\partial t} + \frac{\partial G(U)}{\partial x} = 0 \tag{14}$$

where U is the conserved variable vector and $G(U)$ is its nonlinear flux vector, given by

$$U = \begin{bmatrix} \rho \\ \rho u \\ \rho E \end{bmatrix} \text{ and } G(U) = \begin{bmatrix} \rho u \\ p + \rho u^2 \\ pu + \rho u E \end{bmatrix} \tag{15}$$

Here, ρ is the density, u is the fluid velocity, p is the pressure and E is the total (internal + kinetic) energy, given by $E = \frac{p}{\rho(\gamma-1)} + \frac{1}{2}u^2$. The above equations can be obtained as moments of the 1-D Boltzmann equation, with the BGK model, given by

$$\frac{\partial f}{\partial t} + \frac{\partial h}{\partial x} = -\frac{1}{\epsilon}\left[f - f^{eq}\right] \tag{16}$$

where $h = vf$. The 1-D Maxwellian is given by

$$f^{eq} = \rho\frac{\sqrt{\beta}}{\sqrt{\pi}} e^{-\beta(v-u)^2} e^{-\frac{I}{I_0}} \text{ with } \beta = \frac{1}{2RT} \text{ and } I_0 = \frac{(3-\gamma)}{2(\gamma-1)}RT \tag{17}$$

where R is the gas constant and T is the temperature, with the ideal gas equation of state being given by $p = \rho RT$. The internal energy variable corresponding to non-translational degrees of freedom, I, takes the role of providing the right value of γ for poly-atomic case. The moments to obtain the macroscopic variables are defined by

$$U_i = \int_0^\infty \int_{-\infty}^\infty \psi_i f \, dv dI \text{ and } G_i(U) = \int_0^\infty \int_{-\infty}^\infty \psi_i v f \, dv dI, \ i = 1,2,3 \tag{18}$$

with

$$\psi = \begin{bmatrix} 1 \\ v \\ I + \frac{1}{2}v^2 \end{bmatrix} \tag{19}$$

Introducing a truncated distribution as

$$\tilde{f} = \int_0^\infty f \, dI \tag{20}$$

we can redefine the moment relations as

$$U_i = \int_{-\infty}^\infty \psi_i \tilde{f} \, dv \text{ and } G(U) = \int_{-\infty}^\infty \psi_i v \tilde{f} \, dv \tag{21}$$

3.1. Two velocity model

In order to obtain a discrete velocity model, let us start with the moment relations for the equilibrium distribution function given by

$$U_i = \int_{-\infty}^{\infty} \psi_i \tilde{f}^{eq} dv \text{ and } G(U)_i = \int_{-\infty}^{\infty} \psi_i v \tilde{f}^{eq} dv \tag{22}$$

Introducing a set of Dirac delta functions comprising of two discrete velocities λ^+ and λ^- (for replacing the molecular velocity v) and also two corresponding components of $\tilde{f}^{eq}$ as f_+^{eq} and f_-^{eq} (which absorb the effect of ψ), we write

$$\psi_i \tilde{f}^{eq} = \left\{ f_+^{eq} \delta(v - \lambda^+) + f_-^{eq} \delta(v - \lambda^-) \right\}_i \tag{23}$$

where $\delta\left(v - \lambda^{\pm}\right)$ is the Dirac delta function. Thus, the conserved variable vector becomes

$$U_i = \left[\int_{-\infty}^{\infty} f_+^{eq} \delta(v - \lambda^+) dv + \int_{-\infty}^{\infty} f_-^{eq} \delta(v - \lambda^-) dv \right]_i \tag{24}$$

Let us further assume, for simplicity, that the discrete velocities, λ^+ and λ^- for each i are given by

$$\lambda_i^+ = \lambda_i \quad \text{and} \quad \lambda_i^- = -\lambda_i \tag{25}$$

Thus, we have three unknowns, namely, f_+^{eq}, f_-^{eq} and λ to be fixed for the equilibrium distribution, for each i. The moment relations in (22) are utilized to fix $f_{\pm}^{eq}$ here. The fixing of λ is explained in later sections, to develop a low diffusion scheme.

$$\begin{aligned} U_i &= \int_{-\infty}^{\infty} \psi_i \tilde{f}^{eq} dv \\ &= \int_{-\infty}^{\infty} \left\{ f_+^{eq} \delta(v - \lambda^+) + f_-^{eq} \delta(v - \lambda^-) \right\}_i dv \\ &= \left\{ f_+^{eq} + f_-^{eq} \right\}_i \end{aligned}$$

Thus

$$\left\{ f_+^{eq} + f_-^{eq} \right\}_i = U_i \tag{26}$$

$$\begin{aligned} G(U)_i &= \int_{-\infty}^{\infty} v \psi_i \tilde{f}^{eq} dv \\ &= \int_{-\infty}^{\infty} v \left\{ f_+^{eq} \delta(v - \lambda^+) + f_-^{eq} \delta(v - \lambda^-) \right\}_i dv \\ &= \left\{ f_+^{eq} \int_{-\infty}^{\infty} \phi(v) \delta(v - \lambda^+) dv + f_-^{eq} \int_{-\infty}^{\infty} \phi(v) \delta(v - \lambda^-) dv \right\}_i, \\ &\quad (\phi(v) = v) \\ &= \left\{ f_+^{eq} \lambda^+ + f_-^{eq} \lambda^- \right\}_i \end{aligned}$$

Thus

$$\left\{ f_+^{eq} \lambda^+ + f_-^{eq} \lambda^- \right\}_i = G(U)_i \tag{27}$$

Solving the above two equations and simplifying of discrete velocities for each i using (25), we get

$$f_{+\,i}^{eq} = \frac{1}{2} U_i + \frac{1}{2\lambda_i} G(U)_i \text{ and } f_{-\,i}^{eq} = \frac{1}{2} U_i - \frac{1}{2\lambda_i} G(U)_i \tag{28}$$

Therefore, the *Discrete Velocity Boltzmann Equation* (DVBE) based on the two discrete velocity model can be written as

$$\left\{ \frac{\partial \mathbf{f}}{\partial t} + \frac{\partial \mathbf{h}}{\partial \mathbf{x}} = -\frac{\mathbf{1}}{\epsilon} \left[\mathbf{f} - \mathbf{f^{eq}} \right] \right\}_i \quad i = 1, 2, 3 \tag{29}$$

where $\mathbf{h_i} = \mathbf{\Lambda_i f_i}$ and

$$\mathbf{f_i} = \begin{bmatrix} f_+ \\ f_- \end{bmatrix}_\mathbf{i}, \ \mathbf{\Lambda_i} = \begin{bmatrix} \lambda^+ & 0 \\ 0 & \lambda^- \end{bmatrix}_\mathbf{i} \text{ and } \mathbf{f^{eq}_i} = \begin{bmatrix} f_+^{eq} \\ f_-^{eq} \end{bmatrix}_\mathbf{i} = \begin{bmatrix} \frac{1}{2} U + \frac{1}{2\lambda} G(U) \\ \frac{1}{2} U - \frac{1}{2\lambda} G(U) \end{bmatrix}_\mathbf{i} \tag{30}$$

The Eq. (29) can be written in compact form as

$$\frac{\partial \mathbf{F}}{\partial t} + \frac{\partial \mathbf{H}}{\partial \mathbf{x}} = -\frac{\mathbf{1}}{\epsilon} \left[\mathbf{F} - \mathbf{F^{eq}} \right] \tag{31}$$

where $\mathbf{H} = \tilde{\mathbf{\Lambda}} \mathbf{F}$ is the flux and

$$\mathbf{F} = \begin{bmatrix} \mathbf{f_1} \\ \mathbf{f_2} \\ \mathbf{f_3} \end{bmatrix} = \begin{bmatrix} f_{+1} \\ f_{-1} \\ f_{+2} \\ f_{-2} \\ f_{+3} \\ f_{-3} \end{bmatrix}, \text{ and } \mathbf{F^{eq}} = \begin{bmatrix} \mathbf{f^{eq}}_\mathbf{1} \\ \mathbf{f^{eq}}_\mathbf{2} \\ \mathbf{f^{eq}}_\mathbf{3} \end{bmatrix} = \begin{bmatrix} f_{+\,1}^{eq} \\ f_{-\,1}^{eq} \\ f_{+\,2}^{eq} \\ f_{-\,2}^{eq} \\ f_{+\,3}^{eq} \\ f_{-\,3}^{eq} \end{bmatrix} \tag{32}$$

$\tilde{\mathbf{\Lambda}}$ is given by

$$\tilde{\mathbf{\Lambda}} = \begin{bmatrix} \mathbf{\Lambda_1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Lambda_2} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{\Lambda_3} \end{bmatrix} = \begin{bmatrix} \lambda_1^+ & 0 & 0 & 0 & 0 & 0 \\ 0 & \lambda_1^- & 0 & 0 & 0 & 0 \\ 0 & 0 & \lambda_2^+ & 0 & 0 & 0 \\ 0 & 0 & 0 & \lambda_2^- & 0 & 0 \\ 0 & 0 & 0 & 0 & \lambda_3^+ & 0 \\ 0 & 0 & 0 & 0 & 0 & \lambda_3^+ \end{bmatrix} \tag{33}$$

where $\mathbf{f_i}$, $\mathbf{\Lambda_i}$ and $\mathbf{f^{eq}_i}$ are obtained from (30). It is worth noting that moments of the above discrete velocity Boltzmann equation, for the 2-velocity model, yield the relaxation system of Jin and Xin [19], as noted by Aregba-Driollet & Natalini [18]. To recover the macroscopic system of equations, we take moments, which in this case will be premultiplying the system with $\mathbf{P}$, which is defined as

$$\mathbf{P} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \tag{34}$$

3.2. Three velocity model

The basic motivation for this part is to keep the additional discrete velocity and its corresponding discrete distribution as free parameters, to be fixed for avoiding entropy condition violation in a novel way. Let us replace the equilibrium distribution by another combination of Dirac delta functions as

$$\psi_i \tilde{f}^{eq} = \left\{ f_+^{eq} \delta(v - \lambda^+) + f_o^{eq} \delta(v - \lambda_o) + f_-^{eq} \delta(v - \lambda^-) \right\}_i \tag{35}$$

Let us further assume, for simplicity, that the discrete velocities, λ^+ and λ^- for each i are given by

$$\lambda_i^+ = \lambda_i \text{ and } \lambda_i^- = -\lambda_i \tag{36}$$

Let us assume that λ^0 and f_0^{eq} for each i are known (which will be fixed later). Then, using the two moment relations in (21) we obtain the following.

$$\begin{aligned} U_i &= \int_{-\infty}^{\infty} \psi_i \tilde{f}^{eq} dv \\ &= \int_{-\infty}^{\infty} \left\{ f_+^{eq} \delta(v - \lambda^+) + f_o^{eq} \delta(v - \lambda_o) + f_-^{eq} \delta(v - \lambda^-) \right\}_i dv \\ &= \left\{ f_+^{eq} + f_o^{eq} + f_-^{eq} \right\}_i \end{aligned}$$

or

$$\left\{ f_+^{eq} + f_-^{eq} \right\}_i = U_i - f_{o\,i}^{eq} \tag{37}$$

$$\begin{aligned} G(U)_i &= \int_{-\infty}^{\infty} v \psi_i \tilde{f}^{eq} dv \\ &= \int_{-\infty}^{\infty} v \left\{ f_+^{eq} \delta(v - \lambda^+) + f_o^{eq} \delta(v - \lambda_o) + f_-^{eq} \delta(v - \lambda^-) \right\}_i dv \\ &= \left\{ f_+^{eq} \int_{-\infty}^{\infty} \phi(v) \delta(v - \lambda^+) dv + f_o^{eq} \int_{-\infty}^{\infty} \phi(v) \delta(v - \lambda_o) dv \right. \\ &\quad \left. + f_-^{eq} \int_{-\infty}^{\infty} \phi(v) \delta(v - \lambda^-) dv \right\}_i, \quad (\phi(v) = v) \\ &= \left\{ f_+^{eq} \lambda^+ + f_o^{eq} \lambda_o + f_-^{eq} \lambda^- \right\}_i \end{aligned}$$

or

$$\left\{ f_+^{eq} \lambda^+ + f_-^{eq} \lambda^- \right\}_i = G(U)_i - \left\{ f_o^{eq} \lambda_o \right\}_i \tag{38}$$

Solving the above two equations and simplifying the discrete velocities using (36), we get

$$f^{eq}_{+\,i} = \frac{1}{2}U_i + \frac{1}{2\lambda_i}G(U)_i - \left\{\frac{\lambda+\lambda_o}{2\lambda}f^{eq}_o\right\}_i$$
$$\text{and } f^{eq}_{-\,i} = \frac{1}{2}U_i - \frac{1}{2\lambda_i}G(U)_i - \left\{\frac{\lambda-\lambda_o}{2\lambda}f^{eq}_o\right\}_i \tag{39}$$

This is similar to the equilibria derived using the two velocity model (28) with the respective additional $-\frac{\lambda+\lambda_o}{2\lambda}f^{eq}_o$ and $-\frac{\lambda-\lambda_o}{2\lambda}f^{eq}_o$ terms. These additional terms will later be utilized in the numerical scheme to avoid possible entropy violations.

The *Discrete Velocity Boltzmann Equation* (DVBE) for three velocity model can now be written as

$$\left\{\frac{\partial \mathbf{f}}{\partial t} + \frac{\partial \mathbf{h}}{\partial x} = -\frac{1}{\epsilon}\left[\mathbf{f} - \mathbf{f^{eq}}\right]\right\}_i \quad i = 1,2,3 \tag{40}$$

where $\mathbf{h_i} = \mathbf{\Lambda_i f_i}$, with the above Boltzmann equation being in conservation form. Here

$$\mathbf{f_i} = \begin{bmatrix} f_+ \\ f_o \\ f_- \end{bmatrix}_\mathbf{i},\ \mathbf{\Lambda_i} = \begin{bmatrix} \lambda^+ & 0 & 0 \\ 0 & \lambda_o & 0 \\ 0 & 0 & \lambda^- \end{bmatrix}_\mathbf{i} \text{ and}$$
$$\mathbf{f^{eq}_i} = \begin{bmatrix} f^{eq}_+ \\ f^{eq}_o \\ f^{eq}_- \end{bmatrix}_\mathbf{i} = \begin{bmatrix} \frac{1}{2}U + \frac{1}{2\lambda}G(U) - \frac{\lambda+\lambda_o}{2\lambda}f^{eq}_o \\ f^{eq}_o \\ \frac{1}{2}U - \frac{1}{2\lambda}G(U) - \frac{\lambda-\lambda_o}{2\lambda}f^{eq}_o \end{bmatrix}_\mathbf{i} \tag{41}$$

f^{eq}_0 and λ^0 for each i will be chosen later in such a way that entropy condition violation is avoided. The Eq. (40) can be written in compact form as

$$\frac{\partial \mathbf{F}}{\partial t} + \frac{\partial \mathbf{H}}{\partial x} = -\frac{1}{\epsilon}\left[\mathbf{F} - \mathbf{F^{eq}}\right] \tag{42}$$

where

$$\mathbf{F} = \begin{bmatrix} \mathbf{f_1} \\ \mathbf{f_2} \\ \mathbf{f_3} \end{bmatrix} = \begin{bmatrix} f_{+1} \\ f_{o1} \\ f_{-1} \\ f_{+2} \\ f_{o2} \\ f_{-2} \\ f_{+3} \\ f_{o3} \\ f_{-3} \end{bmatrix} \text{ and } \mathbf{F^{eq}} = \begin{bmatrix} \mathbf{f^{eq}}_1 \\ \mathbf{f^{eq}}_2 \\ \mathbf{f^{eq}}_3 \end{bmatrix} = \begin{bmatrix} f^{eq}_{+\,1} \\ f^{eq}_{o\,1} \\ f^{eq}_{-\,1} \\ f^{eq}_{+\,2} \\ f^{eq}_{o\,2} \\ f^{eq}_{-\,2} \\ f^{eq}_{+\,3} \\ f^{eq}_{o\,3} \\ f^{eq}_{-\,3} \end{bmatrix} \tag{43}$$

$\tilde{\mathbf{\Lambda}}$ is given by

$$\tilde{\mathbf{\Lambda}} = \begin{bmatrix} \mathbf{\Lambda_1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Lambda_2} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{\Lambda_3} \end{bmatrix} = \begin{bmatrix} \lambda^+_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \lambda^o_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \lambda^-_1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \lambda^+_2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \lambda^o_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \lambda^-_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \lambda^+_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \lambda^o_3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \lambda^-_3 \end{bmatrix} \tag{44}$$

where $\mathbf{f_i}$, $\mathbf{\Lambda_i}$ and $\mathbf{f^{eq}_i}$ are obtained from (41). To recover the macroscopic system of equations, we take moments, which in this case will be by premultiplying the system with $\mathbf{P}$, which is defined as

$$\mathbf{P} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix} \tag{45}$$

Let us now develop a kinetic scheme for the discrete velocity models thus developed using a flux difference splitting approach.

4. Kinetic Flux Difference Splitting (KFDS) scheme

4.1. KFDS method with two velocity model

Let us write (29) in a finite volume framework

$$\left\{\mathbf{f}^{n+1}_j = \mathbf{f}^n_j - \frac{\Delta t}{\Delta x}\left[\mathbf{h}^n_{j+\frac{1}{2}} - \mathbf{h}^n_{j-\frac{1}{2}}\right]\right\}_i \tag{46}$$

where $\mathbf{h}_i = \mathbf{\Lambda}_i\mathbf{f}_i$. To introduce upwinding in two velocity model based DVBE, let us split the discrete velocity matrix $\mathbf{\Lambda}_i$ in to two parts, separating the positive and negative velocities.

$$\mathbf{\Lambda}_i = \begin{bmatrix} \lambda^+ & 0 \\ 0 & \lambda^- \end{bmatrix}_i = \begin{bmatrix} \lambda^+ & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & \lambda^- \end{bmatrix}_i = \mathbf{\Lambda}^+_i + \mathbf{\Lambda}^-_i \tag{47}$$

It is possible to write

$$|\mathbf{\Lambda}|_i = \mathbf{\Lambda}^+_i - \mathbf{\Lambda}^-_i \tag{48}$$

Thus the upwind fluxes for each i, applied on a three point stencil, can be written as

$$\left\{\mathbf{h}^n_{j+\frac{1}{2}} = [\mathbf{\Lambda}^+\mathbf{f}_{eq}]_j + [\mathbf{\Lambda}^-\mathbf{f}_{eq}]_{j+1}\right\}_i \tag{49}$$

$$\left\{\mathbf{h}^n_{j-\frac{1}{2}} = [\mathbf{\Lambda}^+\mathbf{f}_{eq}]_{j-1} + [\mathbf{\Lambda}^-\mathbf{f}_{eq}]_j\right\}_i \tag{50}$$

Using (47) and (48) we can write

$$\left\{\mathbf{h}^n_{j+\frac{1}{2}} = \underbrace{\frac{1}{2}[\mathbf{h}^n_{j+1} + \mathbf{h}^n_j]}_{\text{average flux}} - \underbrace{\frac{1}{2}|\Lambda|[\mathbf{f}^{eq}_{j+1} - \mathbf{f}^{eq}_j]}_{\text{diffusive flux}}\right\}_i \tag{51}$$

$$\left\{\mathbf{h}^n_{j-\frac{1}{2}} = \frac{1}{2}[\mathbf{h}^n_j + \mathbf{h}^n_{j-1}] - \frac{1}{2}|\Lambda|[\mathbf{f}^{eq}_j - \mathbf{f}^{eq}_{j-1}]\right\}_i \tag{52}$$

Clearly, Λ represents the coefficients of numerical diffusion. Choosing the values of λ_i for numerical considerations is an efficient strategy to control the numerical diffusion without losing conservation, as the above expressions for cell-interface fluxes in the flux difference splitting form enforce conservation in the basic finite volume framework. The above expressions can further be rewritten in flux difference splitting form as

$$\left\{\mathbf{h}^n_{j+\frac{1}{2}} = \frac{1}{2}[\mathbf{h}^n_{j+1} + \mathbf{h}^n_j] - \frac{1}{2}[\Delta\mathbf{h}^+_{j+\frac{1}{2}} - \Delta\mathbf{h}^-_{j+\frac{1}{2}}]\right\}_i \tag{53}$$

$$\left\{\mathbf{h}^n_{j-\frac{1}{2}} = \frac{1}{2}[\mathbf{h}^n_j + \mathbf{h}^n_{j-1}] - \frac{1}{2}[\Delta\mathbf{h}^+_{j-\frac{1}{2}} - \Delta\mathbf{h}^-_{j-\frac{1}{2}}]\right\}_i \tag{54}$$

where $\Delta\mathbf{h}^\pm_{j\pm\frac{1}{2}} = [\mathbf{\Lambda}^\pm\Delta\mathbf{f}_{eq}]_{j\pm\frac{1}{2}}$. In the next sub-section, the coefficient of numerical diffusion, which corresponds to $|\mathbf{\Lambda}|$, is chosen a function of both the left and right states. Thus, the flux difference splitting is the appropriate choice rather than flux vector splitting.

To recover the macroscopic update formula, let is take moments by multiplying with P as given in (34). Therefore the macroscopic update formula for the Kinetic Flux Difference Splitting (KFDS) scheme thus developed using 2-velocity model based DVBE can be written as

$$\left\{U^{n+1}_j = U^n_j - \frac{\Delta t}{\Delta x}\left[G(U)^n_{j+\frac{1}{2}} - G(U)^n_{j-\frac{1}{2}}\right]\right\}_i \tag{55}$$

where the interface fluxes are given by

$$\left\{G(U)^n_{j+\frac{1}{2}} = \frac{1}{2}[G(U)^n_{j+1} + G(U)^n_j] - \frac{1}{2}[\Delta G(U)^{+,n}_{j+\frac{1}{2}} - \Delta G(U)^{-,n}_{j+\frac{1}{2}}]\right\}_i \tag{56}$$

$$\left\{G(U)^n_{j-\frac{1}{2}} = \frac{1}{2}[G(U)^n_j + G(U)^n_{j-1}] - \frac{1}{2}[\Delta G(U)^{+,n}_{j-\frac{1}{2}} - \Delta G(U)^{-,n}_{j-\frac{1}{2}}]\right\}_i \tag{57}$$

$$\left\{\Delta G(U)^\pm_{j+\frac{1}{2}} = \frac{1}{2}[G(U)_{j+1} - G(U)_j] \pm \frac{1}{2}|\lambda|[U_{j+1} - U_j]\right\}_i \tag{58}$$

$$\left\{\Delta G(U)^\pm_{j-\frac{1}{2}} = \frac{1}{2}[G(U)_j - G(U)_{j-1}] \pm \frac{1}{2}|\lambda|[U_j - U_{j-1}]\right\}_i \tag{59}$$

Clearly, λ controls the numerical diffusion.

4.2. Fixing λ

For developing an accurate shock capturing scheme, the value of λ is fixed such that numerical diffusion vanishes for a steady discontinuity, as suggested in [6]. As a consequence, though there is a jump in the conserved variables across a steady discontinuity, the fluxes on the left and right side of the discontinuity are equal. This principle is referred to as *flux equivalence across a steady discontinuity* here. Equating the cell-interface flux separately to the left flux ($G(U)_{j+\frac{1}{2}} = G(U)_j$) and the right flux ($G(U)_{j+\frac{1}{2}} = G(U)_{j+1}$) and generalizing leads to

$$\Delta G(U) = |\lambda| \Delta U \tag{60}$$

which is nothing but the Rankine–Hugoniot (R–H) jump conditions. Thus, we can choose

$$|\lambda| = \left| \frac{\Delta G(U)}{\Delta U} \right| \tag{61}$$

However, as U and $G(U)$ are vectors, choosing a scalar numerical diffusion is not easy. We choose a diagonal matrix for representing λ (as in [40]) and this leads to

$$|\lambda|_i = \left| \frac{\Delta G(U)}{\Delta U} \right|_i, \; i = 1, 2, 3 \tag{62}$$

Choosing this value for λ_i, which represent the coefficients of numerical diffusion, will enforce R–H conditions directly in the discretization process. As R–H conditions represent the quintessential conservation of fluxes across discontinuities, conservation is enforced, as is also ensured by beginning with the conservative form of equations and the utilization of the finite volume framework. As a consequence of the above choice, this 2-velocity model based KFDS scheme can capture steady discontinuities exactly, without numerical diffusion. However, as this variant is low in numerical diffusion, entropy condition violation is likely to occur. A 3-velocity model based KFDS scheme is introduced in the next section in which the additional velocity and the corresponding component of its distribution function are chosen in a novel way to avoid any possible entropy condition violation.

It is worth noting here that though the numerical scheme initially begins with the discrete velocity form of the Boltzmann equation, as λ turns out to be the coefficient of numerical diffusion, a suitable choice for the value of λ (without losing conservation, in the finite volume and FDS framework, as noted before) changes the discrete nature of the velocities, thus leading to an essentially new framework. Yet, the name DVBE is retained here, for the sake of convenience.

4.3. KFDS method with three velocity model

To derive the finite volume scheme for the three velocity based DVBE, we follow the same steps as in the previous section and obtain the update formula at kinetic level to be same as that of two velocity DVBE scheme. The definitions of Λ are as given below.

$$\mathbf{\Lambda}_i = \begin{bmatrix} \lambda^+ & 0 & 0 \\ 0 & \lambda_o & 0 \\ 0 & 0 & \lambda^- \end{bmatrix}_i = \begin{bmatrix} \lambda^+ & 0 & 0 \\ 0 & \lambda_o^+ & 0 \\ 0 & 0 & 0 \end{bmatrix}_i + \begin{bmatrix} 0 & 0 & 0 \\ 0 & \lambda_o^- & 0 \\ 0 & 0 & \lambda^- \end{bmatrix}_i = \mathbf{\Lambda}_i^+ + \mathbf{\Lambda}_i^- \tag{63}$$

Again it is possible to define $|\mathbf{\Lambda}|$ in a similar way as

$$|\mathbf{\Lambda}|_\mathbf{i} = \mathbf{\Lambda}_\mathbf{i}^+ - \mathbf{\Lambda}_\mathbf{i}^- \tag{64}$$

Following the same finite volume procedure and applying it to (46), together with (63), (64) and taking moments using P as given in (45), we get

$$\left\{ G(U)^n_{j+\frac{1}{2}} = \frac{1}{2}[G(U)^n_{j+1} + G(U)^n_j] - \frac{1}{2}[\Delta G(U)^{+,n}_{j+\frac{1}{2}} - \Delta G(U)^{-,n}_{j+\frac{1}{2}}] \right\}_i \tag{65}$$

$$\left\{ G(U)^n_{j-\frac{1}{2}} = \frac{1}{2}[G(U)^n_j + G(U)^n_{j-1}] - \frac{1}{2}[\Delta G(U)^{+,n}_{j-\frac{1}{2}} - \Delta G(U)^{-,n}_{j-\frac{1}{2}}] \right\}_i \tag{66}$$

$$\left\{ \begin{aligned} &\Delta G(U)^{\pm}_{j+\frac{1}{2}} = \frac{1}{2}[G(U)_{j+1} - G(U)_j] \pm \frac{1}{2}|\lambda|[U_{j+1} - U_j] \\ &\mp \frac{1}{2}(\lambda - |\lambda_o|)[f^o_{eq\,j+1} - f^o_{eq\,j}] \end{aligned} \right\}_i \tag{67}$$

$$\left\{ \begin{aligned} &\Delta G(U)^{\pm}_{j-\frac{1}{2}} = \frac{1}{2}[G(U)_j - G(U)_{j-1}] \pm \frac{1}{2}|\lambda|[U_j - U_{j-1}] \\ &\mp \frac{1}{2}(\lambda - |\lambda_o|)[f^o_{eq\,j} - f^o_{eq\,j-1}] \end{aligned} \right\}_i \tag{68}$$

The split flux differences in the interface flux for the three velocity DVBE model based KFDS derived in (67) have taken an interesting form in comparison with those in the two velocity DVBE model based KFDS derived in (58). We can write

$$\left\{ G(U)^{3V-KFDS}_{j\pm\frac{1}{2}} \right\}_i = \left\{ G(U)^{2V-KFDS}_{j\pm\frac{1}{2}} \mp \frac{1}{2}(\lambda - |\lambda_o|)(\Delta f^o_{eq}) \right\}_i \tag{69}$$

wherein the interface flux for 3-Velocity Model is essentially that of 2-Velocity Model with additional terms, which are just functions of λ_0 and f^0_{eq}. In the next subsection, λ, which represents the essential contribution to the coefficient of numerical diffusion of the scheme, is fixed in such a way that numerical diffusion vanishes for a steady discontinuity, leading to the enforcement of R–H conditions, as described in the 2-velocity model. λ_0 and f^0_{eq} are chosen in such a way that entropy condition violation is avoided and hence are activated only in the expansion regions. This happens automatically in the formulation, based on the utilization of relative entropy or the D^2-distance. Since λ_0 and f^0_{eq} are activated only in the smooth regions, their contribution to the numerical diffusion in the scheme will be low. Here, for simplicity, we reduce the unknowns (λ_0 and f^0_{eq}) from two to one, by expressing one in terms of the other. Thus, to determine f^o_{eq} for each i, we equate the diffusion flux term of Eq. (67) to zero.

$$\frac{1}{2}|\lambda|\Delta U_{j+\frac{1}{2}} - \frac{1}{2}(\lambda - |\lambda_o|)\Delta f^o_{eq\,j+\frac{1}{2}} = 0 \tag{70}$$

$$or \; \Delta f^o_{eq\,j+\frac{1}{2}} = \frac{|\lambda|}{\lambda - |\lambda_o|} \Delta U_{j+\frac{1}{2}} \tag{71}$$

$$\text{Thus we infer, } f^o_{eq\,i} = \frac{|\lambda|_i}{\lambda_i - |\lambda_o|_i} U_i \tag{72}$$

For convenience, together with f^o_{eq}, a coefficient which enables a control on the additional diffusion flux is introduced. Thus we fix f^o_{eq} for each i as

$$f^o_{eq\,i} = \left(\frac{k|\lambda|}{\lambda - |\lambda_o|} U \right)_i = \left(\frac{\lambda_A}{\lambda - |\lambda_o|} U \right)_i \tag{73}$$

where λ_A is the wave speed for the additional diffusion term, which is required to avoid any possible violation of entropy condition. The flux differences in (67) can be rewritten as

$$\left\{ \Delta G(U)^{\pm}_{j+\frac{1}{2}} = \frac{1}{2}[G(U)^n_{j+1} - G(U)^n_j] \pm \frac{1}{2}|\lambda|[U_{j+1} - U_j] \mp \frac{1}{2}\lambda_A[U_{j+1} - U_j] \right\}_i \tag{74}$$

$$\left\{ \Delta G(U)^{\pm}_{j-\frac{1}{2}} = \frac{1}{2}[G(U)^n_j - G(U)^n_{j-1}] \pm \frac{1}{2}|\lambda|[U_j - U_{j-1}] \mp \frac{1}{2}\lambda_A[U_j - U_{j-1}] \right\}_i \tag{75}$$

It can be observed from the above expressions that two parameters, λ and λ_A are now required to be fixed, to control numerical diffusion. Out of these two, λ is assigned the role of controlling the primary numerical diffusion to capture steady discontinuities exactly, which is described in the next subsection. λ_A is assigned the role of introducing additional numerical diffusion to avoid any possible entropy condition violation, which is described in a later subsection.

4.4. Fixing the primary numerical diffusion through λ

In the construction of the KFDS scheme, fixing of the coefficients λ and λ_A will be critical in determining the fundamental capabilities of the scheme. As mentioned earlier, the idea behind introducing λ_A is to provide a means to introduce additional diffusion in specific regions of expansion in order to prevent the entropy violation which is typical of low diffusion schemes. The basic definition of λ comes with the assumption that λ_A is zero. λ is fixed to obtain a steady shock with zero numerical diffusion. For a steady shock to be captured exactly, when the left and right fluxes are equal the numerical diffusion must vanish, according to Harten, Lax and van Leer [6]. This *flux equivalence across a steady shock* is the consequence of flux conservation. Thus, in the expression for the cell-interface flux $G_{j+\frac{1}{2}}$ for each i, we can take $G_j = G_{j+\frac{1}{2}} = G_{j+1}$ to obtain an expression for λ. First we substitute $G_{j+\frac{1}{2}} = G_j$. Then, we get

$$|\lambda_i|\Delta U_i = \left(G_{j+1} - G_j\right)_i = \Delta G_i \tag{76}$$

Similarly, $G_{j+\frac{1}{2}} = G_{j+1}$ leads to

$$|\lambda|_i \Delta U_i = -\Delta G_i \tag{77}$$

Generalizing from both the above expressions, we write

$$|\lambda|_i = \left|\frac{\Delta G_i}{\Delta U_i}\right| \tag{78}$$

Therefore, the values of $|\lambda|$ for the continuity, momentum and energy equations in a one dimensional flow will take the form as given below.

$$|\lambda|_1 = \left|\frac{\Delta(\rho u)}{\Delta \rho}\right|; \quad |\lambda|_2 = \left|\frac{\Delta(p+\rho u^2)}{\Delta(\rho u)}\right|; \quad |\lambda|_3 = \left|\frac{\Delta(pu+\rho u E)}{\Delta(\rho E)}\right| \tag{79}$$

As the coefficient of numerical diffusion is a function of both the left and right states across a cell-interface, flux difference splitting approach chosen is appropriate. In order to avoid numerical overflow, $|\lambda|_i$ is restricted to the minimum eigenvalue when $\Delta U_i \leq \epsilon$ where $\epsilon = 10^{-10}$. For $G_j = G_{j+1}$ or $\Delta G = 0$, the coefficient of diffusion λ becomes zero, thus capturing the steady shocks and contact discontinuities exactly.

4.5. Need for additional diffusion

The requirement of the additional diffusion arises in smooth regions of the flow. In particular, sonic points can occur in regions of expansions in a flow field and the low diffusive numerical schemes would require an entropy fix to prevent the occurrence of unphysical expansion shocks. Such an entropy fix introduces additional non-zero numerical diffusion near the sonic points. However, if this is done everywhere, the exact shock capturing ability will be lost. It is therefore desirable to retain the low diffusive nature of the basic scheme and add additional numerical diffusion only near the expansive sonic points. The identification of smooth regions, in contrast with the regions of discontinuity, is done in a novel way utilizing the relative entropy or the D^2-distance.

4.5.1. D^2-distance and entropy

Relative entropy, also known as Kullback–Liebler divergence, directed divergence or D^2-distance, is a measure of variation between two distribution functions. This measure and its variants are popularly used in statistics to identify the distinguishing features between two sets of statistical data (discrete samples) or continuous functions. It is worth noting that the form of D^2-distance resembles the Boltzmann H-function and thereby is a comprehensive representation of entropy at the kinetic level. The relative entropy to measure divergence between two distributions is given by (80).

$$D\left(f_k^{eq}, f_j^{eq}\right) = \int_0^{\infty}\int_{-\infty}^{\infty}\left(f_j^{eq} - f_k^{eq}\right)\ln\left(\frac{f_j^{eq}}{f_k^{eq}}\right)dv\,dI \tag{80}$$

Here, $f_{k,j}^{eq}$ represents the equilibrium distribution and k, j represent two locations, which represent in the present framework the left and right state of the interface. Utilizing the definition of classical Maxwellians for the equilibrium distribution, the moment of the above D^2-distance yields [41]

$$D^2\left(f_k^{eq}, f_j^{eq}\right) = \left(\rho_j - \rho_k\right)\ln\left[\frac{\rho_j}{\rho_k}\left(\frac{T_k}{T_j}\right)^{\frac{5}{2}}\right] + \frac{\rho_k}{2RT_k}\left[\frac{\rho_j}{\rho_k} + \frac{T_k}{T_j}\right]\left(u_k - u_j\right)^2 + \frac{5}{2}\left[\frac{\rho_k\left(T_k - T_j\right)}{T_j} + \frac{\rho_j\left(T_j - T_k\right)}{T_k}\right] \tag{81}$$

The thermodynamic entropy used in this work is obtained from gas dynamic relations as

$$S_j = -R\left(\ln\rho_j + \frac{\ln\beta_j}{\gamma - 1} + \text{constant}\right) \tag{82}$$

To evaluate the capabilities of the relative entropy as a sensor function, numerical tests were performed on the exact solutions of several one dimensional Euler test cases. The behaviour of the relative entropy and the corresponding entropy for the some of most representative test cases are shown in Figs. 1 and 2.

It can be seen that the relative entropy gives a positive signal of distinct magnitude for shocks and contact discontinuities. Even in case of expansion waves the function produces a positive signal whose magnitude is one order lower than that of strong shocks and contact discontinuities. It is evident that this function (81) senses gradients arising in pressure, temperature, density and velocity and can be effective in identifying flow gradients and discontinuities of many kinds. While identification of shocks, contact discontinuities or expansions individually is challenging, the D^2-distance along with the entropy function seems to be a good indicator for this task. In the present work, the D^2-distance along with the entropy function is used to identify expansion regions and we introduce additional diffusion only at these regions to avoid unphysical expansion shocks. The rendering of the discrete version of relative entropy to fit the discrete Boltzmann system presented here, with the study of the required mathematical properties, is beyond the scope of this paper and will be pursued elsewhere.

4.6. Fixing the additional numerical diffusion through λ_A

For fixing the additional numerical diffusion, λ_A is defined as follows.

$$\lambda_A = \begin{cases} \lambda_{DD} \text{ for } D^2 > 0 \quad \& \quad \Delta S = 0 \\ 0 \quad \text{Otherwise} \end{cases} \tag{83}$$

where D^2 refers to the relative entropy or D^2-distance and ΔS is the estimated difference in thermodynamic entropy across the cell interface. Two possibilities are presented here for fixing λ_{DD}.

4.6.1. Maximum eigenvalue based λ_{DD}

A simple and yet robust strategy to capture expansion region without violating entropy conditions is to fix the coefficient of numerical diffusion as the maximum of the eigenvalues, as in Rusanov of LLF method. It is done only at expansions, as it would yet be possible retain the steady discontinuity capturing ability of the scheme. Thus the first possible definition of λ_{DD} will be

$$\lambda_{DD} = \lambda_{max} - \lambda \tag{84}$$

$$\text{where } \lambda_{\max} = \max\left\{\max(|u+a|, |u|, |u-a|)_L, \max(|u+a|, |u|, |u-a|)_R\right\} \tag{85}$$

This makes the coefficient of the total numerical diffusion as λ_{max} at the expansions.

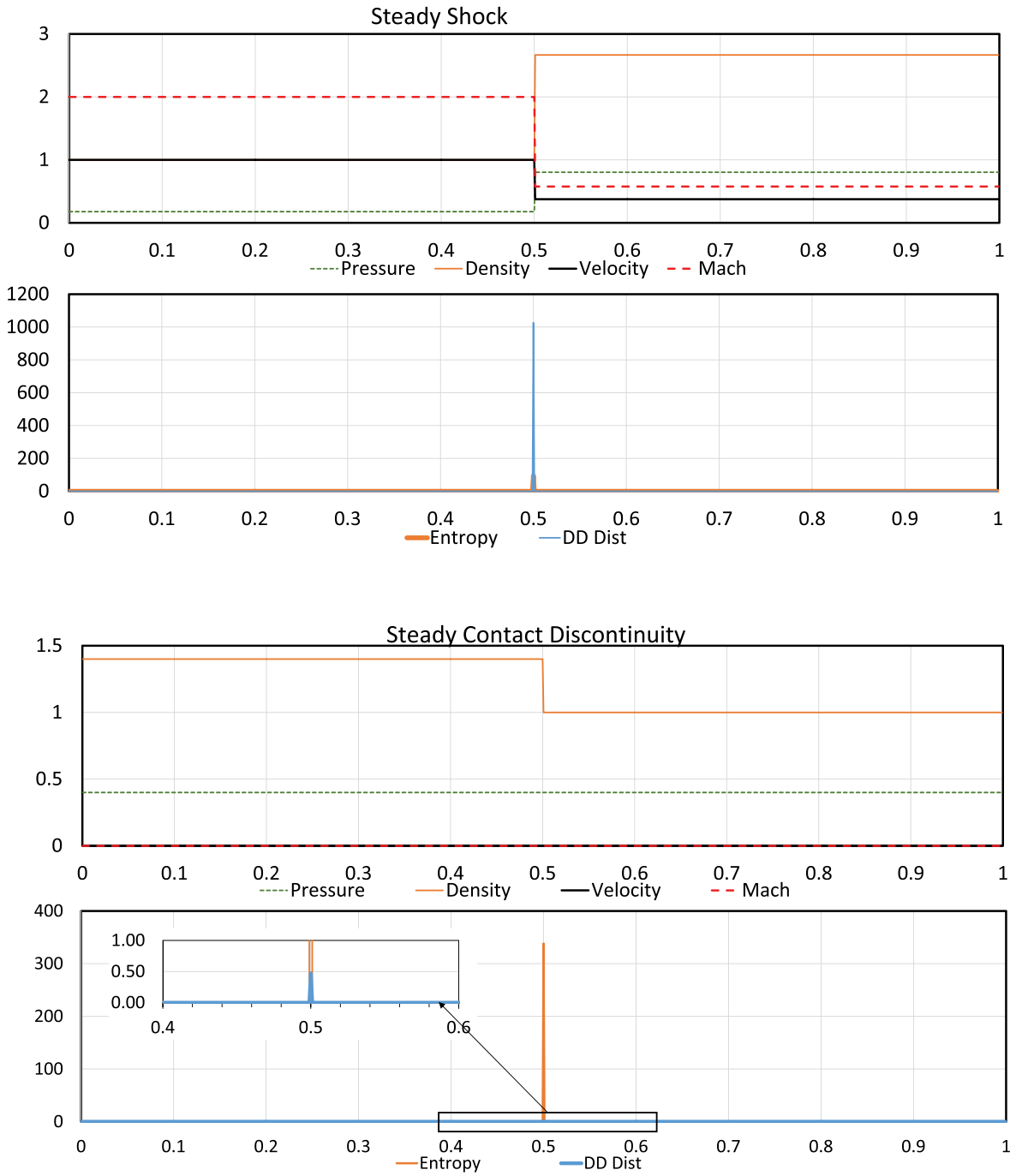

Fig. 1. Entropy and D^2-distance (relative entropy) for steady shock and steady contact discontinuity.

4.6.2. *Fluid velocity based* λ_{DD}

Another choice for λ_A is to select a coefficient of numerical diffusion based on normal fluid velocity across the cell interface. Raghavendra [27], Ramesh [42] and Ramesh et al. [43] have demonstrated that the numerical diffusion can be fixed based on Riemann invariants and have reported stable capturing of the expansion regions without the need for an entropy fix. This approach automatically leads to the normal fluid velocity as the coefficient of numerical diffusion. We define

$$\lambda_{DD} = \lambda_{RI} - \lambda \tag{86}$$

$$\text{where } \lambda_{RI} = \frac{|u_L| + |u_R|}{2} \tag{87}$$

This makes the coefficient of total numerical diffusion in the scheme as λ_{RI} in expansions.

4.6.3. *Activation strategy of additional numerical diffusion*

To evaluate the efficiency of (83), preliminary numerical tests were performed on shock tube problems. The capability of the basic scheme with no additional diffusion is presented in Table 1 for comparison.

Upon introducing additional diffusion in regions where directed divergence is nonzero, the scheme is stable but loses its ability to capture steady shocks and steady contact discontinuities exactly. When the condition in (83) is strictly enforced, the diffusion required for the solution to evolve in the regions of shocks and contact discontinuities is found to be not adequate. As a result, expansion shocks appear in SOD test case and the solution blows up for Toro case 5. It is observed that

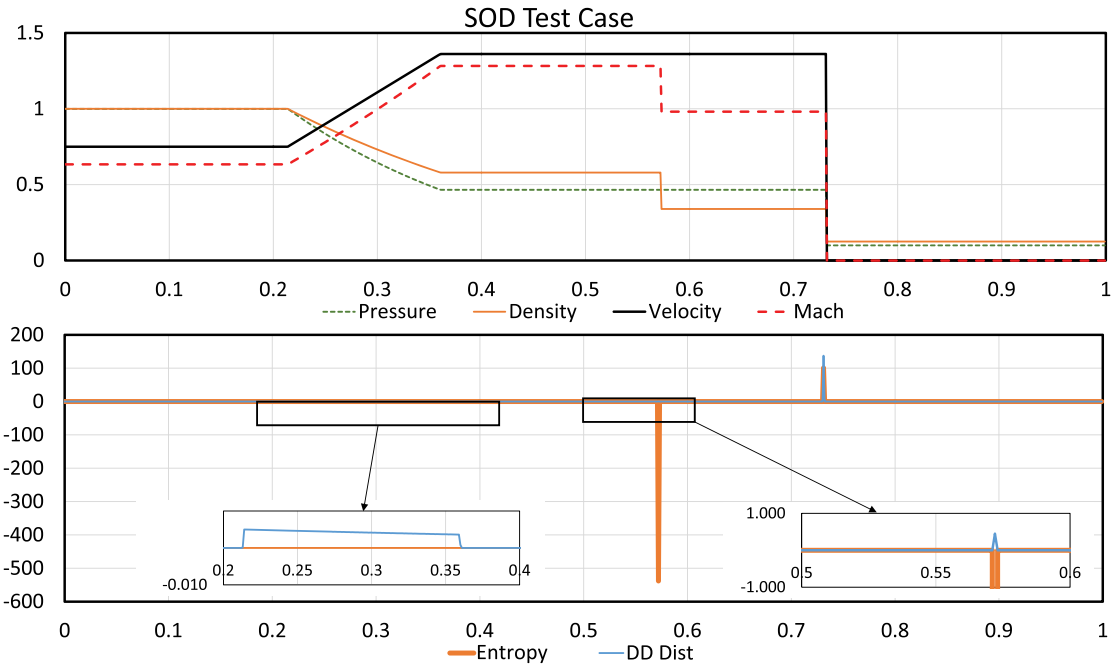

Fig. 2. Entropy and D^2-distance (relative entropy) for SOD test case.

Table 1
Numerical experiments to deduce activation strategy for additional numerical diffusion.

TEST CASE	KFDS No additional Diffusion	KFDS + DD > 0	KFDS + DD > 0 $+\Delta S = 0$	KFDS + DD > 0 $+\Delta S \leq S_{max}$	KFDS + DD > 0 $+\Delta S \leq S_{mean}$
Steady C.D.	✔ Exact	✔ Diffused	✔ Exact	✔ Exact	✔ Exact
Steady shock	✔ Exact	✔ Diffused	✔ Exact	✔ Diffused	✔ Exact
SOD case	✖ Expansion shocks	✔ Smooth	✖ Expansion shocks	✔ Smooth	✔ Smooth
Overheating	✖ Blows up	✔ Smooth	✔ Smooth	✔ Smooth	✔ Smooth
Toro Case 3	✔ Not smooth	✔ Smooth	✔ Not smooth	✔ Smooth	✔ Smooth
Shock collision	✔ Not smooth	✔ Smooth	✔ Not smooth	✔ Smooth	✔ Smooth
Toro Case 5	✖ Blows up	✔ Smooth	✖ Blows up	✔ Smooth	✔ Smooth

additional diffusion is required in regions of both shocks and expansions in the unsteady test cases for the solution to evolve. This means that the condition $\Delta S = 0$ has to be revised to $\Delta S \leq$ a parameter. The numerical experiments further confirmed that the parameter cannot be a constant and its magnitude scales up according to the maximum energy/entropy level of the system. Therefore, the parameter has to be a function of the primitive variables defining the system. After careful numerical experiments, maximum entropy and mean entropy of the system were identified as the limits, up to which entropy change (ΔS) is acceptable to permit additional diffusion. While both the limits provided positive solutions, the use of mean entropy limit for regulating additional diffusion zone is capable of retaining the exact steady shock and steady contact discontinuity capturing ability of the numerical scheme. Therefore, the condition to introduce additional diffusion as given in (83) is revised to the following.

$$\lambda_A = \begin{cases} \lambda_{DD} \text{ for } D^2 > 0 \quad \& \quad \Delta S \leq S_{mean} \\ 0 \quad \text{Otherwise} \end{cases} \tag{88}$$

It is clear that the additional numerical diffusion to get all the typical and different test cases working well is quite sensitive to the involved parameters, a fact well-known to the macroscopic CFD algorithm developers. The optimal numerical diffusion to avoid entropy violation in smooth regions still remains an active research topic.

4.7. Final expressions for KFDS scheme

The final form of the numerical scheme is given as

$$\left\{ U_j^{n+1} = U_j^n - \frac{\Delta t}{\Delta x} \left[G(U)^n_{j+\frac{1}{2}} - G(U)^n_{j-\frac{1}{2}} \right] \right\}_i \tag{89}$$

where the interface fluxes are given by

$$\left\{ G(U)^n_{j+\frac{1}{2}} = \frac{1}{2}[G(U)^n_{j+1} + G(U)^n_j] - \frac{1}{2}[\Delta G(U)^{+,n}_{j+\frac{1}{2}} - \Delta G(U)^{-,n}_{j+\frac{1}{2}}] \right\}_i \tag{90}$$

$$\left\{ G(U)^n_{j-\frac{1}{2}} = \frac{1}{2}[G(U)^n_j + G(U)^n_{j-1}] - \frac{1}{2}[\Delta G(U)^{+,n}_{j-\frac{1}{2}} - \Delta G(U)^{-,n}_{j-\frac{1}{2}}] \right\}_i \tag{91}$$

The split flux differences are given by

$$\left\{ \Delta G(U)^{\pm}_{j+\frac{1}{2}} = \frac{1}{2}[G(U)^n_{j+1} - G(U)^n_j] \pm \frac{1}{2}|\lambda|[U_{j+1} - U_j] \mp \frac{1}{2}\lambda_A[U_{j+1} - U_j] \right\}_i \tag{92}$$

$$\left\{ \Delta G(U)^{\pm}_{j-\frac{1}{2}} = \frac{1}{2}[G(U)^n_j - G(U)^n_{j-1}] \pm \frac{1}{2}|\lambda|[U_j - U_{j-1}] \mp \frac{1}{2}\lambda_A[U_j - U_{j-1}] \right\}_i \tag{93}$$

where $|\lambda_i| = \left|\frac{\Delta G(U)_i}{\Delta U_i}\right|$ and λ_A for each i is defined by

$$\lambda_A = \begin{cases} \lambda_{DD} \text{ for } \quad D^2 > 0 \quad \& \quad \Delta S \le S_{mean} \\ 0 \quad \text{Otherwise} \end{cases} \tag{94}$$

λ_A is determined by the choice of the λ_{DD} and correspondingly the scheme will be named as KFDS-A for the use of maximum eigenvalue based wave speed and KFDS-B for the use of normal fluid velocity based wave speed.

4.8. Higher order accuracy

There are many ways to obtain higher order accuracy [44–47]. One of the simplest of the ways is to reconstruct the primitive or the conserved variables by a linear combination of the solution variables from the appropriate neighbouring cells. It is important to note that a mere linear combination can result in spurious oscillations and can lead to unphysical results. Thus such linear combinations are generally accompanied with moderation of the slopes or fluxes using limiter functions. In this paper, second order accuracy is obtained by means of a piece-wise linear reconstruction of the distribution functions from neighbouring cells, at a cell-interface. We take the same approach as in Section 4.3 for discrete kinetic upwinding, with linear reconstructions of the distributions. Although this section is presented without suffix i for simplicity, each of the expressions does have a valid suffix i, representing the respective conservation equation.

$$\mathbf{h}^n_{j+\frac{1}{2}} = \left[\left(\Lambda^+\mathbf{f}_{eq}\right)_L + \left(\Lambda^-\mathbf{f}_{eq}\right)_R\right]_{j+\frac{1}{2}} \tag{95}$$

$$\mathbf{h}^n_{j-\frac{1}{2}} = \left[\left(\Lambda^+\mathbf{f}_{eq}\right)_L + \left(\Lambda^-\mathbf{f}_{eq}\right)_R\right]_{j-\frac{1}{2}} \tag{96}$$

In order to attain second order accuracy, we do a piece-wise linear reconstruction of the conserved variable (in this case, the distribution function f) to obtain the equivalent left and right state variables for the given cell-interface.

$$\mathbf{f}_j(x, t^n) = \left[\mathbf{f}^n_j + \left(\frac{\partial \mathbf{f}}{\partial x}\right)_j (x - x_j)\right] \tag{97}$$

The slope can be limited by a minmod limiter to obtain a non-oscillatory behaviour.

$$\left(\frac{\partial \mathbf{f}}{\partial x}\right)_j = \mathbf{minmod}\left[\frac{\mathbf{f}_{j+1} - \mathbf{f}_j}{\Delta x}, \frac{\mathbf{f}_j - \mathbf{f}_{j-1}}{\Delta x}\right] \tag{98}$$

Using such piece-wise linear reconstructions at appropriate locations, the left and right states $\mathbf{f}_L$ and $\mathbf{f}_R$ at the cell-interface $I = j + \frac{1}{2}$ turn out to be

$$\mathbf{f}_{R,j+\frac{1}{2}} = \mathbf{f}_{j+1} - \frac{1}{2}\mathbf{minmod}\left[(\mathbf{f}_{j+2} - \mathbf{f}_{j+1}), (\mathbf{f}_{j+1} - \mathbf{f}_j)\right] \tag{99}$$

$$\mathbf{f}_{L,j+\frac{1}{2}} = \mathbf{f}_j + \frac{1}{2}\mathbf{minmod}\left[(\mathbf{f}_{j+1} - \mathbf{f}_j), (\mathbf{f}_j - \mathbf{f}_{j-1})\right] \tag{100}$$

Therefore the interface flux in the finite volume update formula can be written in flux difference split form as

$$\mathbf{h}^n_{j+\frac{1}{2}} = \frac{1}{2}\left[\mathbf{h}^n_R + \mathbf{h}^n_L\right]_{j+\frac{1}{2}} - \frac{1}{2}[\Delta\mathbf{h}^+_{j+\frac{1}{2}} - \Delta\mathbf{h}^-_{j+\frac{1}{2}}] \tag{101}$$

$$\mathbf{h}^n_{j-\frac{1}{2}} = \frac{1}{2}\left[\mathbf{h}^n_R + \mathbf{h}^n_L\right]_{j-\frac{1}{2}} - \frac{1}{2}[\Delta\mathbf{h}^+_{j-\frac{1}{2}} - \Delta\mathbf{h}^-_{j-\frac{1}{2}}] \tag{102}$$

where

$$\Delta\mathbf{h}^{\pm}_{j\pm\frac{1}{2}} = [\Lambda^{\pm}(\mathbf{f}_R - \mathbf{f}_L)]_{j\pm\frac{1}{2}} \tag{103}$$

Let us evaluate the moments of the above equations to recover the corresponding macroscopic update formula. Eq. (46) gives

$$U^{n+1}_j = U^n_j - \frac{\Delta t}{\Delta x}\left[G(U)^n_{j+\frac{1}{2}} - G(U)^n_{j-\frac{1}{2}}\right] \tag{104}$$

where the interface fluxes are obtained by taking moments of (101).

$$\begin{aligned} \mathbf{Ph}_{j+\frac{1}{2}} &= G(U)_{j+\frac{1}{2}} \\ &= \frac{1}{2}\left[G(U)_R + G(U)_L\right]_{j+\frac{1}{2}} - \frac{1}{2}[\Delta G(U)^+_{j+\frac{1}{2}} - \Delta G(U)^-_{j+\frac{1}{2}}] \\ &= \frac{1}{2}\left[G(U)_j + G(U)_{j+1}\right] \\ &-\frac{1}{4}\mathbf{minmod}\left[(G(U)_{j+2} - G(U)_{j+1}), (G(U)_{j+1} - G(U)_j)\right] \\ &+\frac{1}{4}\mathbf{minmod}\left[(G(U)_{j+1} - G(U)_j), (G(U)_j - G(U)_{j-1})\right] \\ &-\frac{1}{2}[\Delta G(U)^+_{j+\frac{1}{2}} - \Delta G(U)^-_{j+\frac{1}{2}}] \end{aligned} \tag{105}$$

Similarly

$$\begin{aligned} \mathbf{Ph}_{j-\frac{1}{2}} &= G(U)_{j-\frac{1}{2}} \\ &= \frac{1}{2}\left[G(U)_R + G(U)_L\right]_{j-\frac{1}{2}} - \frac{1}{2}[\Delta G(U)^+_{j-\frac{1}{2}} - \Delta G(U)^-_{j-\frac{1}{2}}] \\ &= \frac{1}{2}\left[G(U)_{j-1} + G(U)_j\right] \\ &-\frac{1}{4}\mathbf{minmod}\left[(G(U)_{j+1} - G(U)_j), (G(U)_j - G(U)_{j-1})\right] \\ &+\frac{1}{4}\mathbf{minmod}\left[(G(U)_j - G(U)_{j-1}), (G(U)_{j-1} - G(U)_{j-2})\right] \\ &-\frac{1}{2}[\Delta G(U)^+_{j-\frac{1}{2}} - \Delta G(U)^-_{j-\frac{1}{2}}] \end{aligned} \tag{106}$$

Here, the split flux differences, namely $\Delta G(U)^{\pm}_{j\pm\frac{1}{2}}$, can be evaluated by taking moments of (103).

$$\begin{aligned} \mathbf{P}\Delta\mathbf{h}^+_{j+\frac{1}{2}} &= \Delta G(U)^+_{j+\frac{1}{2}} \\ &= \mathbf{P}[\Lambda^+(\mathbf{f}_R - \mathbf{f}_L)]_{j+\frac{1}{2}} \\ &= \mathbf{P}\Lambda^+\left[\mathbf{f}_{j+1} - \frac{1}{2}\mathbf{minmod}\left[(\mathbf{f}_{j+2} - \mathbf{f}_{j+1}), (\mathbf{f}_{j+1} - \mathbf{f}_j)\right]\right] \\ &- \mathbf{P}\Lambda^+\left[\mathbf{f}_j + \frac{1}{2}\mathbf{minmod}\left[(\mathbf{f}_{j+1} - \mathbf{f}_j), (\mathbf{f}_j - \mathbf{f}_{j-1})\right]\right] \end{aligned} \tag{107}$$

Upon expanding and rearranging, we get

$$\begin{aligned} \Delta G(U)^+_{j+\frac{1}{2}} &= \frac{1}{2}\left(G(U)_{j+1} - G(U)_j\right) + \frac{1}{2}|\lambda|\left(U_{j+1} - U_j\right) \\ &-\frac{(\lambda - |\lambda_o|)}{2}\left(\mathbf{f}^o_{eq,j+1} - \mathbf{f}^o_{eq,j}\right) \\ &-\frac{1}{2}\mathbf{minmod}\left[\begin{array}{c} \left\{\frac{1}{2}\left(G(U)_{j+2} - G(U)_{j+1}\right) + \frac{1}{2}|\lambda|\left(U_{j+2} - U_{j+1}\right)\right. \\ \left. -\frac{(\lambda - |\lambda_o|)}{2}\left(\mathbf{f}^o_{eq,j+2} - \mathbf{f}^o_{eq,j+1}\right)\right\}, \\ \left\{\frac{1}{2}\left(G(U)_{j+1} - G(U)_j\right) + \frac{1}{2}|\lambda|\left(U_{j+1} - U_j\right)\right. \\ \left. -\frac{(\lambda - |\lambda_o|)}{2}\left(\mathbf{f}^o_{eq,j+1} - \mathbf{f}^o_{eq,j}\right)\right\} \end{array}\right] \\ &-\frac{1}{2}\mathbf{minmod}\left[\begin{array}{c} \left\{\frac{1}{2}\left(G(U)_{j+1} - G(U)_j\right) + \frac{1}{2}|\lambda|\left(U_{j+1} - U_j\right)\right. \\ \left. -\frac{(\lambda - |\lambda_o|)}{2}\left(\mathbf{f}^o_{eq,j+1} - \mathbf{f}^o_{eq,j}\right)\right\}, \\ \left\{\frac{1}{2}\left(G(U)_j - G(U)_{j-1}\right) + \frac{1}{2}|\lambda|\left(U_j - U_{j-1}\right)\right. \\ \left. -\frac{(\lambda - |\lambda_o|)}{2}\left(\mathbf{f}^o_{eq,j} - \mathbf{f}^o_{eq,j-1}\right)\right\} \end{array}\right] \end{aligned}$$

Similarly the $\Delta G(U)^{-}_{j+\frac{1}{2}}$ term can be obtained as

$$\begin{aligned}\Delta G(U)^{-}_{j+\frac{1}{2}} &= \frac{1}{2}\left(G(U)_{j+1}-G(U)_j\right)-\frac{1}{2}|\lambda|\left(U_{j+1}-U_j\right)\\ &+\frac{(\lambda-|\lambda_o|)}{2}\left(\mathbf{f}^o_{eq,j+1}-\mathbf{f}^o_{eq,j}\right)\\ &-\frac{1}{2}\mathbf{minmod}\left[\begin{array}{c}\left\{\frac{1}{2}\left(G(U)_{j+2}-G(U)_{j+1}\right)-\frac{1}{2}|\lambda|\left(U_{j+2}-U_{j+1}\right)\right.\\ \left.+\frac{(\lambda-|\lambda_o|)}{2}\left(\mathbf{f}^o_{eq,j+2}-\mathbf{f}^o_{eq,j+1}\right)\right\},\\ \left\{\frac{1}{2}\left(G(U)_{j+1}-G(U)_j\right)-\frac{1}{2}|\lambda|\left(U_{j+1}-U_j\right)\right.\\ \left.+\frac{(\lambda-|\lambda_o|)}{2}\left(\mathbf{f}^o_{eq,j+1}-\mathbf{f}^o_{eq,j}\right)\right\}\end{array}\right]\\ &-\frac{1}{2}\mathbf{minmod}\left[\begin{array}{c}\left\{\frac{1}{2}\left(G(U)_{j+1}-G(U)_j\right)-\frac{1}{2}|\lambda|\left(U_{j+1}-U_j\right)\right.\\ \left.+\frac{(\lambda-|\lambda_o|)}{2}\left(\mathbf{f}^o_{eq,j+1}-\mathbf{f}^o_{eq,j}\right)\right\},\\ \left\{\frac{1}{2}\left(G(U)_j-G(U)_{j-1}\right)-\frac{1}{2}|\lambda|\left(U_j-U_{j-1}\right)\right.\\ \left.+\frac{(\lambda-|\lambda_o|)}{2}\left(\mathbf{f}^o_{eq,j}-\mathbf{f}^o_{eq,j-1}\right)\right\}\end{array}\right]\end{aligned}$$

In the same way, the flux difference split fluxes at $j-\frac{1}{2}$ interface can be obtained as

$$\begin{aligned}\Delta G(U)^{+}_{j-\frac{1}{2}} &= \frac{1}{2}\left(G(U)_j-G(U)_{j-1}\right)+\frac{1}{2}|\lambda|\left(U_j-U_{j-1}\right)\\ &-\frac{(\lambda-|\lambda_o|)}{2}\left(\mathbf{f}^o_{eq,j}-\mathbf{f}^o_{eq,j-1}\right)\\ &-\frac{1}{2}\mathbf{minmod}\left[\begin{array}{c}\left\{\frac{1}{2}\left(G(U)_{j+1}-G(U)_j\right)+\frac{1}{2}|\lambda|\left(U_{j+1}-U_j\right)\right.\\ \left.-\frac{(\lambda-|\lambda_o|)}{2}\left(\mathbf{f}^o_{eq,j+1}-\mathbf{f}^o_{eq,j}\right)\right\},\\ \left\{\frac{1}{2}\left(G(U)_j-G(U)_{j-1}\right)+\frac{1}{2}|\lambda|\left(U_j-U_{j-1}\right)\right.\\ \left.-\frac{(\lambda-|\lambda_o|)}{2}\left(\mathbf{f}^o_{eq,j}-\mathbf{f}^o_{eq,j-1}\right)\right\}\end{array}\right]\\ &-\frac{1}{2}\mathbf{minmod}\left[\begin{array}{c}\left\{\frac{1}{2}\left(G(U)_j-G(U)_{j-1}\right)+\frac{1}{2}|\lambda|\left(U_j-U_{j-1}\right)\right.\\ \left.-\frac{(\lambda-|\lambda_o|)}{2}\left(\mathbf{f}^o_{eq,j}-\mathbf{f}^o_{eq,j-1}\right)\right\},\\ \left\{\frac{1}{2}\left(G(U)_{j-1}-G(U)_{j-2}\right)+\frac{1}{2}|\lambda|\left(U_{j-1}-U_{j-2}\right)\right.\\ \left.-\frac{(\lambda-|\lambda_o|)}{2}\left(\mathbf{f}^o_{eq,j-1}-\mathbf{f}^o_{eq,j-2}\right)\right\}\end{array}\right]\end{aligned}$$

$$\begin{aligned}\Delta G(U)^{-}_{j-\frac{1}{2}} &= \frac{1}{2}\left(G(U)_j-G(U)_{j-1}\right)-\frac{1}{2}|\lambda|\left(U_j-U_{j-1}\right)\\ &+\frac{(\lambda-|\lambda_o|)}{2}\left(\mathbf{f}^o_{eq,j}-\mathbf{f}^o_{eq,j-1}\right)\\ &-\frac{1}{2}\mathbf{minmod}\left[\begin{array}{c}\left\{\frac{1}{2}\left(G(U)_{j+1}-G(U)_j\right)-\frac{1}{2}|\lambda|\left(U_{j+1}-U_j\right)\right.\\ \left.+\frac{(\lambda-|\lambda_o|)}{2}\left(\mathbf{f}^o_{eq,j+1}-\mathbf{f}^o_{eq,j}\right)\right\},\\ \left\{\frac{1}{2}\left(G(U)_j-G(U)_{j-1}\right)-\frac{1}{2}|\lambda|\left(U_j-U_{j-1}\right)\right.\\ \left.+\frac{(\lambda-|\lambda_o|)}{2}\left(\mathbf{f}^o_{eq,j}-\mathbf{f}^o_{eq,j-1}\right)\right\}\end{array}\right]\\ &-\frac{1}{2}\mathbf{minmod}\left[\begin{array}{c}\left\{\frac{1}{2}\left(G(U)_j-G(U)_{j-1}\right)-\frac{1}{2}|\lambda|\left(U_j-U_{j-1}\right)\right.\\ \left.+\frac{(\lambda-|\lambda_o|)}{2}\left(\mathbf{f}^o_{eq,j}-\mathbf{f}^o_{eq,j-1}\right)\right\},\\ \left\{\frac{1}{2}\left(G(U)_{j-1}-G(U)_{j-2}\right)-\frac{1}{2}|\lambda|\left(U_{j-1}-U_{j-2}\right)\right.\\ \left.+\frac{(\lambda-|\lambda_o|)}{2}\left(\mathbf{f}^o_{eq,j-1}-\mathbf{f}^o_{eq,j-2}\right)\right\}\end{array}\right]\end{aligned}$$

The above set of expressions can be put in a compact form by absorbing the minmod terms in (105) & (106) into the respective flux difference split terms. The resulting second order update formulation can be written as

$$U^{n+1}_j = U^n_j-\frac{\Delta t}{\Delta x}\left[G(U)^n_{j+\frac{1}{2}}-G(U)^n_{j-\frac{1}{2}}\right] \tag{108}$$

where in the interface flux are computed using

$$G(U)_{j+\frac{1}{2}} = \frac{1}{2}\left[G(U)_j+G(U)_{j+1}\right]-\frac{1}{2}[\mathbf{\Delta G(U)}^{+}_{j+\frac{1}{2}}-\mathbf{\Delta G(U)}^{-}_{j+\frac{1}{2}}] \tag{109}$$

$$G(U)_{j-\frac{1}{2}} = \frac{1}{2}\left[G(U)_{j-1}+G(U)_j\right]-\frac{1}{2}[\mathbf{\Delta G(U)}^{+}_{j-\frac{1}{2}}-\mathbf{\Delta G(U)}^{-}_{j-\frac{1}{2}}] \tag{110}$$

Using the definition of f^o_{eq}, the split flux differences in the above equation get redefined as

$$\begin{aligned}\mathbf{\Delta G(U)}^{+}_{j+\frac{1}{2}} &= \frac{1}{2}\left(G(U)_{j+1}-G(U)_j\right)\\ &+\frac{1}{2}|\lambda|\left(U_{j+1}-U_j\right)-\frac{1}{2}\lambda_A\left(U_{j+1}-U_j\right)\\ &+\frac{1}{2}\mathbf{minmod}\left[(G(U)_{j+2}-G(U)_{j+1}),(G(U)_{j+1}-G(U)_j)\right]\\ &-\frac{1}{2}\mathbf{minmod}\left[\begin{array}{c}\left\{\frac{1}{2}\left(G(U)_{j+2}-G(U)_{j+1}\right)+\frac{1}{2}|\lambda|\left(U_{j+2}-U_{j+1}\right)\right.\\ \left.-\frac{1}{2}\lambda_A\left(U_{j+2}-U_{j+1}\right)\right\},\\ \left\{\frac{1}{2}\left(G(U)_{j+1}-G(U)_j\right)+\frac{1}{2}|\lambda|\left(U_{j+1}-U_j\right)\right.\\ \left.-\frac{1}{2}\lambda_A\left(U_{j+1}-U_j\right)\right\}\end{array}\right]\\ &-\frac{1}{2}\mathbf{minmod}\left[\begin{array}{c}\left\{\frac{1}{2}\left(G(U)_{j+1}-G(U)_j\right)+\frac{1}{2}|\lambda|\left(U_{j+1}-U_j\right)\right.\\ \left.-\frac{1}{2}\lambda_A\left(U_{j+1}-U_j\right)\right\},\\ \left\{\frac{1}{2}\left(G(U)_j-G(U)_{j-1}\right)+\frac{1}{2}|\lambda|\left(U_j-U_{j-1}\right)\right.\\ \left.-\frac{1}{2}\lambda_A\left(U_j-U_{j-1}\right)\right\}\end{array}\right]\end{aligned} \tag{111}$$

$$\begin{aligned}\mathbf{\Delta G(U)}^{-}_{j+\frac{1}{2}} &= \frac{1}{2}\left(G(U)_{j+1}-G(U)_j\right)-\frac{1}{2}|\lambda|\left(U_{j+1}-U_j\right)\\ &+\frac{1}{2}\lambda_A\left(U_{j+1}-U_j\right)\\ &+\frac{1}{2}\mathbf{minmod}\left[(G(U)_{j+1}-G(U)_j),(G(U)_j-G(U)_{j-1})\right]\\ &-\frac{1}{2}\mathbf{minmod}\left[\begin{array}{c}\left\{\frac{1}{2}\left(G(U)_{j+2}-G(U)_{j+1}\right)-\frac{1}{2}|\lambda|\left(U_{j+2}-U_{j+1}\right)\right.\\ \left.+\frac{1}{2}\lambda_A\left(U_{j+2}-U_{j+1}\right)\right\},\\ \left\{\frac{1}{2}\left(G(U)_{j+1}-G(U)_j\right)-\frac{1}{2}|\lambda|\left(U_{j+1}-U_j\right)\right.\\ \left.+\frac{1}{2}\lambda_A\left(U_{j+1}-U_j\right)\right\}\end{array}\right]\\ &-\frac{1}{2}\mathbf{minmod}\left[\begin{array}{c}\left\{\frac{1}{2}\left(G(U)_{j+1}-G(U)_j\right)-\frac{1}{2}|\lambda|\left(U_{j+1}-U_j\right)\right.\\ \left.+\frac{1}{2}\lambda_A\left(U_{j+1}-U_j\right)\right\},\\ \left\{\frac{1}{2}\left(G(U)_j-G(U)_{j-1}\right)-\frac{1}{2}|\lambda|\left(U_j-U_{j-1}\right)\right.\\ \left.+\frac{1}{2}\lambda_A\left(U_j-U_{j-1}\right)\right\}\end{array}\right]\end{aligned} \tag{112}$$

$$\begin{aligned}\mathbf{\Delta G(U)}^{+}_{j-\frac{1}{2}} &= \frac{1}{2}\left(G(U)_j - G(U)_{j-1}\right) + \frac{1}{2}|\lambda|\left(U_j - U_{j-1}\right) \\ &-\frac{1}{2}\lambda_A\left(U_j - U_{j-1}\right) \\ &+\frac{1}{2}\mathbf{minmod}\left[(G(U)_{j+1} - G(U)_j), (G(U)_j - G(U)_{j-1})\right] \\ &-\frac{1}{2}\mathbf{minmod}\begin{bmatrix}\left\{\frac{1}{2}\left(G(U)_{j+1} - G(U)_j\right) + \frac{1}{2}|\lambda|\left(U_{j+1} - U_j\right) \\ -\frac{1}{2}\lambda_A\left(U_{j+1} - U_j\right)\right\}, \\ \left\{\frac{1}{2}\left(G(U)_j - G(U)_{j-1}\right) + \frac{1}{2}|\lambda|\left(U_j - U_{j-1}\right) \\ -\frac{1}{2}\lambda_A\left(U_j - U_{j-1}\right)\right\}\end{bmatrix} \\ &-\frac{1}{2}\mathbf{minmod}\begin{bmatrix}\left\{\frac{1}{2}\left(G(U)_j - G(U)_{j-1}\right) + \frac{1}{2}|\lambda|\left(U_j - U_{j-1}\right) \\ -\frac{1}{2}\lambda_A\left(U_j - U_{j-1}\right)\right\}, \\ \left\{\frac{1}{2}\left(G(U)_{j-1} - G(U)_{j-2}\right) + \frac{1}{2}|\lambda|\left(U_{j-1} - U_{j-2}\right) \\ -\frac{1}{2}\lambda_A\left(U_{j-1} - U_{j-2}\right)\right\}\end{bmatrix}\end{aligned} \tag{113}$$

$$\begin{aligned}\mathbf{\Delta G(U)}^{-}_{j-\frac{1}{2}} &= \frac{1}{2}\left(G(U)_j - G(U)_{j-1}\right) - \frac{1}{2}|\lambda|\left(U_j - U_{j-1}\right) \\ &+\frac{1}{2}\lambda_A\left(U_j - U_{j-1}\right) \\ &+\frac{1}{2}\mathbf{minmod}\left[(G(U)_j - G(U)_{j-1}), (G(U)_{j-1} - G(U)_{j-2})\right] \\ &-\frac{1}{2}\mathbf{minmod}\begin{bmatrix}\left\{\frac{1}{2}\left(G(U)_{j+1} - G(U)_j\right) - \frac{1}{2}|\lambda|\left(U_{j+1} - U_j\right) \\ +\frac{1}{2}\lambda_A\left(U_{j+1} - U_j\right)\right\}, \\ \left\{\frac{1}{2}\left(G(U)_j - G(U)_{j-1}\right) + \frac{1}{2}|\lambda|\left(U_j - U_{j-1}\right) \\ +\frac{1}{2}\lambda_A\left(U_j - U_{j-1}\right)\right\}\end{bmatrix} \\ &-\frac{1}{2}\mathbf{minmod}\begin{bmatrix}\left\{\frac{1}{2}\left(G(U)_j - G(U)_{j-1}\right) - \frac{1}{2}|\lambda|\left(U_j - U_{j-1}\right) \\ +\frac{1}{2}\lambda_A\left(U_j - U_{j-1}\right)\right\}, \\ \left\{\frac{1}{2}\left(G(U)_{j-1} - G(U)_{j-2}\right) - \frac{1}{2}|\lambda|\left(U_{j-1} - U_{j-2}\right) \\ +\frac{1}{2}\lambda_A\left(U_{j-1} - U_{j-2}\right)\right\}\end{bmatrix}\end{aligned} \tag{114}$$

4.9. KFDS scheme for viscous flows

Consider 1-D Navier–Stokes equations given by

$$\frac{\partial U}{\partial t} + \frac{\partial G(U)}{\partial x} = \frac{\partial G_v(U)}{\partial x} \tag{115}$$

where U is the conserved variable vector, $G(U)$ its nonlinear inviscid flux vector and $G_v(U)$ is the viscous flux vector, given by

$$U = \begin{bmatrix}\rho \\ \rho u \\ \rho E\end{bmatrix},\ G(U) = \begin{bmatrix}\rho u \\ p + \rho u^2 \\ pu + \rho u E\end{bmatrix} \text{ and } G_v(U) = \begin{bmatrix}0 \\ \tau \\ u\tau - q\end{bmatrix} \tag{116}$$

Here, τ is the one dimensional component of the stress tensor and q is the corresponding component of the heat flux vector. μ is the coefficient of fluid viscosity and k is the coefficient of thermal conductivity. The kinetic theory framework for this system of equations as moments, based on the Boltzmann equation with the BGK model, can in a similar way as in (16), be given by

$$\frac{\partial f}{\partial t} + \frac{\partial h}{\partial x} = -\frac{1}{\epsilon}\left[f - f_{CE}\right] \tag{117}$$

and the 1-D Navier–Stokes equations can be recovered by taking moments as

$$\left\langle \Psi\left(\frac{\partial f}{\partial t} + \frac{\partial h}{\partial x} = 0, f = f_{CE}\right)\right\rangle \tag{118}$$

Here, the distribution function instantaneously relaxes to f_{CE}, the Chapman–Enskog distribution function, in the collision step. The derivation of this non-equilibrium distribution and its moments to recover the viscous conservation equations are well documented in [7,37,38]. The Chapman–Enskog distribution function for 1D is given by

$$f_{CE} = f^{eq}(1 + P_{CE}) \tag{119}$$

$$\text{where } \quad f^{eq} = \rho\frac{\sqrt{\beta}}{\sqrt{\pi}}e^{-\beta(v-u)^2}e^{-\frac{I}{I_0}} \tag{120}$$

The perturbation term P_{CE} is given by

$$P_{CE} = \frac{\tau_{CE}}{p}Z_\tau - \frac{q_{CE}}{p\sqrt{2RT}}Z_q \tag{121}$$

where

$$\tau_{CE} = (3-\gamma)\epsilon p\frac{\partial u}{\partial x} \quad and \quad q_{CE} = \frac{\gamma}{\gamma-1}\epsilon p\frac{\partial}{\partial x}\left(\frac{p}{\rho}\right) \tag{122}$$

The coefficients Z_τ and Z_q are given by

$$Z_\tau(v, I) = \frac{3\gamma - 5}{2(3-\gamma)} + \frac{(v-u)^2}{2RT} - \frac{4(\gamma-1)^2}{(3-\gamma)^2}\frac{I}{2RT} \tag{123}$$

$$Z_q(v, I) = \frac{(\gamma-1)}{\gamma RT}\left[\frac{(v-u)^3}{\sqrt{2RT}} - \frac{5}{2}(v-u)\sqrt{2RT} + \frac{4(\gamma-1)}{3-\gamma}\frac{I(v-u)}{\sqrt{2RT}}\right] \tag{124}$$

Thus perturbation term is a function of $(v-u)$ and I with the analogous fluid viscosity and thermal conductivity coefficients. The moments to obtain the macroscopic variables [7] are defined by

$$\begin{aligned}U_i &= \int_0^\infty\int_{-\infty}^\infty \psi_i f_{CE}\ dvdI \text{ and} \\ G_{T,i}(U) &= G(U)_i - G_{v,i}(U) = \int_0^\infty\int_{-\infty}^\infty \psi_i v f_{CE}\ dvdI\end{aligned} \tag{125}$$

with

$$\psi = \begin{bmatrix}1 \\ v \\ I + \frac{1}{2}v^2\end{bmatrix} \text{ and } i = 1, 2, 3 \quad (\text{for } 1-\text{D case}) \tag{126}$$

Introducing a truncated distribution as

$$\tilde{f}_{CE} = \int_0^\infty f_{CE}\ dI \tag{127}$$

we can redefine the moment relations as

$$U_i = \int_{-\infty}^\infty \psi_i \tilde{f}_{CE}\ dv \text{ and } G_{T,i}(U) = \int_{-\infty}^\infty \psi_i v\tilde{f}_{CE}\ dv \tag{128}$$

4.9.1. Prandtl number fix

The KFDS schemes are based on BGK model for the collision terms. The limitation of this model is the Prandtl number being always unity. In order to overcome this limitation, a Prandtl number fix is employed to the heat flux term at the macroscopic level as suggested in [48]. The Prandtl number fix for the viscous flux term in 1D is given by

$$G'_v(U) = G_v(U) + \begin{bmatrix}0 \\ 0 \\ (1-Pr)q\end{bmatrix} \tag{129}$$

where Pr refers to the actual Prandtl number.

4.9.2. Discrete velocity model for viscous flow

To arrive at a discrete velocity model for the above framework, the continuous Chapman–Enskog distribution $\tilde{f}_{CE}$ is approximated by a combination of Dirac delta functions as

$$\psi_i\tilde{f}_{CE} = \left\{f_{CE+}\delta(v-\lambda^+) + f_{CE_o}\delta(v-\lambda_o) + f_{CE-}\delta(v-\lambda^-)\right\}_i \tag{130}$$

Let us further assume, for simplicity, that the discrete velocities, λ^+ and λ^- for each i are given by

$$\lambda_i^+ = \lambda_i \text{ and } \lambda_i^- = -\lambda_i \tag{131}$$

Let us assume that λ^o and $f^{eq}_{CE_o}$ for each i are known (which will be fixed later, as done in the inviscid case). Then, using the two moment relations in (128) we obtain the following.

$$\begin{aligned} U_i &= \int_{-\infty}^{\infty} \psi_i \tilde{f}_{CE} dv \\ &= \int_{-\infty}^{\infty} \left\{ f_{CE_+}\delta(v-\lambda^+) + f_{CE_o}\delta(v-\lambda_o) + f_{CE_-}\delta(v-\lambda^-) \right\}_i dv \\ &= \left\{ f_{CE_+} + f_{CE_o} + f_{CE_-} \right\}_i \end{aligned}$$

or

$$\left\{ f_{CE_+} + f_{CE_-} \right\}_i = U_i - f_{CE_oi} \tag{132}$$

$$\begin{aligned} G_{T,i}(U) &= \int_{-\infty}^{\infty} v\psi_i \tilde{f}_{CE} dv \\ &= \int_{-\infty}^{\infty} v \left\{ f_{CE_+}\delta(v-\lambda^+) + f_{CE_o}\delta(v-\lambda_o) + f_{CE_-}\delta(v-\lambda^-) \right\}_i dv \\ &= \left\{ \begin{aligned} & f_{CE_+} \int_{-\infty}^{\infty} \phi(v)\delta(v-\lambda^+)dv + f_{CE_o} \int_{-\infty}^{\infty} \phi(v)\delta(v-\lambda_o)dv \\ & + f_{CE_-} \int_{-\infty}^{\infty} \phi(v)\delta(v-\lambda^-)dv \end{aligned} \right\}_i, \ (\phi(v) = v) \\ &= \left\{ f_{CE_+}\lambda^+ + f_{CE_o}\lambda_o + f_{CE_-}\lambda^- \right\}_i \end{aligned}$$

or

$$\left\{ f_{CE_+}\lambda^+ + f_{CE_-}\lambda^- \right\}_i = G_{T,i}(U) - \left\{ f_{CE_o}\lambda_o \right\}_i \tag{133}$$

Solving the above two equations and simplifying, we get

$$\begin{aligned} f_{CE+i} &= \frac{1}{2}U_i + \frac{1}{2\lambda_i}G_{T,i}(U) - \left\{ \frac{\lambda+\lambda_o}{2\lambda} f_{CE_o} \right\}_i \text{ and} \\ f_{CE-i} &= \frac{1}{2}U_i - \frac{1}{2\lambda_i}G_{T,i}(U) - \left\{ \frac{\lambda-\lambda_o}{2\lambda} f_{CE_o} \right\}_i \end{aligned} \tag{134}$$

The *Discrete Velocity Boltzmann Equation* (DVBE) for three velocity model thus derived can be written as

$$\left\{ \frac{\partial \mathbf{f}}{\partial t} + \frac{\partial \mathbf{h}}{\partial x} = -\frac{1}{\epsilon}\left[\mathbf{f} - \mathbf{f_{CE}} \right] \right\}_i \quad \text{with} \quad i = 1,2,3 \tag{135}$$

where

$$\mathbf{f_{CEi}} = \begin{bmatrix} f_{CE_+} \\ f_{CE_o} \\ f_{CE_-} \end{bmatrix}_{\mathbf{i}}, \ \mathbf{\Lambda_i} = \begin{bmatrix} \lambda^+ & 0 & 0 \\ 0 & \lambda_o & 0 \\ 0 & 0 & \lambda^- \end{bmatrix}_{\mathbf{i}} \text{ and} \tag{136}$$

$$\mathbf{f_{CEi}} = \begin{bmatrix} f_{CE_+} \\ f_{CE_o} \\ f^{eq}_{CE_-} \end{bmatrix}_{\mathbf{i}} = \begin{bmatrix} \frac{1}{2}U + \frac{1}{2\lambda}\left(G(U) - G_v(U) \right) - \frac{\lambda+\lambda_o}{2\lambda} f_{CE_o} \\ f_{CE_o} \\ \frac{1}{2}U - \frac{1}{2\lambda}\left(G(U) - G_v(U) \right) - \frac{\lambda-\lambda_o}{2\lambda} f_{CE_o} \end{bmatrix}_{\mathbf{i}} \tag{137}$$

It is interesting to note that the discrete velocity model for the inviscid flow is a subset of the model for viscous flow, can be obtained in the limit of viscous fluxes being set to zero and therefore the model is consistent.

4.9.3. KFDS scheme for viscous flows

We follow the same procedures as in Section 4.3 to arrive at the finite volume formulation for viscous flows along with the Prandtl number fix. We get the update formula as

$$\left(U_j^{n+1} = U_j^n - \frac{\Delta t}{\Delta x}\left[G(U)^n_{j+\frac{1}{2}} - G(U)^n_{j-\frac{1}{2}} \right] + \frac{\Delta t}{\Delta x}\left[G_v(U)^n_{j+\frac{1}{2}} - G_v(U)^n_{j-\frac{1}{2}} \right] \right)_i \tag{138}$$

where $i = 1,2,3$ for 1D system of equations. The interface fluxes are given by

$$\left(G(U)_{j+\frac{1}{2}} = \frac{1}{2}\left[G(U)_{j+1} + G(U)_j \right] - \frac{1}{2}\left[\Delta G(U)^+_{j+\frac{1}{2}} - \Delta G(U)^-_{j+\frac{1}{2}} \right] \right)_i \tag{139}$$

$$\left(G(U)_{j-\frac{1}{2}} = \frac{1}{2}\left[G(U)_j + G(U)_{j-1} \right] - \frac{1}{2}\left[\Delta G(U)^+_{j-\frac{1}{2}} - \Delta G(U)^-_{j-\frac{1}{2}} \right] \right)_i \tag{140}$$

and

$$\left(\Delta G(U)^{\pm}_{j+\frac{1}{2}} = \frac{1}{2}\left[G(U)_{j+1} - G(U)_j \right] \pm \frac{1}{2}|\lambda|\left[U_{j+1} - U_j \right] \mp \frac{1}{2}\lambda_A \left[U_{j+1} - U_j \right] \right)_i \tag{141}$$

$$\left(\Delta G(U)^{\pm}_{j-\frac{1}{2}} = \frac{1}{2}\left[G(U)_j - G(U)_{j-1} \right] \pm \frac{1}{2}|\lambda|\left[U_j - U_{j-1} \right] \mp \frac{1}{2}\lambda_A \left[U_j - U_{j-1} \right] \right)_i \tag{142}$$

The viscous interface fluxes are computed as

$$\begin{aligned} \left(G_v(U)_{j+\frac{1}{2}} = \frac{1}{2}\left(G_v(U)_j + G_v(U)_{j+1} \right) \right)_i \\ \left(G_v(U)_{j-\frac{1}{2}} = \frac{1}{2}\left(G_v(U)_{j-1} + G_v(U)_j \right) \right)_i \end{aligned} \tag{143}$$

The values of λ and λ_A for each i in the above update formulation can be fixed using the methodologies described earlier for the inviscid case.

5. KFDS scheme in 2-D

The 2-D KFDS scheme is derived from a 2-D version of discrete velocity Boltzmann equation, which is based on the isotropic relaxation system introduced by Raghurama Rao and utilized in [24–26,49]. Consider a two dimensional hyperbolic system of conservation laws as given by

$$\frac{\partial U}{\partial t} + \frac{\partial G_1(U)}{\partial x} + \frac{\partial G_2(U)}{\partial y} = 0 \tag{144}$$

A five velocity DVBE for the above system can be derived as

$$\frac{\partial \mathbf{f}}{\partial t} + \frac{\partial \mathbf{h_1}}{\partial x} + \frac{\partial \mathbf{h_2}}{\partial y} = -\frac{1}{\epsilon}\left[\mathbf{f} - \mathbf{f}^{eq} \right] \tag{145}$$

where $\mathbf{h_1} = \mathbf{\Lambda_1 f}, \mathbf{h_2} = \mathbf{\Lambda_2 f}$ and the discrete velocity matrices are given by

$$\Lambda_1 = \begin{bmatrix} -\lambda & 0 & 0 & 0 & 0 \\ 0 & \lambda & 0 & 0 & 0 \\ 0 & 0 & \lambda_o & 0 & 0 \\ 0 & 0 & 0 & \lambda & 0 \\ 0 & 0 & 0 & 0 & -\lambda \end{bmatrix} \Lambda_2 = \begin{bmatrix} -\lambda & 0 & 0 & 0 & 0 \\ 0 & -\lambda & 0 & 0 & 0 \\ 0 & 0 & \lambda_o & 0 & 0 \\ 0 & 0 & 0 & \lambda & 0 \\ 0 & 0 & 0 & 0 & \lambda \end{bmatrix} \tag{146}$$

and the equilibria are given by

$$\mathbf{f}^{eq} = \begin{bmatrix} f_1^{eq} \\ f_2^{eq} \\ f_3^{eq} \\ f_4^{eq} \\ f_5^{eq} \end{bmatrix} = \begin{bmatrix} \frac{1}{4}U - \frac{1}{4\lambda}G_1 - \frac{1}{4\lambda}G_2 - \frac{\lambda-2\lambda_o}{4\lambda} f_o^{eq} \\ \frac{1}{4}U + \frac{1}{4\lambda}G_1 - \frac{1}{4\lambda}G_2 - \frac{1}{4} f_o^{eq} \\ f_o^{eq} \\ \frac{1}{4}U + \frac{1}{4\lambda}G_1 + \frac{1}{4\lambda}G_2 - \frac{\lambda+2\lambda_o}{4\lambda} f_o^{eq} \\ \frac{1}{4}U - \frac{1}{4\lambda}G_1 + \frac{1}{4}\lambda G_2 - \frac{1}{4} f_o^{eq} \end{bmatrix} \tag{147}$$

Let us denote $\mathbf{h}_x = \mathbf{\Lambda_1 f}, \mathbf{h}_y = \mathbf{\Lambda_2 f}$ as the fluxes along the $x-$ and $y-$ directions. For a quadrilateral finite volume, the cell-interface fluxes are constructed to be normal to the interfaces and can be obtained as

$$\mathbf{h}_L = \left(\mathbf{h}_x \cos\theta + \mathbf{h}_y \sin\theta \right)_L \tag{148}$$

$$\mathbf{h}_R = \left(\mathbf{h}_x \cos\theta + \mathbf{h}_y \sin\theta \right)_R \tag{149}$$

where the suffix L and R represent the left and the right states of the cell-interface. A finite volume update formula for the 2D Euler system, for a quadrilateral mesh, can be derived as

$$\mathbf{f}^{n+1}_{j,k} = \mathbf{f}^n_{j,k} - \frac{\Delta t}{A_{j,k}} \sum_{I_c=1}^{4} \mathbf{h}^n_{nI_c} \Delta s_{I_c} \tag{150}$$

where $A_{j,k}$ is the area of the cell centred at (j,k) and Δs_{I_c} is the length of the cell interface I_c. Applying the *Kinetic Flux Difference Splitting* across a finite volume cell-interface, we get

$$\mathbf{h}_{n,I_c} = \frac{1}{2}\left[h_R + h_L \right] - \frac{1}{2}\left[\Delta h^+_{I,n} - \Delta h^-_{I,n} \right] \tag{151}$$

where

$$\Delta h^{\pm}_{I,n} = \left(\Lambda_1 \cos\theta \right)^{\pm} \Delta \mathbf{f}^{eq}_{I,n} + \left(\Lambda_2 \sin\theta \right)^{\pm} \Delta \mathbf{f}^{eq}_{In} \tag{152}$$

Upon taking moments of the above equations and extending the same techniques that we had applied in one dimensional system in determining the diffusion coefficients and f_o^{eq}, we recover the macroscopic update formula as

$$\mathbf{U}_{j,k}^{n+1} = \mathbf{U}_{j,k}^{n} - \frac{\Delta t}{A_{j,k}} \sum_{I_c=1}^{4} \mathbf{G}_{n,I_c}^{n} \Delta s_{I_c} \tag{153}$$

where

$$\mathbf{G}_{n,I_c} = \frac{1}{2}\left[G(U)_{n,R} + G(U)_{n,L}\right] - \frac{1}{2}\left[\Delta G(U)_{I_c}^{+} - \Delta G(U)_{I_c}^{-}\right] \tag{154}$$

$$\Delta G(U)_{I_c}^{\pm} = \frac{1}{2}\left[G(U)_{n,R} - G(U)_{n,L}\right]\Big] \pm \frac{1}{2}|\lambda|\left[U_{n,R} - U_{n,L}\right] \mp \frac{1}{2}\lambda_A\left[U_{nR} - U_{nL}\right]. \tag{155}$$

Here again, the primary diffusion coefficient $|\lambda|$ is fixed based on the principle of flux equivalence across a steady discontinuity. It is interesting to note that unlike in the one dimensional cases, there is inherent multi-dimensional diffusion in the 2D version of the numerical scheme owing to the standard finite volume formulation which lacks truly multidimensional modelling. Therefore the strategy for adding additional definition does not require an entropy scale to allow for additional diffusion. However, the numerical errors in calculating the change in thermodynamic entropy still need to be considered. Therefore the strategy for fixing additional diffusion is defined as follows.

$$\lambda_A = \begin{cases} \lambda_{DD} \text{ for } \quad D^2 > 0 \quad \& \quad \Delta S \le \epsilon \\ 0 \quad \text{Otherwise} \end{cases} \tag{156}$$

where ϵ is a small number and is taken as 1.0×10^{-10}. As in 1-D, two variations are introduced for evaluating λ_{DD} in the 2-D case, for the finite volume method.

5.1. Maximum eigenvalue based λ_{DD}

The definition of λ_{DD} in 2D at the cell-interface is

$$\lambda_{DD} = \lambda_{max} - \lambda \tag{157}$$

where

$$\lambda_{\max} = \max\left\{\max(|u_n + a|, |u_n|, |u_n - a|)_L, \max(|u_n + a|, |u_n|, |u_n - a|)_R\right\} \tag{158}$$

with

$$\left(u_n = u_1 \cos\theta + u_2 \sin\theta\right)_{L,R} \tag{159}$$

This makes the coefficient of the total numerical diffusion as λ_{max} at the expansions.

5.2. Fluid velocity based λ_{DD}

$$\lambda_{DD} = \lambda_{RI} - \lambda \tag{160}$$

$$\text{where } \lambda_{RI} = \frac{|u_{n,L}| + |u_{n,R}|}{2} \tag{161}$$

6. Results and discussion

The Kinetic Flux Difference Splitting (KFDS) method is tested systematically on various 1-D and 2-D benchmark test cases to evaluate and establish its capabilities. Presented in this section are the one dimensional test cases [46] which are meant to evaluate the robustness of the scheme, involving nonlinear waves and their interactions. Following them, the numerical scheme is tested for typical benchmark cases in two dimensional Eulerian flow fields, which present various shock waves, contact discontinuities, expansion waves and their interactions. Finally, the scheme is tested for viscous fluid flows involving boundary layers and their interactions with nonlinear waves. Before presenting the results, the experimental order of convergence (EOC) studies are presented in the following subsection, demonstrating the order of accuracy of KFDS scheme.

Table 2
EOC using L1 and L2 error norms for KFDS-1O with a smooth periodic test case.

N	Grid spacing	L1 Error	EOC	L2 Error	EOC
10	0.10000	0.011613777927		0.009267137803	
20	0.05000	0.006088680399	0.93164	0.004795147573	0.95055
40	0.02500	0.003097630000	0.97496	0.002436038922	0.97704
80	0.01250	0.001566789511	0.98335	0.001234024541	0.98117
160	0.00625	0.000838144804	0.90254	0.000692443990	0.83360

Table 3
EOC using L1 and L2 error norms for KFDS-2O with a smooth periodic test case.

N	Grid spacing	L1 Error	EOC	L2 Error	EOC
10	0.10000	0.004957979218		0.004141851951	
20	0.05000	0.001549110966	1.67831	0.001495025225	1.47011
40	0.02500	0.000482411631	1.68310	0.000513302408	1.54229
80	0.01250	0.000136000286	1.82666	0.000166466154	1.62458
160	0.00625	0.000035931520	1.92029	0.000042902870	1.95608

6.1. Experimental Order of Convergence (EOC)

Here, a typical test is presented for evaluating the *Experimental Order of Convergence* (EOC) for the KFDS scheme. Consider a one dimensional computational domain of size [0,2]. The flow variables in the domain are initialized with a sinusoidal variation in the density while keeping the pressure and velocity constant. The exact solution for such initial conditions is given below.

$$\rho(x,t) = 1.0 + 0.2\sin(\pi(x - ut)) \tag{162}$$

$$u(x,t) = 0.1, \;\; p(x,t) = 0.5 \tag{163}$$

The numerical simulations are carried out by changing the number of computational cells methodically as 10, 20, 40, …, 160 cells. The solutions are computed with each grid size for the final time of 0.5 s. The exact solution is used for both initializing the computational domain and for enforcing the periodic boundary condition. The L1 and L2 errors represented by $\|\mathcal{E}_K\|_{L^1}$ *and* $\|\mathcal{E}_K\|_{L^2}$ are calculated at $t = 0.5$ s using Eqs. (164) and (165) as given below.

$$\|\mathcal{E}_K(t)\|_{L^1} = \Delta x \sum_{j=1}^{K} \left|\rho_j^{numerical} - \rho_j^{analytical}\right| \tag{164}$$

$$\|\mathcal{E}_K(t)\|_{L^2} = \sqrt{\Delta x \sum_{j=1}^{K} \left(\rho_j^{numerical} - \rho_j^{analytical}\right)^2} \tag{165}$$

where K is the number of cells. As we expect the error to behave as $\|\mathcal{E}\| = C\Delta x^p + \mathcal{O}\left(\Delta x^{p+1}\right)$ for p^{th} order accuracy, we can write

$$\frac{\rho_{\Delta x} - \rho_{exact}}{\rho_{\frac{\Delta x}{2}} - \rho_{exact}} = \frac{C\Delta x^p + \mathcal{O}\left(\Delta x^{p+1}\right)}{C\left(\frac{\Delta x}{2}\right)^p + \mathcal{O}\left(\frac{\Delta x}{2}\right)^{p+1}} = 2^p + \mathcal{O}(\Delta x) \tag{166}$$

Thus,

$$\log_2\left(\frac{\rho_{\Delta x} - \rho_{exact}}{\rho_{\frac{\Delta x}{2}} - \rho_{exact}}\right) = p + \mathcal{O}(\Delta x) \tag{167}$$

As $K \propto \frac{1}{\Delta x}$, the *Experimental Order of Convergence* (EOC) is computed using (168) as follows.

$$\text{EOC} = \log_2\left(\frac{\|\mathcal{E}_{K/2}\|}{\|\mathcal{E}_K\|}\right) \tag{168}$$

The L1 and L2 norms obtained for each of the test cases of both first order accurate and second order accurate versions of KFDS scheme and are tabulated in Tables 2 and 3 respectively. The log–log plots comparing the L1 and L2 norm errors with slope 2 are shown in Figs. 3 and 4 respectively.

6.2. Test cases for 1-D Euler equations

The initial conditions for the one dimensional Euler test cases are summarized in Table 4.

Table 4
1D Euler equations: Test case matrix.

S No.	Test case	ρ_L	u_L	p_L	ρ_R	u_R	p_R
1	Steady shock	1.0	1.0	0.17875	2.6665	0.375	0.80357
2	Steady contact discontinuity	1.4	0	0.4	1.0	0	0.4
3	Shock tube	1.0	0.75	1.0	0.125	0	0.1
4	Overheating	1.0	−2.0	0.4	1.0	2.0	0.4
5	Toro Case 3	1.0	0	1000	1.0	0	0.01
6	Shock collision	5.99924	19.5975	460.894	5.99242	−6.19633	46.0950
7	Toro Case 5	1.0	−19.5975	1000	1.0	−19.59745	0.01

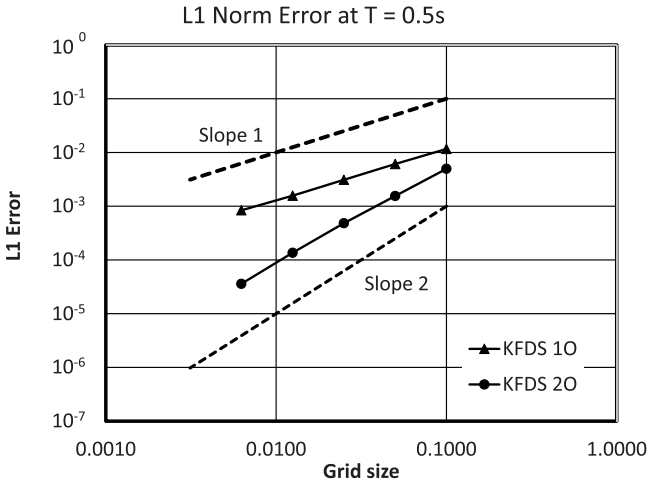

Fig. 3. L1 norm errors for KFDS-1O and KFDS-2O schemes.

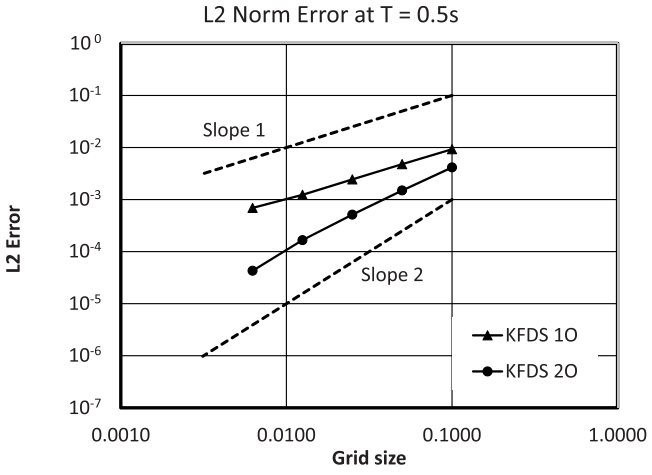

Fig. 4. L2 norm errors for KFDS-1O and KFDS-2O schemes.

The first two test cases involve the steady shock and steady contact discontinuity respectively. The test results for the two test cases are shown in Figs. 5 and 6. The KFDS scheme has the ability to capture steady shock waves and steady contact discontinuities exactly, without numerical diffusion. The introduction of additional numerical diffusion for smooth regions is regulated by the D^2-distance and entropy difference. Thus, the exact discontinuity capturing ability of the scheme is undisturbed, as observed in the results.

The third test case is a classic shock tube test case wherein the gases are maintained at different thermodynamic states on either side of the diaphragm. Upon rupturing the diaphragm, the abrupt difference in the pressure, density and temperature results in the formation and evolution of an expansion wave, a contact discontinuity and a shock wave in the flow field. While the flow is unsteady in nature, this test case assesses the accuracy of the scheme as well as its ability to capture each of the nonlinear and linear waves in the flow field. Ideally, most low diffusive schemes would require an entropy fix to overcome the problems of unphysical expansion shocks that arise in the expansion regions. The use of D^2-distance based additional diffusion has ensured an alternate way of overcoming this problem as is evident in the results [Fig. 7].

The fourth test case is the well-known overheating problem where the initial strong velocity gradient results in the evolution of two expansion waves moving in the opposite directions, separated by a contact discontinuity. This case challenges the ability of the numerical scheme to provide physically relevant results. Both the versions of KFDS scheme could generate stable results [Fig. 8] without blowing up, but with a significant deviation in the internal energy near the contact discontinuity. Test cases 5, 6 and 7 (Figs. 9, 10 and 11) are meant to test the robustness of the scheme and its ability to handle very strong gradients in pressure. Again it can be seen that both versions of KFDS scheme are able to provide solutions without any unphysical behaviour. However, mild oscillations are observed in KFDS-B solutions in certain test cases and appear to die out with lower CFL numbers.

6.3. Test case for 2-D Euler equations

6.3.1. Slip flow

The first of the test cases is a check on the ability to capture grid aligned contact discontinuities [50]. The test case involves a uniform flow of two identical fluids with different speeds of u = 2 and 3, which slip over each other. The computational domain [0,1][0,1] is discretized into a 20 × 20 grid and the domain is initialized with uniform pressure and density. The upper half of the left boundary is maintained at u = 3 while the lower half of the left boundary is maintained at u = 2. The computational results for each version of the KFDS scheme are shown in Fig. 12. The basic ability of the KFDS scheme to capture grid aligned discontinuities exactly is retained.

6.3.2. Oblique shock reflection

The computational domain for this test case [51] is rectangular: $[0,3]\times[0,1]$. An oblique shock wave is introduced from the top-left corner by means of inflow boundary conditions at the left boundary and the post-shock boundary conditions on the top side of the domain – $Inflow$: $(\rho,u,v,p)_{0,y,t} = (1.0, 2.9, 0, 1/1.4)$ and $Post-shock\ conditions$: $(\rho,u,v,p)_{x,1,t} = (1.69997, 2.61934, -0.50633, 1.52819)$ [19]. The bottom side is maintained as a solid wall where flow tangency boundary condition is applicable and at the right boundary the supersonic outflow boundary condition is prescribed. The computational grid used for the numerical simulation is 240 × 80. The test results are shown for both versions of KFDS scheme with first order and second order accuracy in Fig. 13. The shock positions and the point of reflection match well with the results presented in [40]. There is a marked improvement in the shock crispness between the first order and second order accurate results.

6.3.3. Supersonic flow over a compression ramp

The computational domain for this test case [52] involves a 15° ramp at the bottom wall of the computational domain of size $[0,3]\times[0,1]$. A steady inflow of Mach 2 over the ramp results in the generation of an oblique shock wave at the compression corner. This oblique shock hits the top wall and gets further reflected down the flow. This reflected shock wave interacts

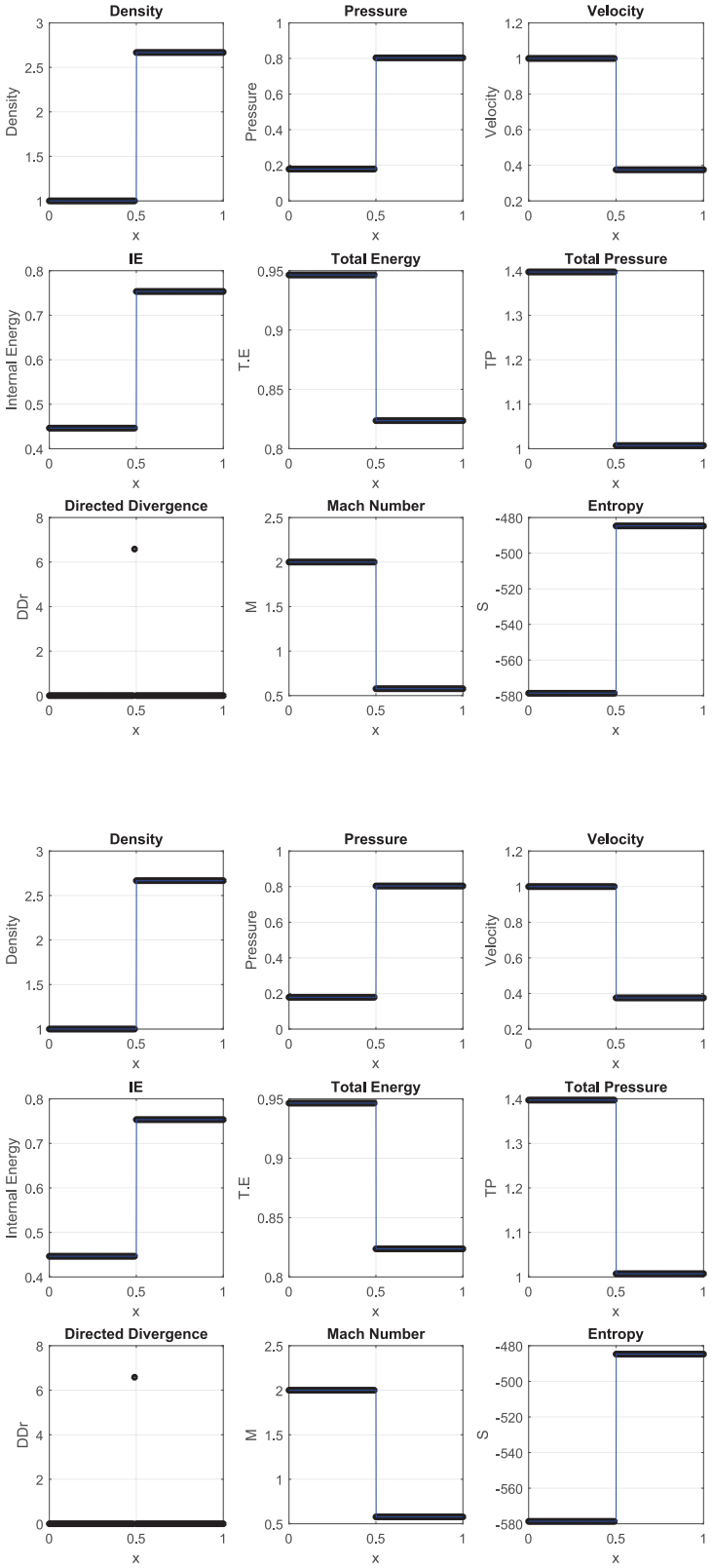

Fig. 5. Test case 1: (a) KFDS-A and (b) KFDS-B.

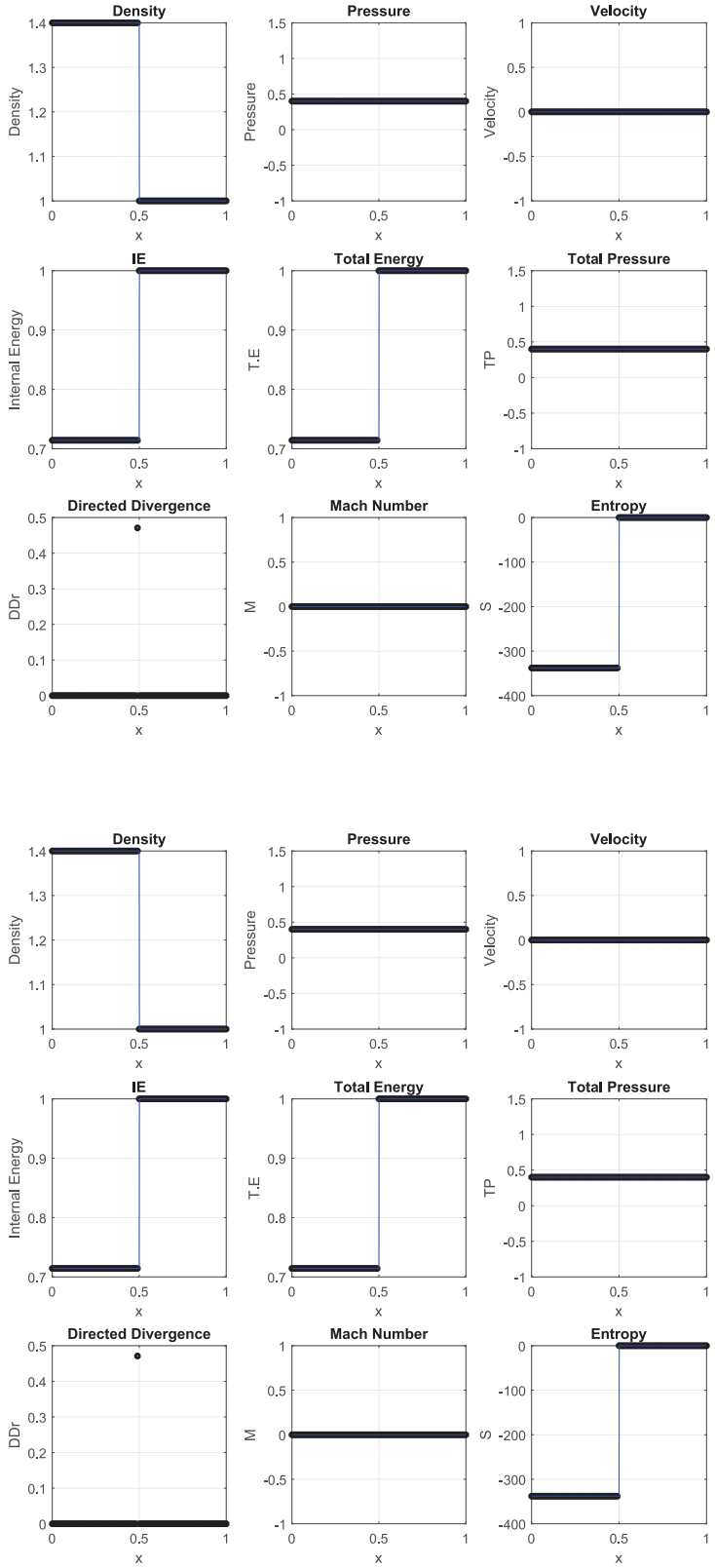

Fig. 6. Test case 2: (a) KFDS-A and (b) KFDS-B.

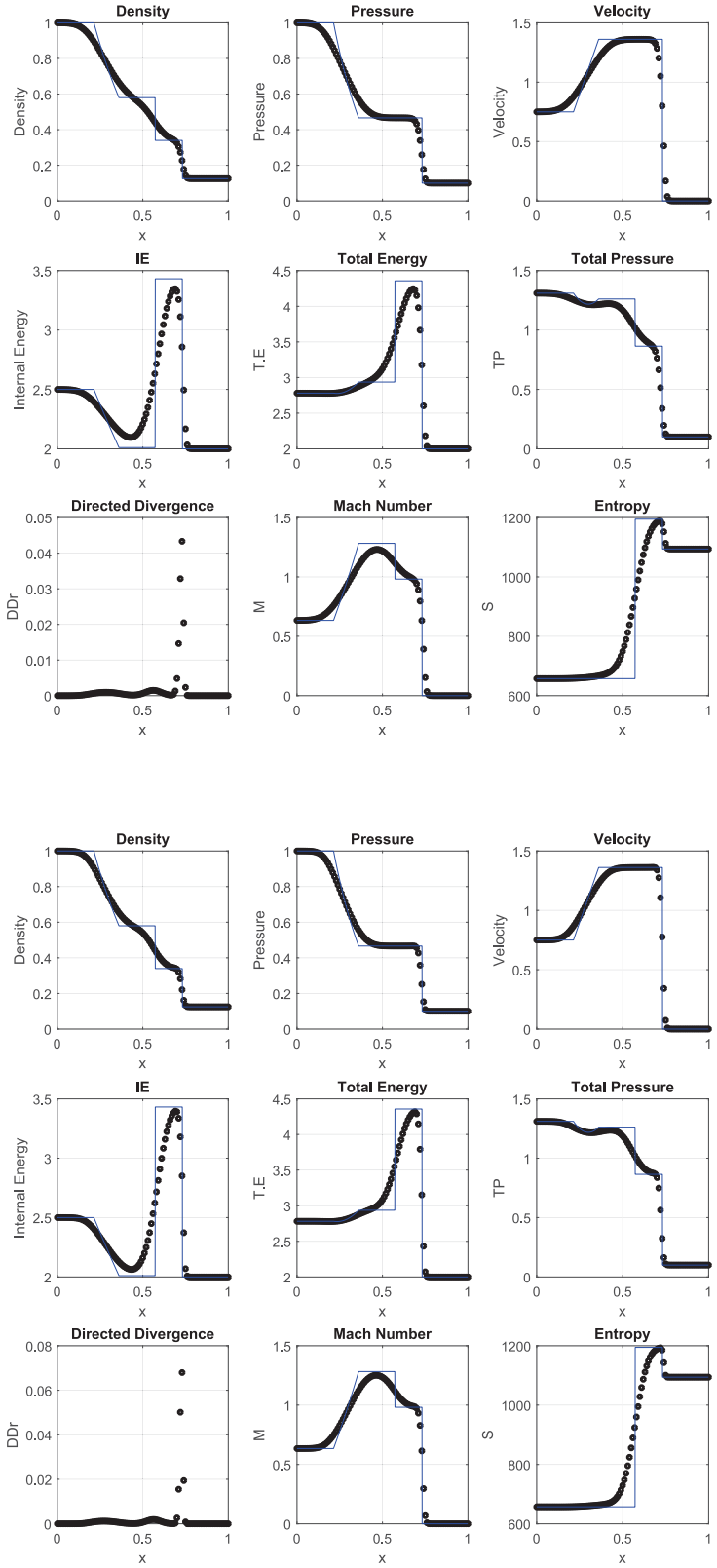

Fig. 7. Test case 3: (a) KFDS-A and (b) KFDS-B.

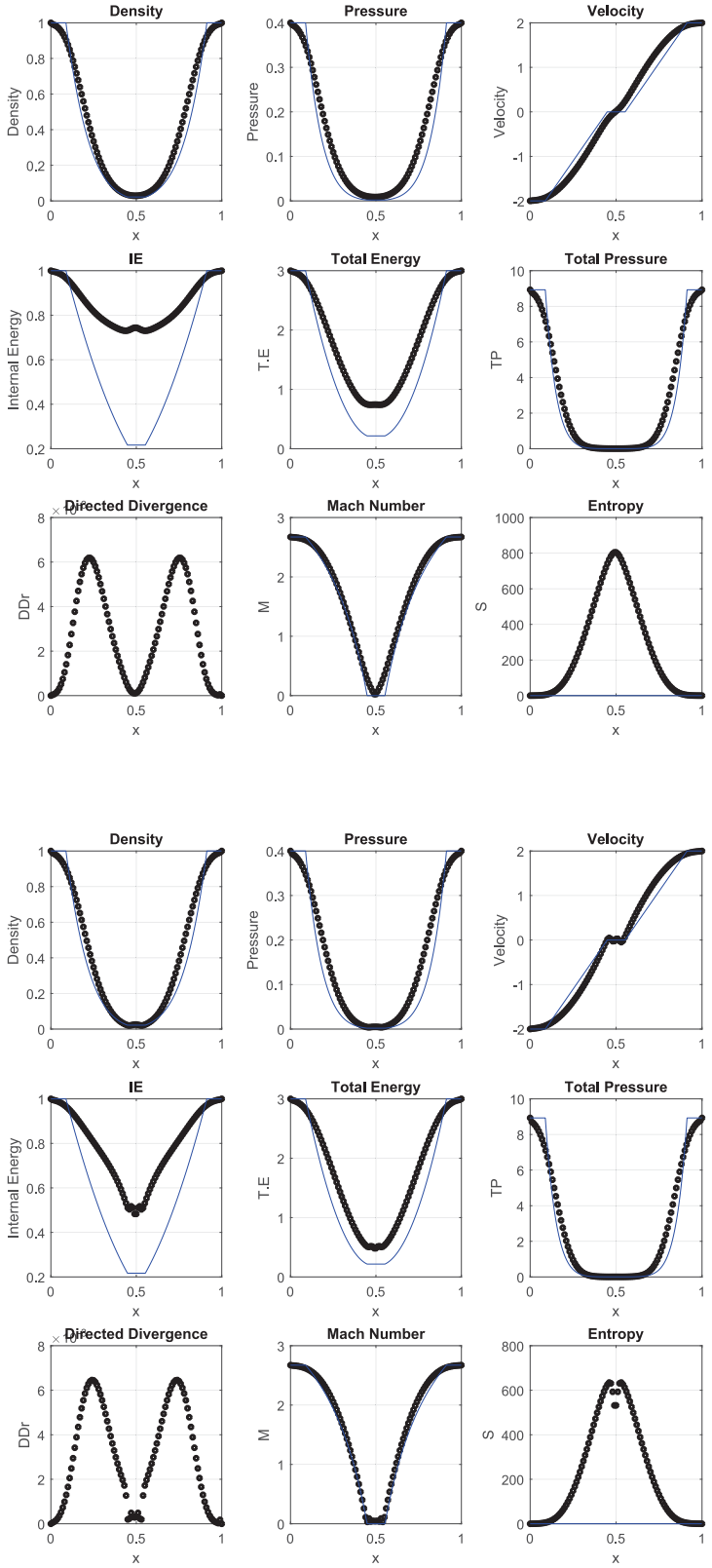

Fig. 8. Test case 4: (a) KFDS-A and (b) KFDS-B.

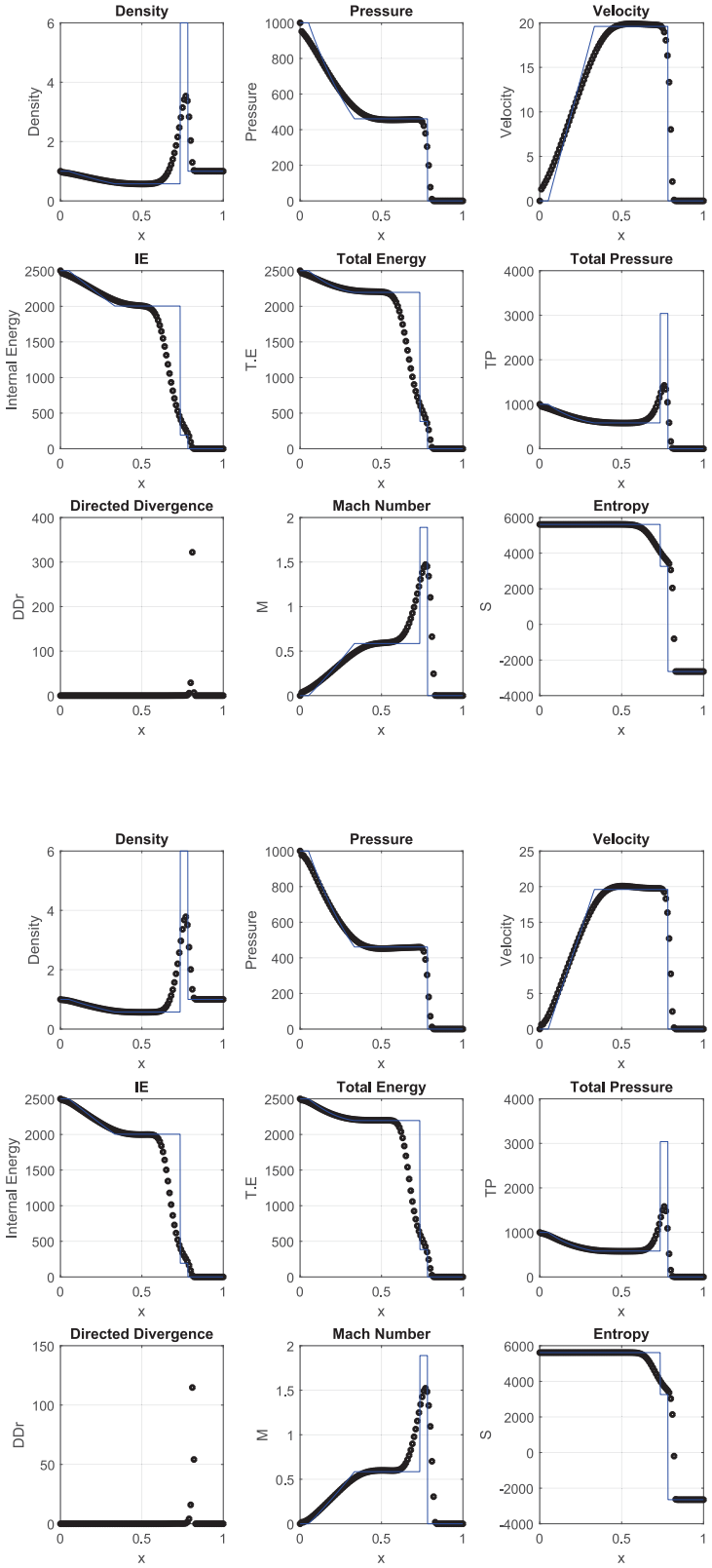

Fig. 9. Test case 5: (a) KFDS-A and (b) KFDS-B.

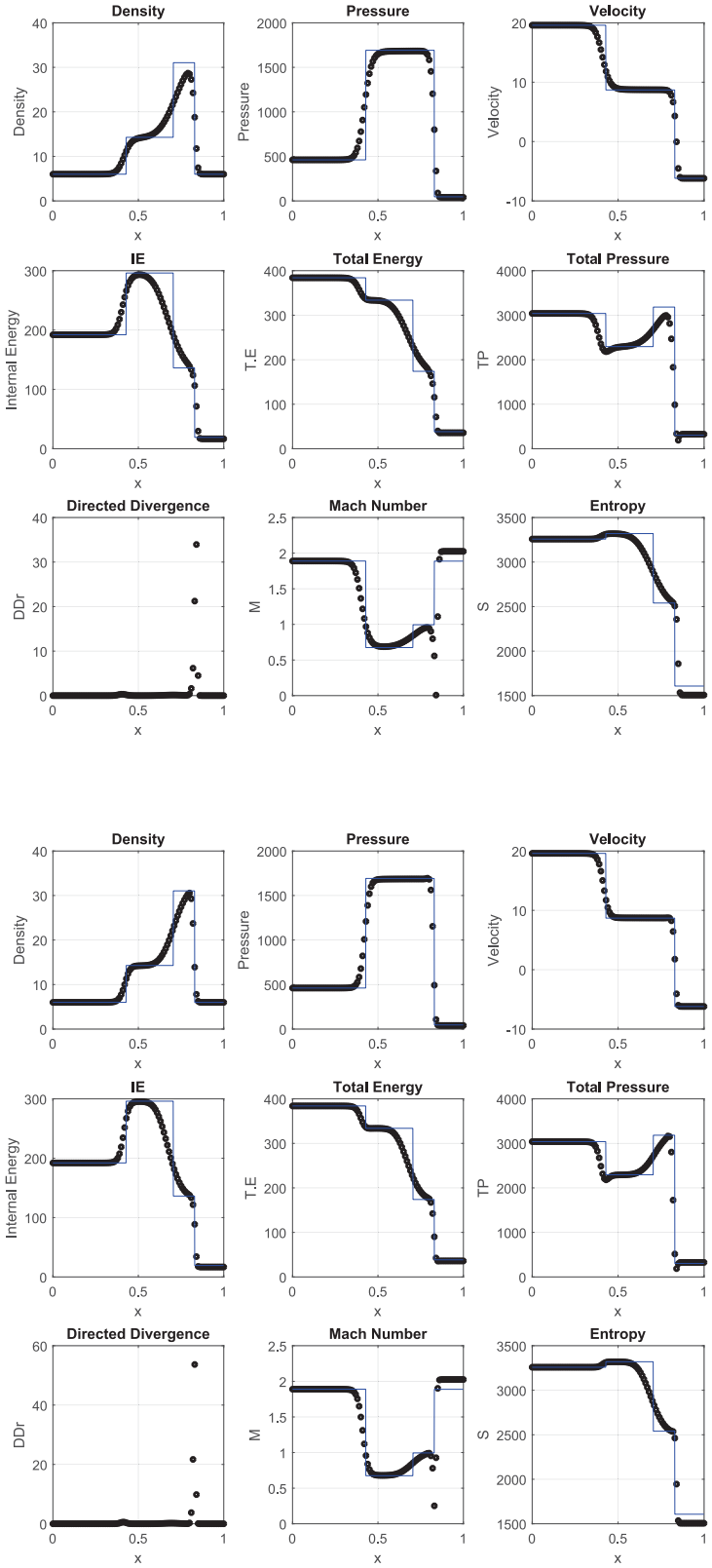

Fig. 10. Test case 6:(a) KFDS-A and (b) KFDS-B.

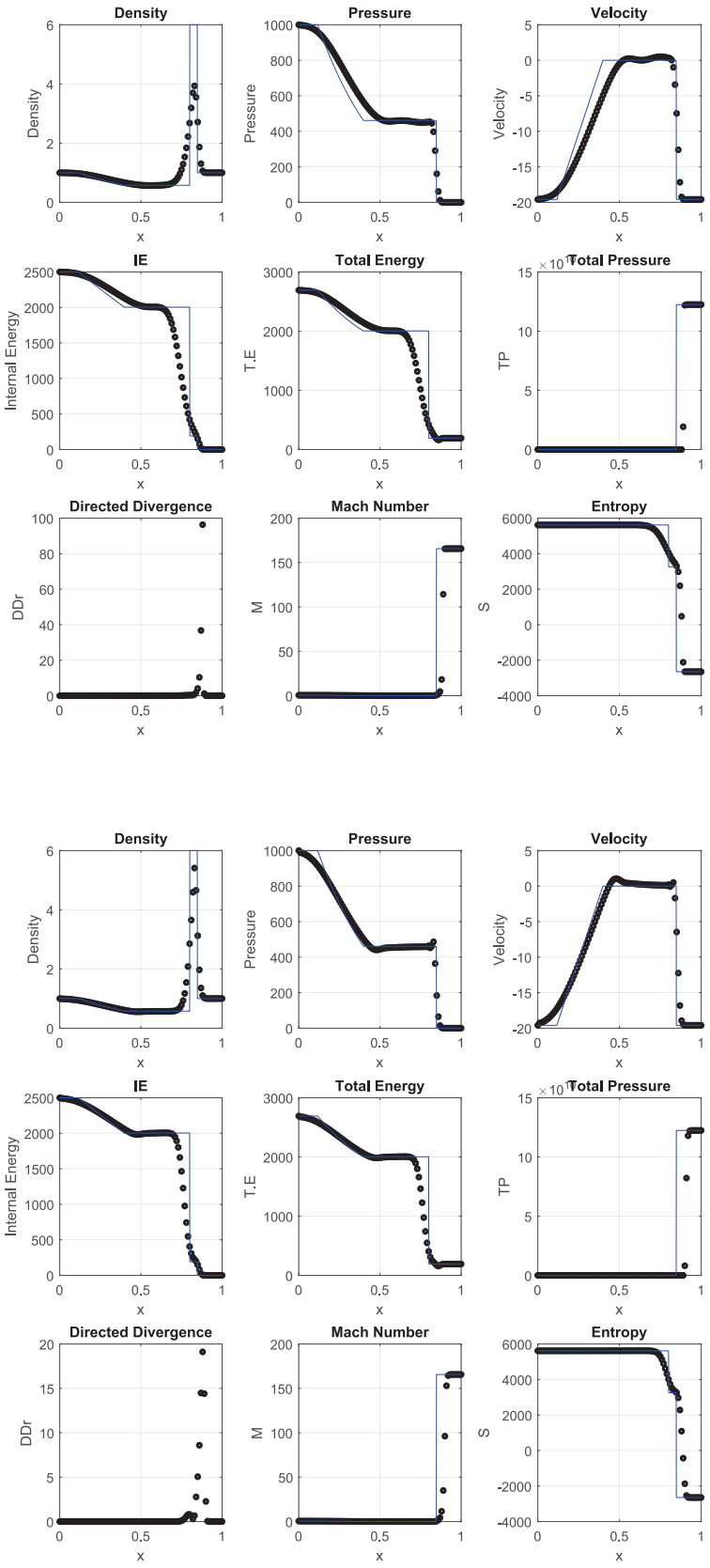

Fig. 11. Test case 7: (a) KFDS-A and (b) KFDS-B.

with the expansion wave generated from the upper corner of the ramp. This test case involves evolution of both shock and expansion waves and their interactions. The test results are shown for both the versions of KFDS with first order and second order accuracy in Fig. 14. No expansion shocks are seen, thus confirming the efficacy of the strategy of adding additional numerical diffusion based on D^2-distance.

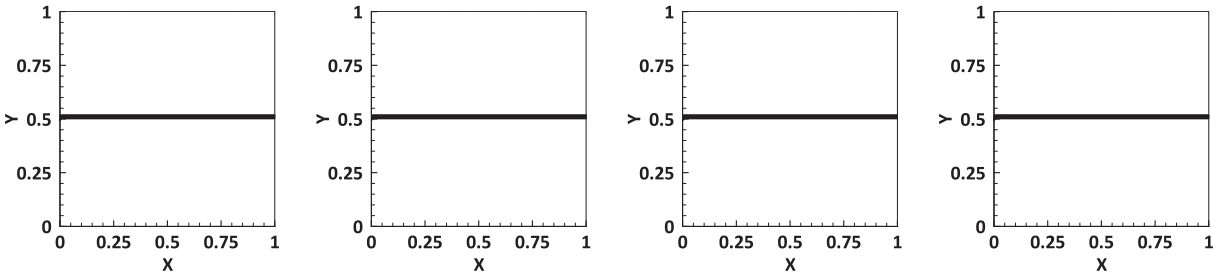

Fig. 12. Test case: Slip Flow (20 × 20) – Mach contours (2.0: 0.025: 3.0) – (a) 1O-KFDS-A (b) 1O-KFDS-B (c) 2O-KFDS-A (d) 2O-KFDS-B.

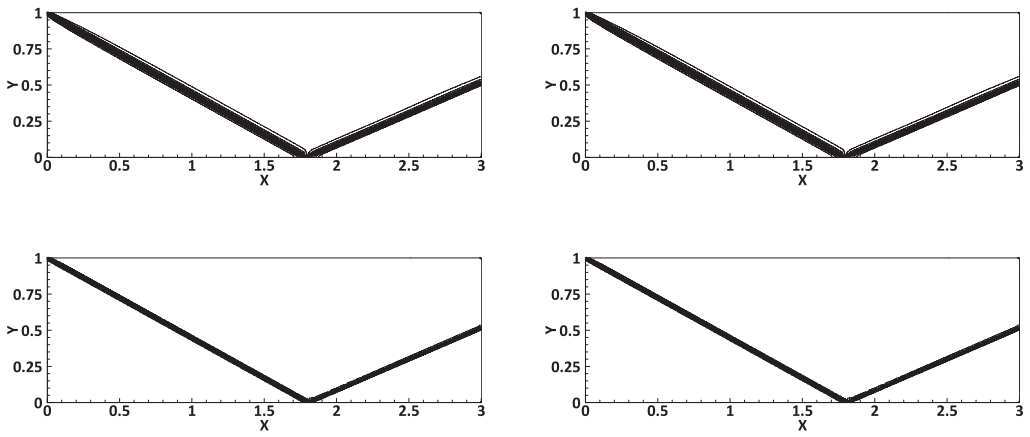

Fig. 13. Test case: Oblique shock reflection (240 × 80) – pressure contours (0.7: 0.1: 2.9) – (a) 1O-KFDS-A (b) 1O-KFDS-B (c) 2O-KFDS-A (d) 2O-KFDS-B.

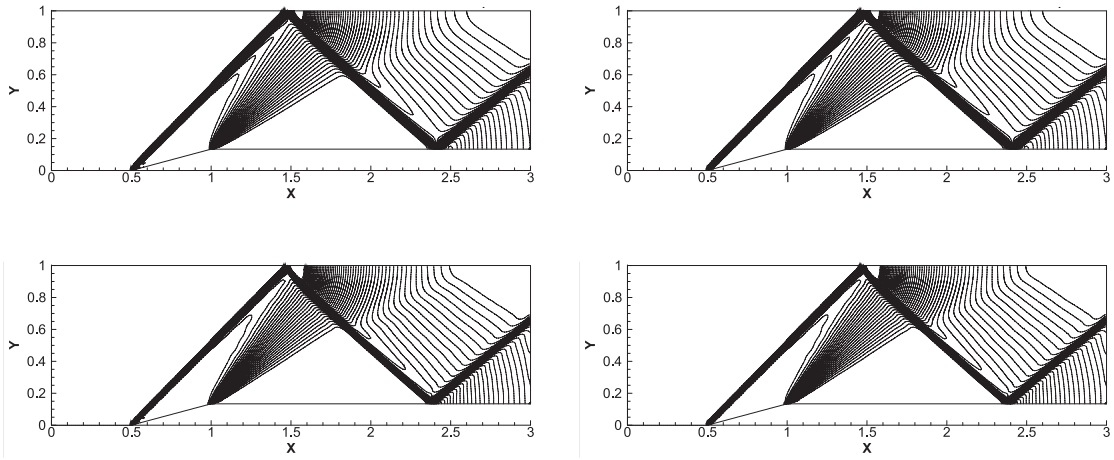

Fig. 14. Test case: Supersonic flow over a ramp (240 × 80) – pressure contours 1.1 : 0.05 :3.8 – (a) 1O-KFDS-A (b) 1O-KFDS-B (c) 2O-KFDS-A (d) 2O-KFDS-B.

6.3.4. *Supersonic flow over a bump*

This test case [53] involves a channel having a 4% thick circular arc bump on the bottom side of the test domain of $[-1,2] \times [0,1]$. The bump has a chord length of 1 unit and is located at $x = 0.5$. The left side is marked as a supersonic inlet with free stream Mach number 1.4. The bottom side is defined as free slip wall. The top side of the domain is defined as an inviscid wall, while the right side of the domain is marked as supersonic outlet. The results for each version of the scheme is shown in Fig. 15. Both versions of KFDS scheme capture the shock initiation, reflections and interactions, and the positions of reflections and interactions are captured with reasonable accuracy .

6.3.5. *Hypersonic flow over a half cylinder*

This test case involves hypersonic inflows at Mach 6 and Mach 20 interacting with the leading half side of the cylinder. Ideally the flow would result in the evolution of a bow shock wave which is located upstream of the cylinder and is detached from it. The profile of the shock is dependent on the upstream Mach number. Incidentally some numerical schemes resolve these detached bow shock waves with anomalies known as carbuncle shock waves, upstream of the bow shock on the stagnation line, and are reported as shock instabilities [54]. The density contours for both the versions of KFDS scheme for Mach 6 are shown in Fig. 16. Similarly, The density contours for both the versions of KFDS scheme for Mach 20 are shown in Fig. 17. As can be seen, both the schemes do not exhibit any form of shock instabilities and resolve the bow shocks with reasonable accuracy.

6.3.6. *Planar shock reflection over a wedge*

This test case [54] comprises of a computational domain $[0,2] \times [0,1.5]$ with a 30° wedge positioned from $x = 0.5$. The test case involves the interaction of a normal shock wave moving at Mach 5.5 with the wedge.

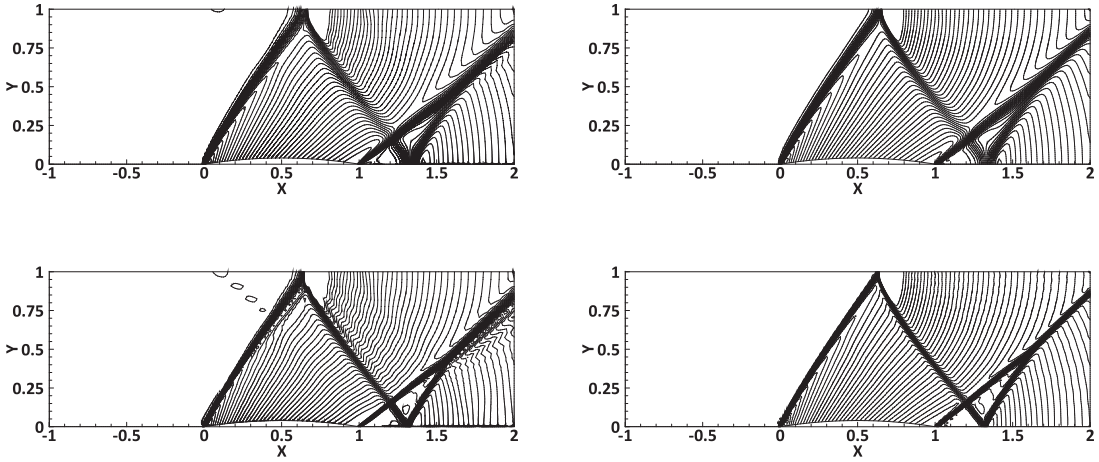

Fig. 15. Test case: Supersonic flow over a bump (240 × 80) – pressure contours (0.7: 0.025: 1.5) – (a) 1O-KFDS-A (b) 1O-KFDS-B (c) 2O-KFDS-A (d) 2O-KFDS-B.

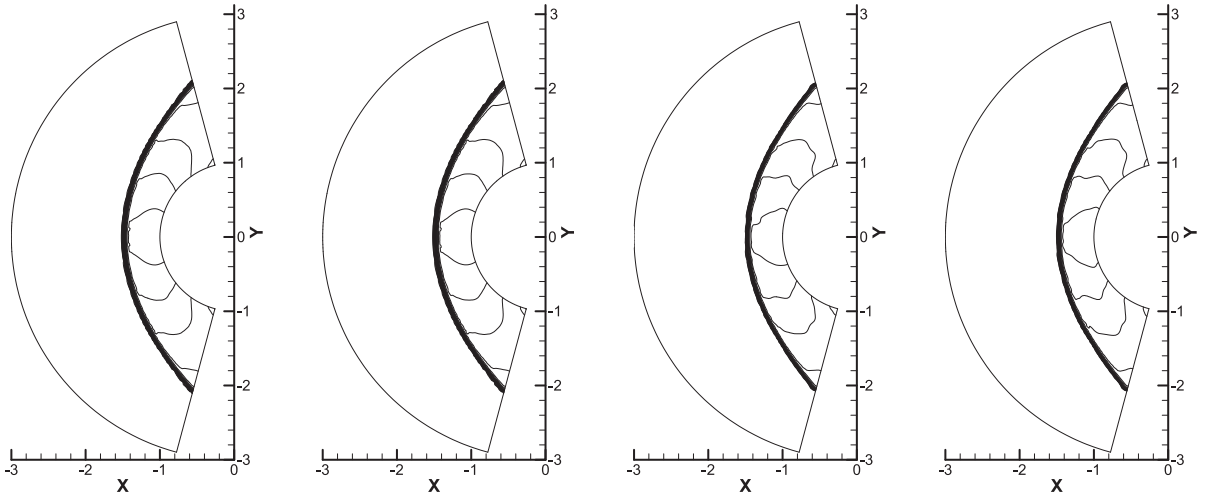

Fig. 16. Testcase: Hypersonic flow over a half cylinder at Mach 6 – Mach contours (0.0:0.4:7.6) – (160 × 80) – (a) 1O-KFDS-A (b) 1O-KFDS-B (c) 2O-KFDS-A (d) 2O-KFDS-B.

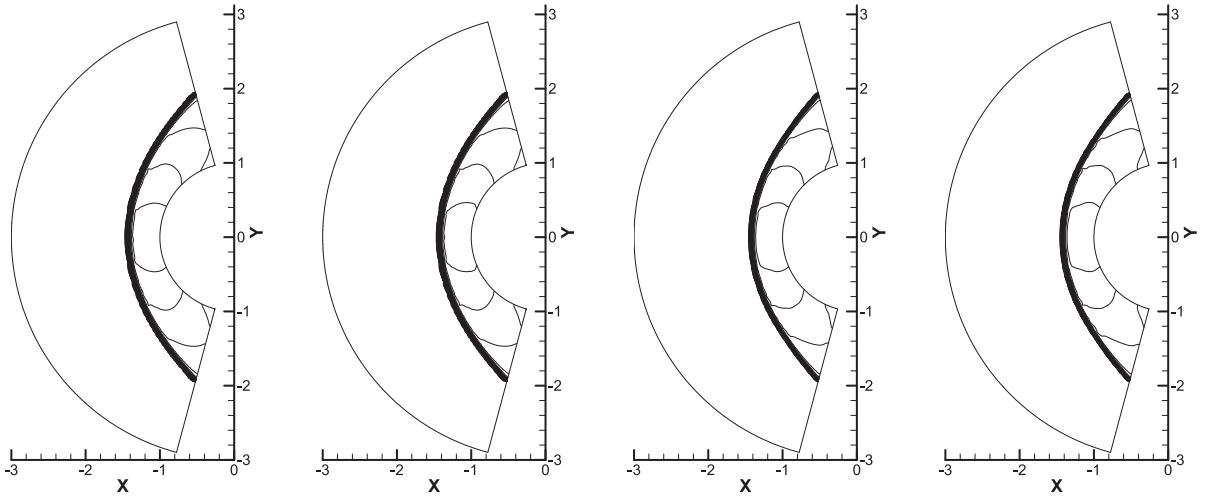

Fig. 17. Testcase: Hypersonic flow over a half cylinder at Mach 20 – Mach contours (0.0:0.4:20.0) – (160 × 80) – (a) 1O-KFDS-A (b) 1O-KFDS-B (c) 2O-KFDS-A (d) 2O-KFDS-B.

The initial shock is located at $x = 0.25$ and the computational domain to the right of x = 0.25 is initialized with stationary fluid with density 1.4 and pressure 1. The domain to the left of the shock is initialized with primitive variable values corresponding to the shock as obtained from the moving shock relations. The supersonic nature of the inflow results in an evolution of an oblique shock at the root of the wedge which interacts with the moving normal shock and gets reflected, thereby forming a triple point of shocks. Essentially this test case evaluates the ability of the numerical

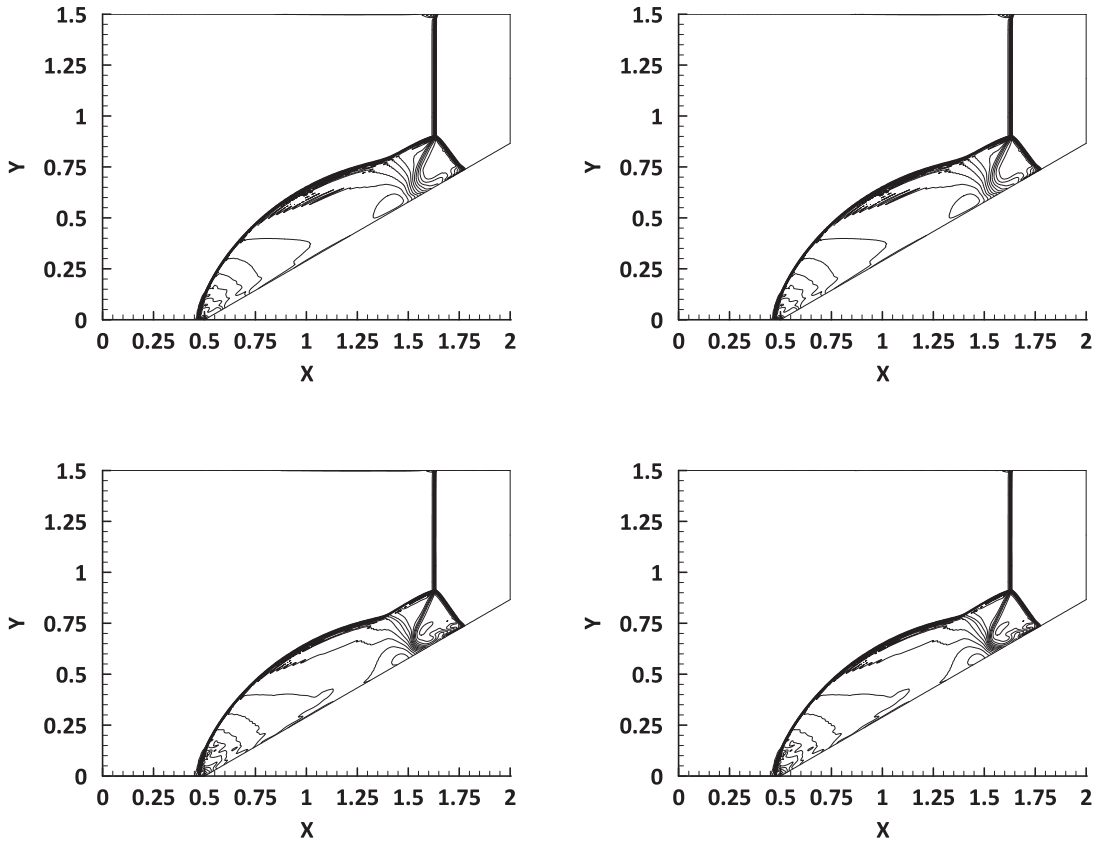

Fig. 18. Test case: Double Mach reflection over a wedge – density contours (5.0:0.5:20.0) (400 × 400) – (a) 1O-KFDS-A (b) 1O-KFDS-B (c) 2O-KFDS-A (d) 2O-KFDS-B.

scheme to handle shock instabilities and to provide a physically realistic solution. Many popular numerical schemes produce kinked Mach stems near the triple point which are unphysical. The results obtained from both the versions of KFDS scheme at t = 0.25 s are shown in Fig. 18. Both versions of KFDS scheme capture the triple point well and do not produce unphysical kinked Mach stems.

6.3.7. Shock diffraction

This test case [54] is important to assess the presence of shock instabilities and unphysical features in the strong expansion fan appearing with some numerical schemes. The test case involves the forward travel of a strong shock with incident Mach Number 5.09 wherein the domain comprises of a sudden expansion around a 90° corner. The shock diffracts around the corner [X = 0.05, Y = 0.625]. At the corner, the sudden area expansion leads to a strong expansion fan. Both the above nonlinear waves further interact. For this test case we discretize the computational domain [0,1][0,1] into a 400 × 400 grid. The initial shock position is specified at x = 0.05. The domain is initialized with the initial conditions $(\rho, u, v, p) = [1.4, 0, 0, 1.0]$ to the left of the shock and with post shock conditions to the right of the shock (computed from moving shock relations). The results obtained from the KFDS schemes at $t = 0.1561$ s are shown in Fig. 19. As can be observed, the both versions of KFDS scheme resolve the flow features arising due to the strong initial gradients well and do not produce any anomalies.

6.4. 2D viscous flow test cases

6.4.1. Blasius flow

This test problem is essentially a validation for the viscous part in a numerical scheme. The test involves viscous laminar flow over a flat plate at zero angle of incidence. The inlet has a freestream Mach number of 0.15 and Reynolds number 10 000. The computational domain $[-0.2, 1.8] \times [0, 1]$ is rectangular with the flat plate on the bottom side from $x = 0$ to $x = 1.8$ defined as a viscous wall. The inlet is defined with uniform total pressure and total temperature with zero vertical velocity component. The top and the outlet are defined with free stream boundary conditions. The remaining portion upstream of the plate is defined with symmetry boundary condition. The boundary layer evolves over the viscous wall and the grows in thickness in proportion to the length of the flat plate. A stretched grid of 105 × 65, with a geometrically increasing ratio of 1.025 is used. The boundary layer profile for each version of KFDS scheme is shown in Fig. 20. The skin friction distribution and the velocity profiles are shown in Figs. 21 and 22 respectively. Comparison with analytical solution (Blasius profile) is provided. Second order versions of the KFDS schemes agree well with the analytical results, while the effect of numerical diffusion can be seen in the first order results.

6.4.2. Supersonic viscous flow over a bump

This test case [55] involves a channel having a 4% thick circular arc bump on the bottom side of the test domain [−1,2] × [0,1]. The bump has a chord length of 1 unit and is located at $x = 0.5$. The left side is marked as a supersonic inlet with free stream Mach number 1.4 and Reynolds number 8000. Symmetry boundary condition is imposed on the bottom wall from $x = -1$ to $x = 0$. The remaining portion of the bottom side is marked as a viscous wall. The top side of the domain is defined as an inviscid wall, while the right side of the domain is marked as supersonic outlet. A geometrically stretched grid of size 240 × 80, with a 4.5% increase is used. The results for each version of the KFDS scheme are shown in Fig. 23. The interaction of the reflected shock wave with separated flow region to the right of the bump results in a weak reflection as can be seen in the results. The skin friction coefficients for the test case are shown in Fig. 24, are compared with those from the Ref. [55] and show reasonable agreement.

6.4.3. Shock wave–boundary layer interaction

This test case [56] constitutes an interaction of an oblique shock wave with a laminar boundary layer that evolves on the bottom side of the

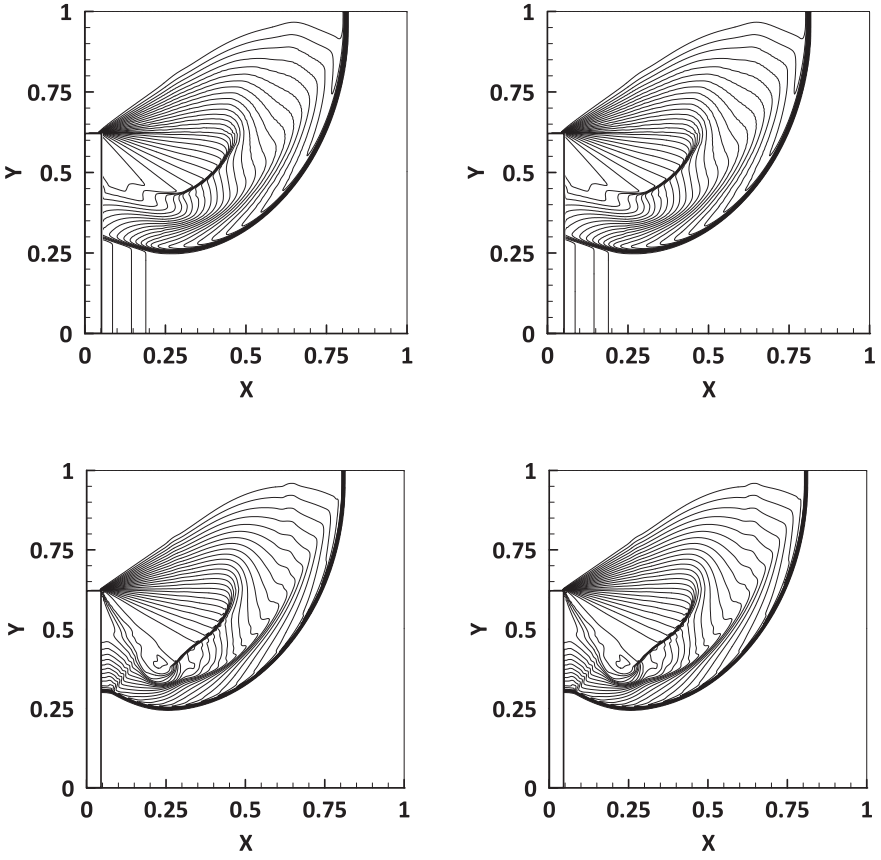

Fig. 19. Test case: Shock Diffraction in a sudden expansion region – density contours (0.5:0.25:6.75) (400 × 400) – (a) 1O-KFDS-A (b) 1O-KFDS-B (c) 2O-KFDS-A (d) 2O-KFDS-B.

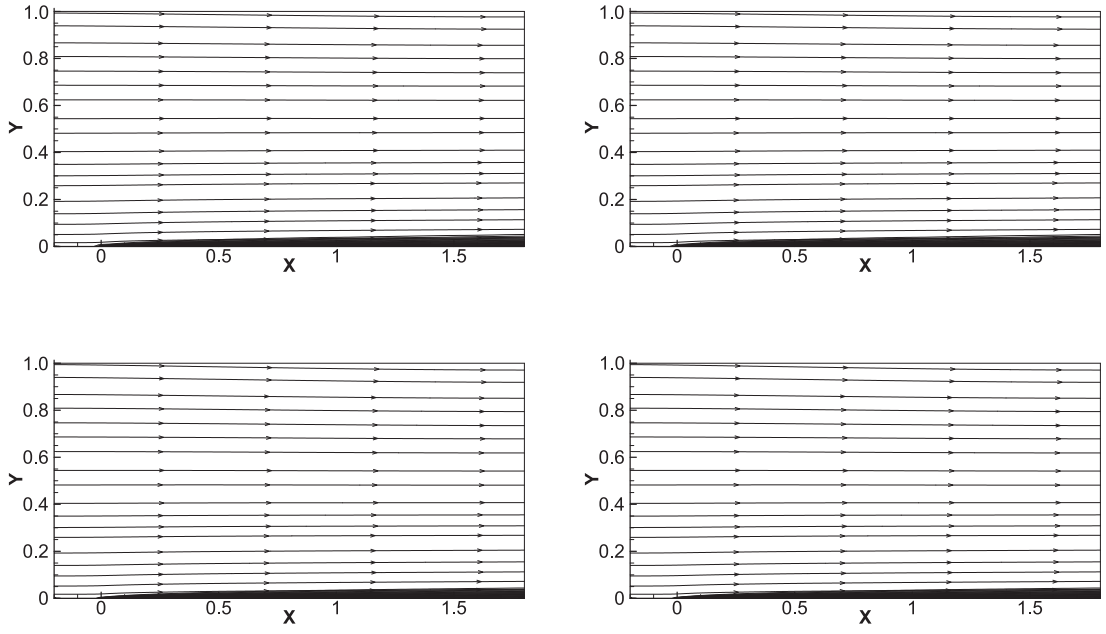

Fig. 20. Test case: Blasius flow over a flat plate (105 × 65) — (a) 1O-KFDS-A (b) 1O-KFDS-B (c) 2O-KFDS-A (d) 2O-KFDS-B.

computational domain [−0.2,1.8] × [0,1]. The interaction results in the formation of a streak of compression waves reflecting from the boundary layer accompanied by an adjacent expansion fan. Further, a recirculation zone in the form of a bubble develops on the surface around which the flow separates from the surface and gets reattached. It is evident that a wide combination of flow features evolve in the solution and thus tests the capability of the numerical scheme to capture each phenomenon listed. The inlet boundary comprises of a supersonic inflow with Mach number 2.0 and Reynolds number 100 000 till y = 0.765. The inlet region above y = 0.765 and the topside of the domain is initialized with post-shock boundary

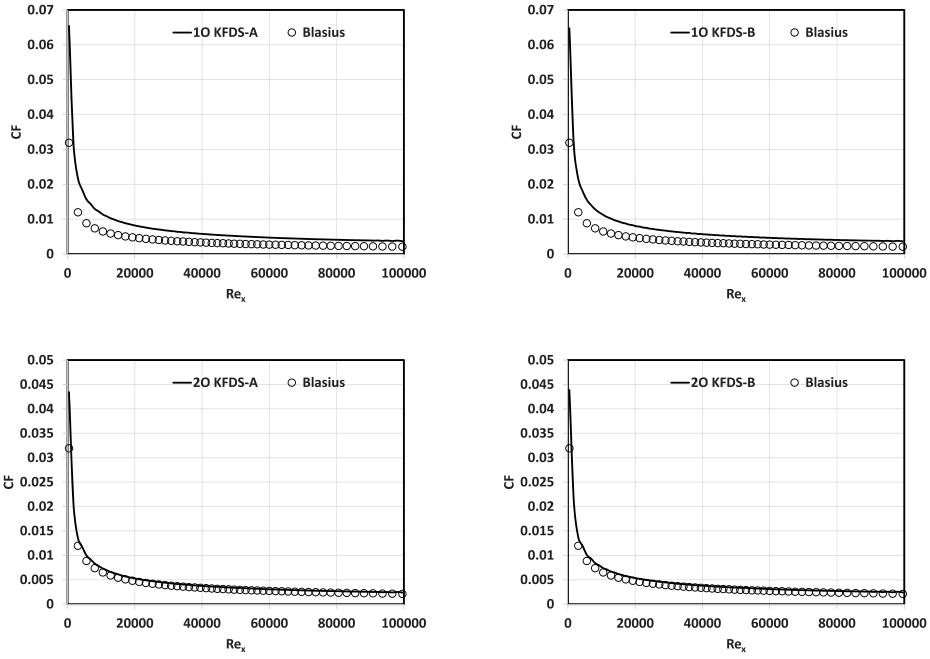

Fig. 21. Test case: Blasius flow over a flat plate (105 × 65) – C_f plots – (a) 1O-KFDS-A (b) 1O-KFDS-B (c) 2O-KFDS-A (d) 2O-KFDS-B.

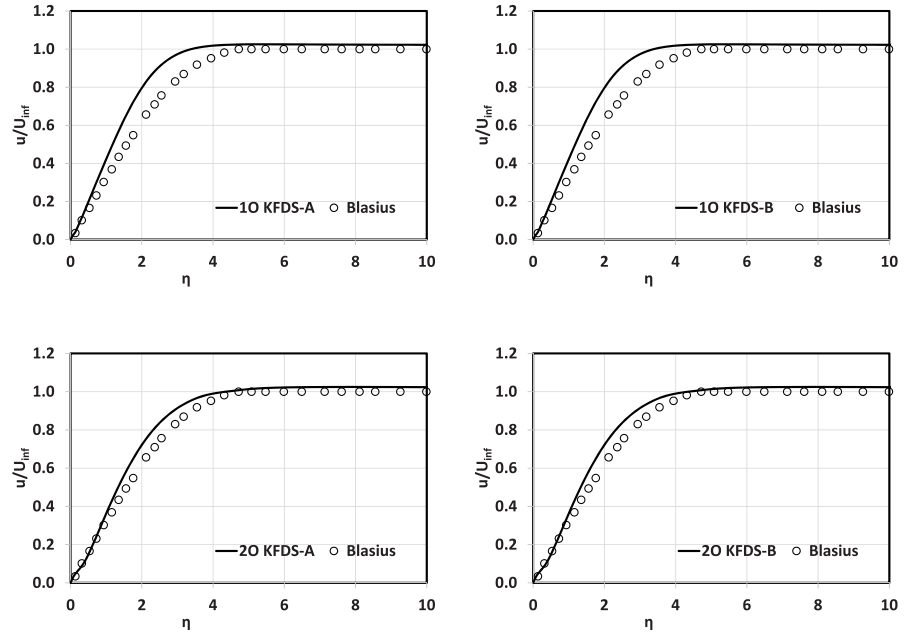

Fig. 22. Test case: Blasius flow over a flat plate (105 × 65) – Velocity Profiles – (a) 1O-KFDS-A (b) 1O-KFDS-B (c) 2O-KFDS-A (d) 2O-KFDS-B.

conditions. The right side of the domain is treated as a supersonic outlet. A geometrically stretched grid of size 141 × 121, with a 4.5% increase is used. Fig. 25 presents the pressure contours for the numerical scheme. The scheme is able to resolve the incident and reflected shock wave along with the expansion regions. Fig. 26 portrays the flow vectors and the streamlines closer to the bottom wall. The bubble resulting from the flow separation and reattachment can be observed. The computed skin friction coefficients and pressure coefficients are shown in Figs. 27 and 28 respectively. Both versions of KFDS schemes are able to capture the relevant flow features with reasonable accuracy.

7. Conclusions

A new discrete velocity model based kinetic flux difference splitting scheme is introduced and its capability to handle the complex flow features of compressible flows without the need of an entropy fix, yet retaining its basic ability to exactly capture grid aligned steady shocks and contact discontinuities, is demonstrated. Enforcing the principle of flux equivalence across a steady discontinuity results in satisfying R–H conditions in the discretization, which leads to exact capturing of steady discontinuities. The use of D^2-distance as a gradient sensor enables in precise addition of

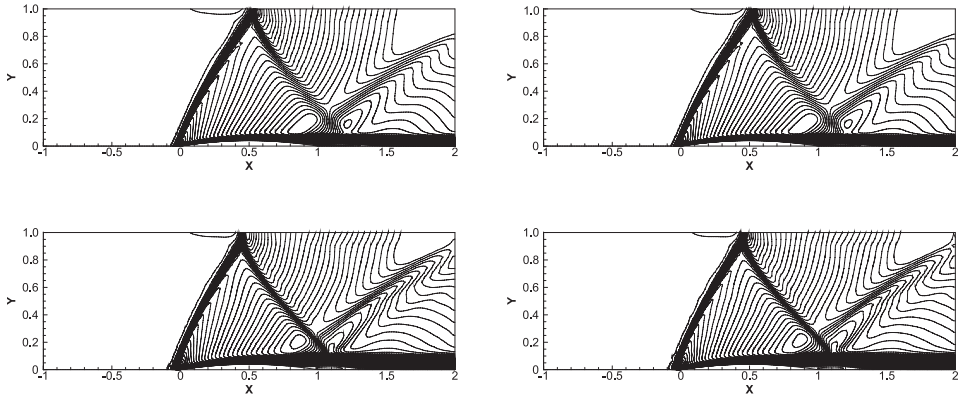

Fig. 23. Test case: Supersonic viscous flow over a bump (240 × 80) – Mach contours (0: 0.0215: 1.5) – (a) 1O-KFDS-A (b) 1O-KFDS-B (c) 2O-KFDS-A (d) 2O-KFDS-B.

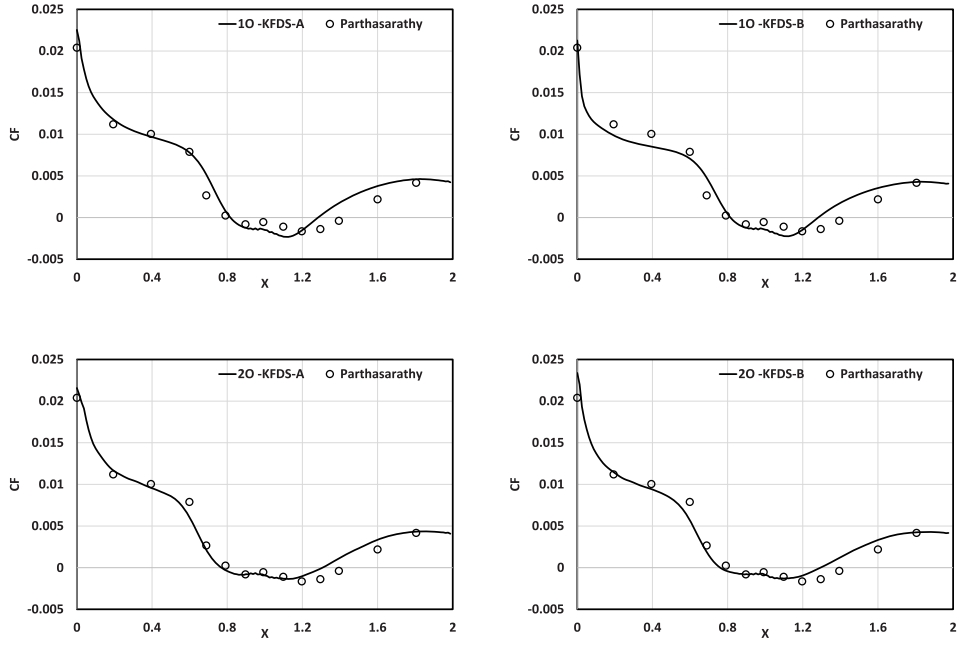

Fig. 24. Test case: Supersonic viscous flow over a bump (240 × 80), C_f plots — (a) 1O-KFDS-A (b) 1O-KFDS-B (c) 2O-KFDS-A (d) 2O-KFDS-B.

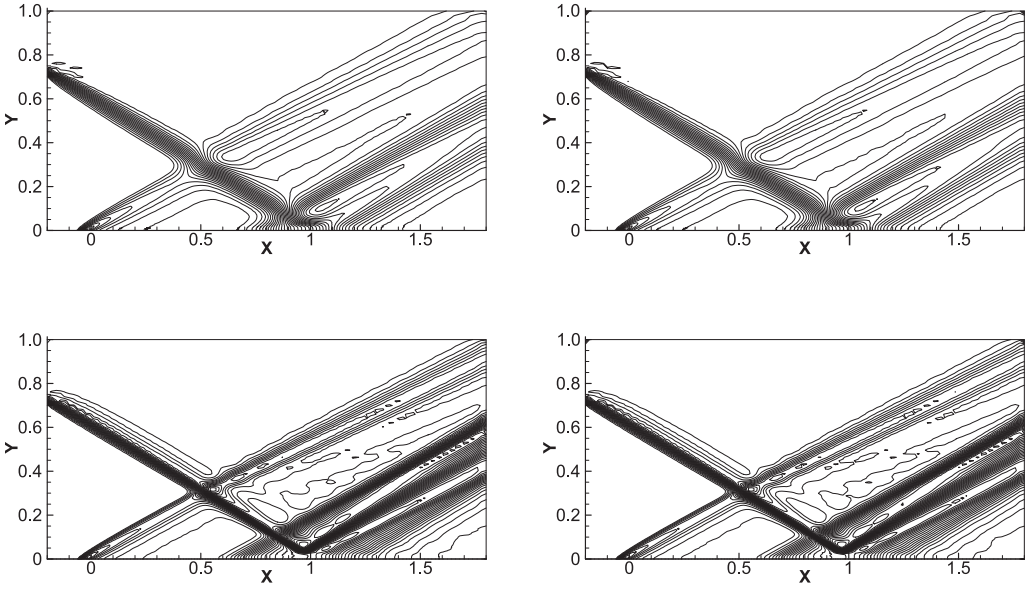

Fig. 25. Test case: Shock wave–boundary layer interaction (141 × 121) — pressure contours (a) 1O-KFDS-A (b) 1O-KFDS-B (c) 2O-KFDS-A (d) 2O-KFDS-B.

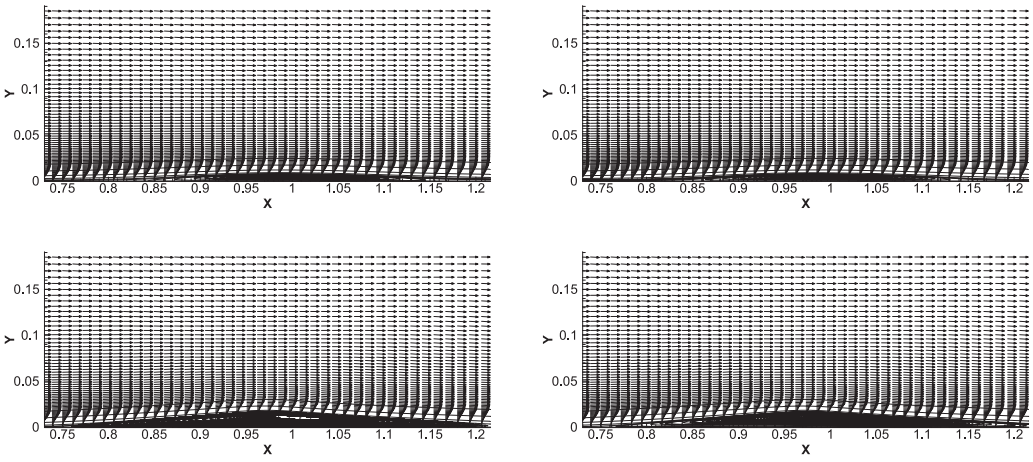

Fig. 26. Test case: Shock wave–boundary layer interaction (141 × 121) — pressure contours (a) 1O-KFDS-A (b) 1O-KFDS-B (c) 2O-KFDS-A (d) 2O-KFDS-B.

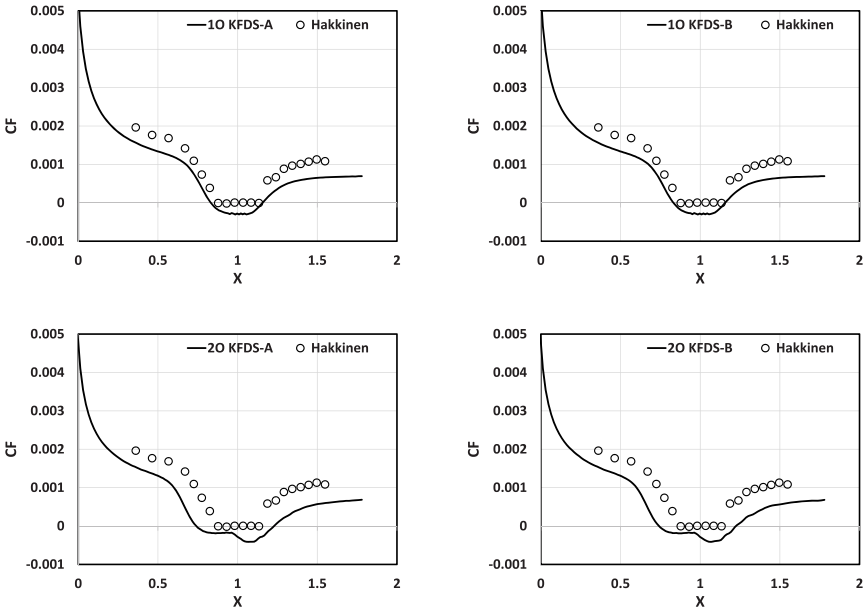

Fig. 27. Test case: Shock wave–boundary layer interaction (141 × 121) – C_f plots – (a) 1O-KFDS-A (b) 1O-KFDS-B (c) 2O-KFDS-A (d) 2O-KFDS-B.

numerical diffusion only in the expansion regions and thus helps avoiding any entropy condition violations. Various benchmark test cases for inviscid and viscous flows demonstrate the robustness and accuracy of the scheme.

CRediT authorship contribution statement

K.S. Shrinath: Methodology, Conceptualization, Software, Validation, Formal analysis, Writing – original draft. **N.H. Maruthi:** Software, Validation, Formal analysis. **S.V. Raghurama Rao:** Conceptualization, Methodology, Formal analysis, Investigation, Supervision, Writing – review & editing, Resources, Project administration. **Veeredhi Vasudeva Rao:** Conceptualization, Methodology, Project administration, Funding acquisition, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgements

The third author thanks the University of South Africa, Johannesburg, South Africa for funding during his sabbatical leave, which facilitated the formulation of the new algorithm presented in this paper. The first author thanks Hindustan Aeronautics Limited, Bangalore for funding the author, which facilitated this research activity.

Appendix. Some important moments for DVBE

$$\mathbf{P}\mathbf{f}^{eq} = U \tag{A.1}$$

$$\mathbf{P}\Lambda\mathbf{f}^{eq} = G(U) \tag{A.2}$$

$$G^{+}(U) = \mathbf{P}\Lambda^{+}\mathbf{f}^{eq} \tag{A.3}$$

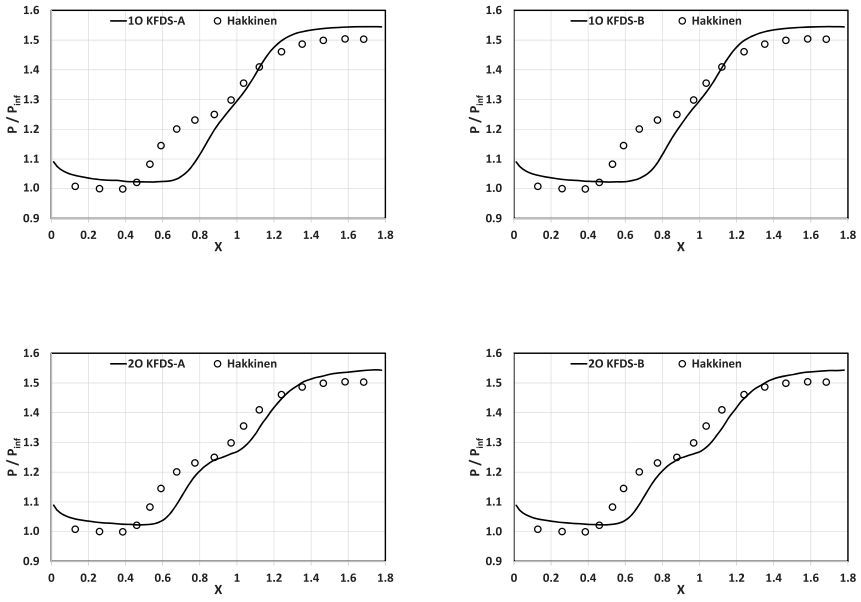

Fig. 28. Test case: Shock wave–boundary layer interaction (141 × 121) – pressure coefficient plots – (a) 1O-KFDS-A (b) 1O-KFDS-B (c) 2O-KFDS-A (d) 2O-KFDS-B.

Therefore

$$\begin{aligned} G^{+}(U)_i &= [1\ 1\ 1]\begin{bmatrix}\lambda^{+} & 0 & 0\\ 0 & \lambda_o^{+} & 0\\ 0 & 0 & 0\end{bmatrix}_i\begin{bmatrix}\mathbf{f}_{+}^{\,eq}\\ \mathbf{f}_{\mathbf{o}}^{\,eq}\\ \mathbf{f}_{-}^{\,eq}\end{bmatrix}_i \\ &= \left\{\lambda^{+}\mathbf{f}_{+}^{\,eq} + \lambda_o^{+}\mathbf{f}_{\mathbf{o}}^{\,eq}\right\}_i \end{aligned}$$

which upon simplifying gives

$$G^{+}(U)_i = \left\{\frac{1}{2}G(U) + \frac{\lambda}{2}U - \frac{(\lambda - |\lambda_o|)}{2}\mathbf{f}_{\mathbf{o}}^{\,eq}\right\}_i \tag{A.4}$$

Similarly, starting with $G^{-}(u) = \mathbf{P}\Lambda^{-}\mathbf{f}^{eq}$, we get

$$\begin{aligned} G^{-}(U)_i &= [1\ 1\ 1]\begin{bmatrix}0 & 0 & 0\\ 0 & \lambda_o^{-} & 0\\ 0 & 0 & \lambda^{-}\end{bmatrix}_i\begin{bmatrix}\mathbf{f}_{+}^{\,eq}\\ \mathbf{f}_{\mathbf{o}}^{\,eq}\\ \mathbf{f}_{-}^{\,eq}\end{bmatrix}_i \\ &= \left\{\lambda_o^{-}\mathbf{f}_{\mathbf{o}}^{\,eq} + \lambda^{-}\mathbf{f}_{-}^{\,eq}\right\}_i \end{aligned}$$

which upon simplifying gives

$$G^{-}(U)_i = \left\{\frac{1}{2}G(U) - \frac{\lambda}{2}U + \frac{(\lambda - |\lambda_o|)}{2}\mathbf{f}_{\mathbf{o}}^{\,eq}\right\}_i \tag{A.5}$$

Further

$$\begin{aligned} \mathbf{P}|\Lambda|\mathbf{f}_i^{eq} &= [1\ 1\ 1]\begin{bmatrix}\lambda & 0 & 0\\ 0 & \lambda_o & 0\\ 0 & 0 & \lambda\end{bmatrix}\begin{bmatrix}\mathbf{f}_{+}^{\,eq}\\ \mathbf{f}_{\mathbf{o}}^{\,eq}\\ \mathbf{f}_{-}^{\,eq}\end{bmatrix}_i \\ &= \left\{\lambda\mathbf{f}_{+}^{\,eq} + \lambda_o\mathbf{f}_{\mathbf{o}}^{\,eq} + \lambda\mathbf{f}_{-}^{\,eq}\right\}_i \\ &= \left\{\lambda U - (\lambda - |\lambda_o|)\mathbf{f}_{\mathbf{o}}^{\,eq}\right\}_i \end{aligned}$$

Similarly

$$\mathbf{P}\mathbf{f}^{CE} = \mathbf{P}\mathbf{f}^{eq} - \mathbf{P}\mathbf{f}_{\mathbf{v}}^{\,eq} = U \tag{A.6}$$

$$\mathbf{P}\Lambda\mathbf{f}^{CE} = \mathbf{P}\Lambda\mathbf{f}^{eq} - \mathbf{P}\Lambda\mathbf{f}_{\mathbf{v}}^{\,eq} = G(U) - G_v(U) \tag{A.7}$$

References

[1] Chu CK. Kinetic-theoretic description of the formation of a shock wave. Phys Fluids 1965;8(1):12–22.
[2] Sanders RH, Prendergast KH. The possible relation of the 3-kilospace arm to explosions in the galactic nucleus. Astrophys J 1974;188:489–500.
[3] van Albada GD, van Leer B, Roberts WW. A comparative study of computational methods in cosmic gas dynamics. Astron Astrophys 1982;108:76–84.
[4] Pullin DI. Direct simulation methods for compressible gas flow. J Comput Phys 1980;34:231–44.
[5] Reitz RD. One-dimensional compressible gas dynamics calculations using the Boltzmann equation. J Comput Phys 1981;42:108–23.
[6] Harten A, Lax P, van Leer B. On upstream differencing and Godunov-type schemes for hyperbolic conservation laws. SIAM Rev 1983;25(1):35–61.
[7] Deshpande SM. A second-order accurate kinetic theory based method for inviscid compressible flows. NASA technical paper 2613, 1986.
[8] Deshpande SM. Kinetic theory based new upwind methods for inviscid compressible flows. AIAA paper no. 86-0275, 1986.
[9] Mandal JC, Deshpande SM. Kinetic flux vector splitting for Euler equations. Comput & Fluids 1994;23(2):447–78.
[10] Kaniel S. A kinetic model for compressible flow equations. Indiana Univ Math J 1988;37(3):537–63.
[11] Perthame B. Boltzmann-type schemes for gas dynamics and entropy property. SIAM J Numer Anal 1990;27(6):1405–21.
[12] Prendergast KH, Xu Kun. Numerical hydrodynamics from gas-kinetic theory. J Comput Phys 1993;109:53–66.
[13] Raghurama Rao SV, Deshpande SM. Peculiar velocity based upwind method for inviscid compressible flows. Comput Fluid Dyn J 1995;3(4):415–32.
[14] Raghurama Rao SV. Peculiar velocity based upwind method for inviscid compressible flows. Lecture notes in physics, vol. 453, Berlin: Springer; 1995, p. 112–6.
[15] Steger JL, Warming RF. Flux vector splitting of the inviscid gasdynamic equations with applications to finite difference methods. J Comput Phys 1981;40(2):263–93.
[16] van Leer B. Flux vector splitting for the Euler equations. Lecture notes in physics, Springer-Verlag; 1982, p. 507–12, no. 170.
[17] Natalini R. A discrete kinetic approximation of entropy solutions to multidimensional scalar conservation laws. J Differential Equations 1998;148:292–317.
[18] Aregba-Driollet D, Natalini R. Discrete kinetic schemes for multidimensional systems of conservation laws. SIAM J Numer Anal 2000;37(6):1973–2004.
[19] Jin S, Xin Z. The relaxation schemes for systems of conservation laws in arbitrary space dimensions. Comm Pure Appl Math 1995;48:235–77.
[20] Bouchut F, Guarguaglini FR, Natalini R. Diffusive BGK approximations for nonlinear multidimensional parabolic equations. Indiana Univ Math J 2000;49:723–49.
[21] Raghurama Rao SV, Balakrishna K. An accurate shock capturing algorithm with a relaxed system for hyperbolic conservation laws. AIAA paper no. AIAA-2003-4115, 2003.
[22] Raghurama Rao SV, Subba Rao MV. A simple multi-dimensional relaxation scheme for hyperbolic conservation laws. AIAA paper no. AIAA-2003-3535, 2003.
[23] Bajpayi Mayank, Raghurama Rao SV. A finite variable difference relaxation scheme for hyperbolic-parabolic equations. J Comput Phys 2009;228(3):7513–42.
[24] Arun KR, Raghurama Rao SV, Lukáčová-Medvid'ová M, Prasad Phoolan. A genuinely multi-dimensional relaxation scheme for hyperbolic conservation laws. In: Proceedings of the seventh ACFD conference. Bangalore: Indian Institute of Science; 2007, p. 1029–39.
[25] Arun KR, Lukáčová-Medvid'ová M, Prasad Phoolan, Raghurama Rao SV. A second order accurate kinetic relaxation scheme for inviscid compressible flows. In: Recent developments in numerics of nonlinear hyperbolic conservation laws. Notes on numerical fluid mechanics and multidisciplinary design, vol. 120, Springer-Verlag; 2013, p. 1–24.

[26] Arun KR, Lukáčová-Medvid'ová M. A characteristics based genuinely multi-dimensional discrete kinetic scheme for the Euler equations. J Sci Comput 2013;55:40–64.
[27] Venkata Raghavendra N. Discrete velocity boltzmann schemes for inviscid compressible flows (Ph.D. thesis), Bangalore: Department of Aerospace Engineering, Indian Institute of Science; 2017.
[28] Venkata Raghavendra N, Raghurama Rao SV. A Boltzmann scheme with physically relevant discrete velocities for Euler equations. Int J Adv Eng Sci Appl Math 2021;13(2–3):305–28.
[29] Abgrall R, Torlo D. High order asymptotic preserving deferred correction implicit-explicit schemes for kinetic models. SIAM J Sci Comput 2020;42(3):B816–45.
[30] Raghurama Rao SV, Deshmukh Rohan L, Kotnala Sourabh. A lattice Boltzmann relaxation scheme for inviscid compressible flows. 2015, arXiv preprint, arXiv: 1504.04089 [math.NA].
[31] Deshmukh Rohan L. Lattice Boltzmann relaxation scheme for high speed flows (Ph.D. thesis), Bangalore: Department of Aerospace Engineering, Indian Institute of Science; 2016.
[32] Kullback S. Information theory and statistics. Dover Publications; 1978.
[33] Mahalanobis PC. On the generalized distance in statistics. Proc Natl Inst Sci India 1936;2(1):49–55.
[34] Raghavendra NV, Varma Mohan, Uthup Biju, Deshpande SM. 3-D grid adaptation using a sensor based on directed divergence between Maxwellians, computational fluid dynamics 2000. In: Satofuka N, editor. Proceedings of ICCFD 2000. Springer; 2001, p. 67–72.
[35] Bhatnagar PL, Gross EP, Krook M. A model for collision processes in gases. I. Small amplitude processes in charged and neutral one-component systems. Phys Rev 1954;94:511–25.
[36] Junk Michael, Raghurama Rao SV. A new discrete velocity method for Navier–Stokes equations. J Comput Phys 1999;155:178–98.
[37] Manoj Kumar B, Raghurama Rao SV, Balasubramanyam R, Deshpande SM. Kinetic flux vector splitting for Navier–Stokes equations based on Chapman-Enskog distribution. Fluid mechanics reports, 97 FM 9, Bangalore, India: Department of Aerospace Engineering, Indian Institute of Science; 1997.
[38] Chou SY, Baganoff D. Kinetic flux vector splitting for the Navier–Stokes equations. J Comput Phys 1997;130:217–30.
[39] Deshpande SM, Ramanan S, Manoj Kumar B, Anandhanarayanan K, Raghurama Rao SV. New Boltzmann schemes for viscous flows. In: Asian conference in CFD (ACFD3). Bangalore; 1998, p. 7–11.
[40] Jaisankar S, Raghurama Rao SV. A central Rankine–Hugoniot solver for hyperbolic conservation laws. J Comput Phys 2009;228:770–98.
[41] Venkata Raghavendra N. D^2-distance based 3-d grid adaptation for a generic fighter aircraft wing (ME thesis), Bangalore: Department of Aerospace Engineering, Indian Institute of Science; 2000.
[42] Ramesh K. Novel, robust and accurate central solvers for real, dense and multicomponent gases (Ph.D. thesis), Bangalore, India: Department of Aerospace Engineering, Indian Institute of Science; 2019.
[43] Ramesh K, Raghavendra NV, Raghurama Rao SV, Sekhar GN. Simple and robust contact-discontinuity capturing central algorithms for high speed compressible flows. Appl Math Comput 2022;414:126660.
[44] Leveque RJ. Numerical methods for conservation laws. Birkhäuser; 1990.
[45] Laney CB. Computational gasdynamics. Cambridge University Press; 1998.
[46] Toro EF. Riemann solvers and numerical methods for fluid dynamics: a practical introduction. 3rd ed.. Springer-Verlag; 2009.
[47] Abgrall R, Shu Chi-Wang, editors. Handbook of numerical methods for hyperbolic problems: basic and fundamental issues. North Holland; 2016.
[48] Zhang LQ, Chen Z, Yang LM, Shu C. An improved discrete gas-kinetic scheme for two-dimensional viscous incompressible and compressible flows. Phys Fluids 2019;31:066103.
[49] Jayaraj. A novel multi-dimensional relaxation scheme for hyperbolic conservation laws (M.Tech. thesis), Davanagere, Karnataka, India: Department of Mechanical Engineering, University B.D.T. College of Engineering; 2006.
[50] Manna M. A three-dimensional high resolution upwind finite volume Euler solver. Technical report, von Karman Institute for Fluid Dynamics; 1992.
[51] Yee H, Warming R, Harten A. A high-resolution numerical technique for inviscid gas-dynamic problems with weak solutions. In: Eighth international conference on numerical methods in fluid dynamics. Lecture notes in physics, vol. 170, Springer-Verlag; 1982, p. 546–52.
[52] Levy D, Powell KG, van Leer B. Use of a rotated Riemann solver for two-dimensional Euler equations. J Comput Phys 1993;106:201–14.
[53] Ripley RC, Lien FS, Yoyanivich MM. Adaptive unstructured mesh refinement of supersonic channel flows. Int J Comput Fluid Dyn 2004;18:189–98.
[54] Quirk JJ. A contribution to the great Riemann solver debate. Int J Numer Methods Fluids 1994;18:555–74.
[55] Parthasarathy V, Kallinderis Y. Directional viscous multigrid using adaptive prismatic meshes. AIAA J 1995;33(1).
[56] Degrez G, Boccadorosand CH, Wendt JF. The interaction of an oblique shock wave with a laminar boundary layer revisited. An experimental and numerical study. J Fluid Mech 1987;177:247–63.