INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2006; **51**:159–196
Published online 15 November 2005 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1099

A grid-free upwind relaxation scheme for inviscid compressible flows

S. Balasubramanyam*,†,‡ and S. V. Raghurama Rao§

ARDB Centre of Excellence for Aerospace CFD, Department of Aerospace Engineering,
Indian Institute of Science, Bangalore 560 012, India

SUMMARY

A new *grid-free upwind relaxation scheme* for simulating inviscid compressible flows is presented in this paper. The non-linear conservation equations are converted to linear convection equations with non-linear source terms by using a *relaxation system* and its interpretation as a discrete Boltzmann equation. A splitting method is used to separate the convection and relaxation parts. *Least squares upwinding* is used for discretizing the convection equations, thus developing a grid-free scheme which can operate on any arbitrary distribution of points. The scheme is grid free in the sense that it works on any arbitrary distribution of points and it does not require any topological information like elements, faces, edges, etc. This method is tested on some standard test cases. To explore the power of the grid-free scheme, solution-based adaptation of points is done and the results are presented, which demonstrate the efficiency of the new grid-free scheme. Copyright © 2005 John Wiley & Sons, Ltd.

KEY WORDS: relaxation system; least squares upwinding; grid-free schemes; diagonal form; discrete kinetic system; inviscid compressible flows

1. INTRODUCTION

For solving inviscid compressible flows numerically, upwind methods have gained popularity in the last two decades. The upwind methods developed for solving Euler equations can be classified into three categories, namely *flux vector splitting* schemes, Riemann solvers (exact or approximate) and *kinetic-theory-based* schemes. Comprehensive reviews of the first two approaches are available in References [1–4]. Reviews of *upwind* methods based on *kinetic theory* are given by Deshpande [5], Godlewski and Raviart [6]. Solving a Riemann problem across the interface of a finite volume is the most popular of all these approaches. Even these

*Correspondence to: S. Balasubramanyam, Fluent India Pvt. Ltd., Plot No. 34/1, Pune Infotech Park, M.I.D.C., Hinjewadi, Pune 411 057, India.
†E-mail: sasanapuri@yahoo.co.uk
‡Research Student.
§Assistant Professor.

Received 11 July 2003
Revised 24 January 2005
Accepted 18 August 2005

methods have their own shortcomings (see Reference [7]) and the search for an ideal Euler solver is still continuing.

Recently, Jin and Xin [8] have introduced a fourth category of upwind methods called relaxation schemes. A relaxation system converts a non-linear convection equation into linear convection equations with non-linear source terms. The numerical methods based on a *relaxation system* are termed as *relaxation schemes*. These schemes avoid complicated Riemann solvers and flux splitting methods. In the present work this research is continued further by developing a new grid-free relaxation scheme (see References [9–14]).

Grid generation is one of the important tasks in computational fluid dynamics (CFD). Grid generation around complex geometries is a difficult and time-consuming task. To reduce the efforts required in grid generation, in the recent past there has been a search for grid-free schemes. One important result in this search is the *least squares kinetic upwind method*, developed by Ghosh and Deshpande [15, 16]. In the present work the ideas of a relaxation approximation and least squares upwinding are combined to develop a new *grid-free upwind relaxation scheme* for inviscid compressible flows. Here, the term grid free (also referred as meshless, mesh free and gridless) is used in the sense that the scheme works on any arbitrary distribution of points. The solver does not need any topological information like elements, faces, edges, etc. All that the solver needs is a distribution of points in the flow domain and on boundaries and a set of neighbours around each point. Such grid-free solvers can also work easily on any combination of several grids (chimera grids), whether structured or unstructured, where traditional finite volume methods are known to encounter difficulties or may even fail.

2. RELAXATION SYSTEM FOR HYPERBOLIC EQUATIONS

2.1. Relaxation system of Jin and Xin

Consider a scalar conservation equation in 1D,

$$\frac{\partial u}{\partial t} + \frac{\partial g(u)}{\partial x} = 0 \tag{1}$$

with the initial condition given by

$$u(x, t=0) = u_0(x) \tag{2}$$

where $g(u)$ is a non-linear function of u. The relaxation system of Jin and Xin [8] for (1) is given by

$$\left.\begin{aligned} \frac{\partial u}{\partial t} + \frac{\partial v}{\partial x} &= 0 \\ \frac{\partial v}{\partial t} + \lambda^2 \frac{\partial u}{\partial x} &= -\frac{1}{\varepsilon}[v - g(u)] \end{aligned}\right\} \textit{relaxation system} \tag{3}$$

with the initial conditions given by

$$u(x, t=0) = u_0(x), \quad v(x, t=0) = g(u_0(x)) \tag{4}$$

Here, v is a new variable, λ is a positive constant and ε is a small positive constant called *relaxation parameter*. By re-arranging, the second equation of (3) can be written as

$$\varepsilon\left[\frac{\partial v}{\partial t}+\lambda^2\frac{\partial u}{\partial x}\right]=-\left[v-g(u)\right] \tag{5}$$

In the limit of $\varepsilon \to 0$, (5) leads to

$$v=g(u) \tag{6}$$

By substituting (6) in the first equation of the relaxation system (3), we get back the original conservation law (1). Therefore, solving (3) with $\varepsilon \to 0$ is equivalent to solving (1). The advantage in dealing with the relaxation system (3) instead of the original non-linear conservation equation (1) is that the convection terms (the left-hand side) of (3) are linear. The non-linear source term (the right-hand side) in (3) can be separated by using a splitting method [8].

In the initial conditions (4), $v(x,t=0)=g(u_0(x))$ leads to initial local equilibrium and avoids the development of an initial layer [8]. In the next section, a Chapman–Enskog-type expansion is used to obtain the condition under which the relaxation system (3) is a dissipative approximation to the original conservation law (1).

2.2. *Chapman–Enskog-type expansion for the relaxation system*

A Chapman–Enskog-type expansion for the relaxation system will show the condition under which the relaxation system is a dissipative approximation to the given conservation law. Chapman–Enskog-type expansion (see References [11, 17] for derivation) for the relaxation system (3) results in the following equation:

$$\frac{\partial u}{\partial t}+\frac{\partial g(u)}{\partial x}=\varepsilon\frac{\partial}{\partial x}\left[\left\{\lambda^2-\left(\frac{\partial g(u)}{\partial u}\right)^2\right\}\frac{\partial u}{\partial x}\right]+O(\varepsilon^2) \tag{7}$$

The right-hand side of (7) contains a second derivative of u, and hence represents a viscous (or dissipation) term. The coefficient represents the coefficient of viscosity. Therefore, the relaxation system provides a vanishing viscosity model for the original conservation law (1). For the relaxation system (3) to be a dissipative approximation for the *conservation law* (1) (i.e. for the coefficient of viscosity to be positive) the following condition has to be satisfied:

$$\lambda^2-\left(\frac{\partial g(u)}{\partial u}\right)^2\geqslant 0 \quad\text{or}\quad \lambda^2\geqslant\left(\frac{\partial g(u)}{\partial u}\right)^2 \tag{8}$$

$$\text{or}\quad -\lambda\leqslant\left(\frac{\partial g(u)}{\partial u}\right)\leqslant\lambda \tag{9}$$

The constant λ in the relaxation system (3) is chosen such that (9) is satisfied.

2.3. Diagonal form of relaxation system

Consider the scalar conservation law (1) and the relaxation system (3). The relaxation system (3) can be re-written in vector form as

$$\frac{\partial Q}{\partial t} + A\frac{\partial Q}{\partial x} = H \tag{10}$$

where

$$Q = \begin{bmatrix} u \\ v \end{bmatrix}, \quad A = \begin{bmatrix} 0 & 1 \\ \lambda^2 & 0 \end{bmatrix} \quad \text{and} \quad H = \begin{bmatrix} 0 \\ -\frac{1}{\varepsilon}[v - g(u)] \end{bmatrix} \tag{11}$$

Since system (10) is hyperbolic, matrix A can be diagonalized as

$$A = R\Lambda R^{-1} \quad \text{or} \quad \Lambda = R^{-1}AR \tag{12}$$

where R is the right eigenvector of A and Λ is the diagonal matrix with the diagonal elements being the eigenvalues of A.

$$R = \begin{bmatrix} 1 & 1 \\ -\lambda & \lambda \end{bmatrix}, \quad \Lambda = \begin{bmatrix} -\lambda & 0 \\ 0 & \lambda \end{bmatrix} \quad \text{and} \quad R^{-1} = \begin{bmatrix} \frac{1}{2} & -\frac{1}{2\lambda} \\ \frac{1}{2} & \frac{1}{2\lambda} \end{bmatrix} \tag{13}$$

By introducing characteristic variables, the relaxation system (10) can be de-coupled as given below. By defining the characteristic variables as

$$\mathbf{f} = R^{-1}Q \tag{14}$$

we obtain the following diagonal system:

$$\frac{\partial \mathbf{f}}{\partial t} + \Lambda\frac{\partial \mathbf{f}}{\partial x} = R^{-1}H \tag{15}$$

From (13) and (11), we obtain

$$\mathbf{f} = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} = R^{-1}Q = \begin{bmatrix} \frac{1}{2}u - \frac{1}{2\lambda}v \\ \frac{1}{2}u + \frac{1}{2\lambda}v \end{bmatrix} \quad \text{and} \quad R^{-1}H = \begin{bmatrix} \frac{1}{2\lambda\varepsilon}[v - g(u)] \\ -\frac{1}{2\lambda\varepsilon}[v - g(u)] \end{bmatrix} \tag{16}$$

Substituting (16) in (15), the following set of de-coupled equations are obtained:

$$\begin{aligned} \frac{\partial f_1}{\partial t} - \lambda\frac{\partial f_1}{\partial x} &= \frac{1}{2\varepsilon\lambda}[v - g(u)] \\ \frac{\partial f_2}{\partial t} + \lambda\frac{\partial f_2}{\partial x} &= -\frac{1}{2\varepsilon\lambda}[v - g(u)] \end{aligned} \tag{17}$$

In the limit of $\varepsilon \to 0$ the de-coupled system (17) is equivalent to the conservation law (1). The left-hand side of the two de-coupled hyperbolic equations are linear, with constant wave speeds neatly split into positive and negative parts. Constructing an upwind scheme for the above system is much simpler compared to the task of developing an upwind scheme for the conservation law (1), when $g(u)$ is a non-linear function of u. From (16), the following expressions can be obtained for u and v:

$$u = f_1 + f_2 \quad \text{and} \quad v = \lambda(f_2 - f_1) \tag{18}$$

Using the above expressions, the original variables of the relaxation system, u and v, can be recovered.

2.4. Diagonal form as discrete kinetic system

The diagonal form of the relaxation system can be interpreted as a *discrete kinetic system*. Using expressions (18) and defining new variables F_1 and F_2 as

$$F_1 = \frac{1}{2}u - \frac{1}{2\lambda}g(u) \quad \text{and} \quad F_2 = \frac{1}{2}u + \frac{1}{2\lambda}g(u) \tag{19}$$

the de-coupled system (17) can be re-written as

$$\begin{aligned} \frac{\partial f_1}{\partial t} - \lambda\frac{\partial f_1}{\partial x} &= \frac{1}{\varepsilon}[F_1 - f_1] \\ \frac{\partial f_2}{\partial t} + \lambda\frac{\partial f_2}{\partial x} &= \frac{1}{\varepsilon}[F_2 - f_2] \end{aligned} \tag{20}$$

or equivalently as

$$\frac{\partial \mathbf{f}}{\partial t} + \Lambda\frac{\partial \mathbf{f}}{\partial x} = \frac{1}{\varepsilon}[\mathbf{F} - \mathbf{f}] \tag{21}$$

where

$$\mathbf{f} = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} \quad \text{and} \quad \mathbf{F} = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix} \tag{22}$$

and Λ is as defined in (13). Here, the initial conditions (4) can be re-written as

$$u(x, t=0) = u_0(x) \quad \text{and} \quad \mathbf{f}(t=0) = \mathbf{F}(u_0(x)) \tag{23}$$

Equation (21) is similar to the classical Boltzmann equation with B–G–K collision model, where ε represents the relaxation time. The difference between the two is that Equation (21) contains two discrete velocities $(-\lambda, \lambda)$ whereas the molecular velocity in the classical Boltzmann equation is continuous. f_1 and f_2 represent the corresponding components of the distribution function. This interpretation was used by Natalini and Driollet [18, 19] to develop numerical schemes using relaxation system and Equation (21) is called *discrete Boltzmann equation*. The discrete Boltzmann equation in 1D is generalized to multi-dimensions by Driollet and Natalini [19] as the relaxation system of Jin and Xin is not diagonalizable in multi-dimensions and it is preferable to work with the diagonal form.

2.5. Relaxation system for Euler equations in 1D

Consider the Euler equations in 1D,

$$\frac{\partial \mathbf{u}}{\partial t}+\frac{\partial \mathbf{g}(\mathbf{u})}{\partial x}=0 \tag{24}$$

with the initial condition

$$\mathbf{u}(x,t=0)=\mathbf{u}_0(x) \tag{25}$$

where

$$\mathbf{u}=\begin{bmatrix}\rho\\ \rho u\\ \rho E\end{bmatrix},\quad \mathbf{g}(\mathbf{u})=\begin{bmatrix}\rho u\\ p+\rho u^2\\ pu+\rho uE\end{bmatrix}\quad \text{and}\quad E=\frac{p}{\rho(\gamma-1)}+\frac{u^2}{2} \tag{26}$$

2.5.1. Relaxation system of Jin and Xin. Introducing a new variable vector $\mathbf{v}$, the relaxation system of Jin and Xin [8] for (24) can be written as

$$\begin{aligned}\frac{\partial \mathbf{u}}{\partial t}+\frac{\partial \mathbf{v}}{\partial x}&=0\\ \frac{\partial \mathbf{v}}{\partial t}+D\frac{\partial \mathbf{u}}{\partial x}&=-\frac{1}{\varepsilon}[\mathbf{v}-\mathbf{g}(\mathbf{u})]\end{aligned} \tag{27}$$

with the initial conditions given by

$$\mathbf{u}(x,t=0)=\mathbf{u}_0(x),\quad \mathbf{v}(x,t=0)=\mathbf{g}(\mathbf{u}_0(x)) \tag{28}$$

Here, D is a constant diagonal matrix with positive elements, which is to be chosen in such a way that approximation (27) is dissipative.

$$D=\begin{bmatrix}\lambda_1^2 & 0 & 0\\ 0 & \lambda_2^2 & 0\\ 0 & 0 & \lambda_3^2\end{bmatrix} \tag{29}$$

In the limit of $\varepsilon\to 0$, solving the relaxation system (27) is equivalent to solving the Euler equations (24). The special initial condition on $\mathbf{v}$ (28) avoids the development of an initial layer, as the initial state is in local equilibrium [8]. Doing Chapman–Enskog-type expansion, based on the eigenvalues of the flux Jacobian matrix of Euler equations, Jin and Xin [8] proposed the following two choices:

$$\text{(i)}\ \lambda^2=\lambda_1^2=\lambda_2^2=\lambda_3^2=\max[\max|u-a|,\max|u|,\max|u+a|] \tag{30}$$

or

$$\text{(ii)}\ \lambda_1^2=\max|u-a|,\quad \lambda_2^2=\max|u|\quad\text{and}\quad \lambda_3^2=\max|u+a| \tag{31}$$

Here, u is the fluid velocity and a is the acoustic speed. The above *maxima* are defined over the whole domain. With the first choice, the diagonal matrix D can be written as

$$D=\lambda^2 I \tag{32}$$

where I is a unit matrix of order 3.

2.5.2. Discrete kinetic system. Discrete kinetic system for the 1D Euler equations is similar to the discrete kinetic system for the scalar conservation law given in Section 2.4, except for the fact that the characteristic variables $\mathbf{f}_1$, $\mathbf{f}_2$ and the discrete distribution functions $\mathbf{F}_1$, $\mathbf{F}_2$ are vectors in this case. Discrete kinetic system for this case is given by

$$\frac{\partial \mathbf{f}}{\partial t} + \Lambda \frac{\partial \mathbf{f}}{\partial x} = \frac{1}{\varepsilon}[\mathbf{F} - \mathbf{f}] \tag{33}$$

where

$$\mathbf{f} = \begin{bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \end{bmatrix} = \begin{bmatrix} \frac{1}{2}\mathbf{u} - \frac{1}{2\lambda}\mathbf{v} \\ \frac{1}{2}\mathbf{u} + \frac{1}{2\lambda}\mathbf{v} \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} \mathbf{F}_1 \\ \mathbf{F}_2 \end{bmatrix} = \begin{bmatrix} \frac{1}{2}\mathbf{u} - \frac{1}{2\lambda}\mathbf{g}(\mathbf{u}) \\ \frac{1}{2}\mathbf{u} + \frac{1}{2\lambda}\mathbf{g}(\mathbf{u}) \end{bmatrix} \quad \text{and} \quad \Lambda = \begin{bmatrix} -\lambda & 0 \\ 0 & \lambda \end{bmatrix} \tag{34}$$

In 2D, the generalization of 1D discrete kinetic system (33) is used as the relaxation system of Jin and Xin is not diagonalizable in multi-dimensions [19].

2.5.3. Discreet Boltzmann equation in comparison with classical Boltzmann equation. The discrete Boltzmann equation (33) is similar to the classical Boltzmann equation of the *kinetic theory* of gases:

$$\frac{\partial f}{\partial t} + v\frac{\partial f}{\partial x} = \frac{1}{\tau}(F - f) \tag{35}$$

where f is now the molecular velocity distribution function, v is the molecular velocity and the right-hand side is the B–G–K collision model with τ as the relaxation time during which the distribution function f relaxes to the local Maxwellian distribution F. In the *kinetic theory*, the expression for the local Maxwellian is given by

$$F = \frac{\rho}{I_0}\frac{\sqrt{\pi}}{\sqrt{\beta}}\, \mathrm{e}^{-\beta(v-u)^2-(I/I_0)} \tag{36}$$

where $\beta = 1/2RT$. Here, I is the internal energy variable for non-translational degrees of freedom and I_0 is the corresponding average internal energy. The expression for the local Maxwellian in the discrete kinetic system is given by (34), which can be re-written as

$$\mathbf{F}_1 = \frac{1}{2}\left[\mathbf{u} - \frac{1}{\lambda}\mathbf{g}(\mathbf{u})\right]; \quad \mathbf{F}_2 = \frac{1}{2}\left[\mathbf{u} + \frac{1}{\lambda}\mathbf{g}(\mathbf{u})\right] \quad \text{where } \lambda^2 I = D \tag{37}$$

We can see that the expression for the local Maxwellian for the discrete kinetic system is much simpler as it is expressed entirely in simple algebraic expressions of macroscopic variables, unlike the classical Maxwellian which contains a molecular velocity variable in an exponential term. The moments of the continuous Maxwellian or the continuous molecular velocity distribution function give the conserved variables as

$$\int_0^\infty \mathrm{d}I \int_{-\infty}^\infty \mathrm{d}v\, F = \int_0^\infty \mathrm{d}I \int_{-\infty}^\infty \mathrm{d}v\, f = u \tag{38}$$

In the discrete kinetic system, this relation is simply given as

$$\mathbf{u} = P\mathbf{F} = P\mathbf{f} \tag{39}$$

where P is a real constant coefficient matrix. In the 1D case P is given by

$$P = [1 \;\; 1] \tag{40}$$

With this definition, we obtain

$$\mathbf{u} = \sum_{i=1}^{N} \mathbf{F}_i = \sum_{i=1}^{N} \mathbf{f}_i \tag{41}$$

where N is the number of discrete velocities ($N = 2$ for the 1D case). The flux vector is recovered from the molecular velocity distribution function by multiplying with the molecular velocity and then taking moments as

$$\mathbf{g}(\mathbf{u}) = \int_0^\infty \mathrm{d}I \int_{-\infty}^{\infty} \mathrm{d}v\, vF \tag{42}$$

In the discrete kinetic system, we first recover the variable $\mathbf{v}$ as

$$\mathbf{v} = P\Lambda\mathbf{f} \tag{43}$$

and then, in the limit of $\varepsilon \to 0$, we recover the flux vector as $\mathbf{v} = g(u)$ (see (6)). Thus, the discrete kinetic system shares the spirit of the classical Boltzmann equation of *kinetic theory* and is much simpler in its expressions.

2.6. Relaxation system for Euler equations in 2D

Consider Euler equations in 2D

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial \mathbf{g}_1(\mathbf{u})}{\partial x} + \frac{\partial \mathbf{g}_2(\mathbf{u})}{\partial y} = 0 \tag{44}$$

where

$$\mathbf{u} = \begin{bmatrix} \rho \\ \rho u_1 \\ \rho u_2 \\ \rho E \end{bmatrix}, \quad \mathbf{g}_1(\mathbf{u}) = \begin{bmatrix} \rho u_1 \\ p + \rho u_1^2 \\ \rho u_1 u_2 \\ (p + \rho E) u_1 \end{bmatrix}, \quad \mathbf{g}_2(\mathbf{u}) = \begin{bmatrix} \rho u_2 \\ \rho u_2 u_1 \\ p + \rho u_2^2 \\ (p + \rho E) u_2 \end{bmatrix} \quad \text{and}$$
$$E = \frac{p}{\rho(\gamma - 1)} + \frac{u_1^2 + u_2^2}{2} \tag{45}$$

As mentioned earlier, since the relaxation system of Jin and Xin [8] for the above Euler equations is not diagonalizable, the discrete kinetic system, which is already in diagonal form, is generalized to 2D [19]. This approach can be easily extended to 3D also.

2.6.1. Discrete kinetic system. The multi-dimensional discrete kinetic system is given by

$$\frac{\partial \mathbf{f}}{\partial t} + \sum_{k=1}^{d} \Lambda_k \frac{\partial \mathbf{f}}{\partial x_k} = \frac{1}{\varepsilon}[\mathbf{F} - \mathbf{f}] \tag{46}$$

where d is the number of translational degrees of freedom, Λ_k are real diagonal $N \times N$ matrices and N is the number of discrete velocities ($N \geqslant d+1$). For the 2D case, N is taken as $N = d+1 = 3$ [19]. Therefore, the discrete kinetic system for 2D case is given by

$$\frac{\partial \mathbf{f}}{\partial t} + \Lambda_1 \frac{\partial \mathbf{f}}{\partial x} + \Lambda_2 \frac{\partial \mathbf{f}}{\partial y} = \frac{1}{\varepsilon}[\mathbf{F} - \mathbf{f}] \tag{47}$$

with

$$\begin{gathered}
\mathbf{f} = \begin{bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \mathbf{f}_3 \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} \mathbf{F}_1 \\ \mathbf{F}_2 \\ \mathbf{F}_3 \end{bmatrix} \\
\Lambda_1 = \begin{bmatrix} \lambda_{11} & 0 & 0 \\ 0 & \lambda_{12} & 0 \\ 0 & 0 & \lambda_{13} \end{bmatrix}, \quad \Lambda_2 = \begin{bmatrix} \lambda_{21} & 0 & 0 \\ 0 & \lambda_{22} & 0 \\ 0 & 0 & \lambda_{23} \end{bmatrix}
\end{gathered} \tag{48}$$

The conservative variable $\mathbf{u}$ and the new variables $\mathbf{v}_1$, $\mathbf{v}_2$ can be recovered as

$$\mathbf{u} = P\mathbf{f}, \quad \mathbf{v}_1 = P\Lambda_1\mathbf{f} \quad \text{and} \quad \mathbf{v}_2 = P\Lambda_2\mathbf{f} \tag{49}$$

where matrix P is defined by

$$P = [1 \quad 1 \quad 1] \tag{50}$$

The ‘moments’ of the local Maxwellian F are given by

$$P\mathbf{F} = \mathbf{u}, \quad P\Lambda_1\mathbf{F} = \mathbf{g}_1(\mathbf{u}) \quad \text{and} \quad P\Lambda_2\mathbf{F} = \mathbf{g}_2(\mathbf{u}) \tag{51}$$

The local Maxwellians are defined as (see Reference [19])

$$\begin{gathered}
\mathbf{F}_{d+1} = \frac{1}{d+1}\left[\mathbf{u} + \frac{1}{\lambda}\sum_{k=1}^{d} \mathbf{g}_k(\mathbf{u})\right] \quad \text{and} \\
\mathbf{F}_i = -\frac{1}{\lambda}\mathbf{g}_i(\mathbf{u}) + \mathbf{F}_{d+1} \quad \text{for } i = 1, 2, \ldots, d
\end{gathered} \tag{52}$$

Therefore, for the 2D case, the expressions for the *discrete Maxwellians* are given by

$$\begin{aligned}
\mathbf{F}_1 &= \frac{\mathbf{u}}{3} - \frac{2}{3\lambda}\mathbf{g}_1(\mathbf{u}) + \frac{1}{3\lambda}\mathbf{g}_2(\mathbf{u}) \\
\mathbf{F}_2 &= \frac{\mathbf{u}}{3} + \frac{1}{3\lambda}\mathbf{g}_1(\mathbf{u}) - \frac{2}{3\lambda}\mathbf{g}_2(\mathbf{u}) \\
\mathbf{F}_3 &= \frac{\mathbf{u}}{3} + \frac{1}{3\lambda}\mathbf{g}_1(\mathbf{u}) + \frac{1}{3\lambda}\mathbf{g}_2(\mathbf{u})
\end{aligned} \tag{53}$$

Using expressions (49) with the above definitions of local Maxwellians, the expressions for elements of the matrices P, Λ_1 and Λ_2 can be obtained as

$$P=[1 \quad 1 \quad 1], \quad \Lambda_1=\begin{bmatrix}-\lambda & 0 & 0\\ 0 & 0 & 0\\ 0 & 0 & \lambda\end{bmatrix} \quad \text{and} \quad \Lambda_2=\begin{bmatrix}0 & 0 & 0\\ 0 & -\lambda & 0\\ 0 & 0 & \lambda\end{bmatrix} \tag{54}$$

Substituting (54) in (47) the three de-coupled equations of the discrete kinetic system can be written as

$$\begin{aligned}\frac{\partial \mathbf{f}_1}{\partial t}-\lambda\frac{\partial \mathbf{f}_1}{\partial x}&=\frac{1}{\varepsilon}[\mathbf{F}_1-\mathbf{f}_1]\\ \frac{\partial \mathbf{f}_2}{\partial t}-\lambda\frac{\partial \mathbf{f}_2}{\partial y}&=\frac{1}{\varepsilon}[\mathbf{F}_2-\mathbf{f}_2]\\ \frac{\partial \mathbf{f}_3}{\partial t}+\lambda\frac{\partial \mathbf{f}_3}{\partial x}+\lambda\frac{\partial \mathbf{f}_3}{\partial y}&=\frac{1}{\varepsilon}[\mathbf{F}_3-\mathbf{f}_3]\end{aligned} \tag{55}$$

From the above three equations the splitting of information propagation along each characteristic direction can be seen very clearly. For the characteristic $\mathbf{f}_1$ the wave speed $-\lambda$ indicates that the information comes from the right side of the Y-axis. For the characteristic $\mathbf{f}_2$ the wave speed $-\lambda$ indicates that the information comes from upper side of the X-axis. For the characteristic $\mathbf{f}_3$ the wave speeds $+\lambda$ and $+\lambda$ indicate that the information comes from lower side of the X-axis and left side of the Y-axis.

2.6.2. Chapman–Enskog-type expansion. Chapman–Enskog-type expansion (see References [11, 19] for derivation) for the relaxation system in 2D gives the following condition:

$$\lambda \geqslant (-A_1-A_2, 2A_1-A_2, -A_1+2A_2) \tag{56}$$

where

$$A_1=\frac{\partial \mathbf{g}_1(\mathbf{u})}{\partial \mathbf{u}} \quad \text{and} \quad A_2=\frac{\partial \mathbf{g}_2(\mathbf{u})}{\partial \mathbf{u}} \tag{57}$$

Let us define

$$K_1=-A_1-A_2, \quad K_2=2A_1-A_2 \quad \text{and} \quad K_3=-A_1+2A_2 \tag{58}$$

To fix λ, eigenvalues for these three matrices are needed. The eigenvalues for the matrices K_1, K_2 and K_3 derived with a symbolic manipulator are given as

$$\begin{aligned}&K_1: -u-v, -u-v, -u-v \pm \sqrt{(1-\gamma)(u^2+v^2-2H)}\\ &K_2: 2u-v, 2u-v, 2u-v \pm \tfrac{1}{2}\sqrt{10(1-\gamma)(u^2+v^2-2H)}\\ &K_3: -u+2v, -u+2v, -u+2v \pm \tfrac{1}{2}\sqrt{10(1-\gamma)(u^2+v^2-2H)}\end{aligned} \tag{59}$$

where

$$H=\frac{\gamma p}{\rho(\gamma-1)}+\frac{u^2+v^2}{2} \tag{60}$$

In this work, the spectral radius of each of these matrices in (59) is found and the maximum of the three is used as λ. Here, the spectral radius means the maximum eigenvalue of the given matrix.

3. RELAXATION SCHEMES

3.1. Relaxation schemes of Jin and Xin

In this section the *relaxation schemes* of Jin and Xin are discussed briefly. Full details of these schemes can be obtained from Jin and Xin [8] and the references there in. Jin and Xin classify their schemes into two categories, namely, relaxing schemes and relaxed schemes. *Relaxing schemes* depend on ε and the artificial variable $\mathbf{v}$. The zero relaxation limit of the relaxing schemes are called relaxed schemes. The *relaxed schemes* are stable discretizations of the original conservation law, and thus are independent of ε and the artificial variable $\mathbf{v}$. Jin and Xin do upwinding based on the characteristic variables. To achieve second-order accuracy they use van Leer's MUSCL scheme. They use a second-order TVD Runge–Kutta splitting scheme for time discretization, which was introduced by Jin [20]. In this section the relaxation schemes are given for 1D relaxation system. For multi-dimensional relaxation systems, which are just the natural dimension-by-dimension extension of 1D case, the reader is referred to Reference [8].

3.2. Diagonal relaxation schemes based on discrete kinetic system

In the previous section the Relaxation schemes of Jin and Xin are discussed briefly, in which the upwinding is done based on the characteristic variables. Driollet and Natalini [19] have introduced discrete kinetic schemes, where they start with a set of de-coupled equations, in which the dependent variables are just the characteristic variables. This system of equations is discussed in Section 2.6.1. The basic advantage with this approach is that upwinding becomes simple and this approach can be easily extended to multi-dimensions.

3.3. Operator splitting

Operator-splitting methods are commonly used in many applications. The basic advantage of this approach is that one can use a specific method for each physical phenomenon (e.g. convection, diffusion, relaxation). Operator-splitting approach is also used for splitting a multi-dimensional problem into a number of one-dimensional problems. For the analysis of *operator splitting* and the related problems the reader is referred to Reference [21]. Let us consider the discrete kinetic system (21).

$$\frac{\partial \mathbf{f}}{\partial t} + \Lambda \frac{\partial \mathbf{f}}{\partial x} = \frac{1}{\varepsilon}[\mathbf{F} - \mathbf{f}] \tag{61}$$

We can re-write the above equation as

$$\frac{\partial \mathbf{f}}{\partial t} = S^{(t)}(\mathbf{f}) + C^{(t)}(\mathbf{f}) \tag{62}$$

where $S^{(t)}$ is the relaxation operator, with S denoting the source part, and $C^{(t)}$ is the convection operator, given by

$$S^{(t)} = \frac{1}{\varepsilon}[\mathbf{F} - \mathbf{f}] \quad \text{and} \quad C^{(t)} = -\Lambda \frac{\partial \mathbf{f}}{\partial x} \tag{63}$$

Using the operator splitting, the procedure for solving the discrete kinetic system can be split into two steps as

$$\textit{Step}\ \mathrm{A}: \frac{\partial \mathbf{f}}{\partial t} = S^{(t)} \ \text{(relaxation step)} \tag{64}$$

$$\textit{Step}\ \mathrm{B}: \frac{\partial \mathbf{f}}{\partial t} = C^{(t)} \ \text{(convection step)} \tag{65}$$

Now, the obvious question that follows is whether we can get a second-order accurate scheme if we use a second-order accurate spatial discretization and a second-order accurate time discretization in both the steps. In general, the answer is *no*. This problem is due to the *splitting error*, which is in general $O(\Delta t)$ for the type of splitting used above, and due to this the above splitting method will be only first-order accurate in time, no matter how accurate the approximations are in the two steps. To achieve second-order accuracy, Strang [22] has introduced a method (called as Strang's splitting method) which is discussed briefly in the next section.

3.4. Strang's splitting

A slight modification of the splitting idea discussed in the previous section will yield second-order accuracy (when each sub-problem is solved with at least second-order accuracy). The idea is to solve the first sub-problem over only a half time step of length $\Delta t/2$. Then use the result as initial data for a full time step on the second sub-problem, and use the result as the initial data and take another half time step on the first sub-problem. This approach is called Strang's splitting.

The use of Strang's splitting is to get formal second-order accuracy during a time step Δt, and applying this method to the discrete kinetic system (61), we get

$$\frac{\partial \mathbf{f}}{\partial t} = S^{(\Delta t/2)} \tag{66}$$

$$\frac{\partial \mathbf{f}}{\partial t} = C^{(\Delta t)} \tag{67}$$

$$\frac{\partial \mathbf{f}}{\partial t} = S^{(\Delta t/2)} \tag{68}$$

For time discretization, a two-stage Runge–Kutta method can be used. With this we should get a second-order accurate scheme. But in the case of a relaxation system there is a special problem called *stiffness* which is due to the very small time scale (relaxation time) involved in the relaxation step, compared to the physical time scales (and space scales) involved in the convection step. In the limit of $\varepsilon \rightarrow 0$, even this Strang splitting reduces to formal first-order accuracy (see Reference [20]). To overcome this problem Jin [20] suggests a modified Strang's splitting method to achieve formal second-order accuracy.

3.5. Problems with second-order accuracy and Jin's modified Strang's splitting

As mentioned earlier, in the stiff limit (i.e. as $\varepsilon \rightarrow 0$), the Strang's splitting method reduces to formal first-order accuracy, despite using second-order accurate discretizations in the convection and relaxation steps. Jin [20] proved this fact mathematically and developed a new modified Strang's splitting method, which gives formal second-order accuracy even in the stiff limit. For the discrete kinetic system (61), Jin's method can be given as

$$\mathbf{f}^{\star} = \mathbf{f}^{n} + a\frac{\Delta t}{\varepsilon}[\mathbf{F}^{\star} - \mathbf{f}^{\star}] \tag{69}$$

$$\mathbf{f}^{(1)} = \mathbf{f}^{\star} - \Delta t \Delta_{\text{conv}} \tag{70}$$

$$\mathbf{f}^{\star\star} = \mathbf{f}^{(1)} + b\frac{\Delta t}{\varepsilon}[\mathbf{F}^{\star\star} - \mathbf{f}^{\star\star}] + c\frac{\Delta t}{\varepsilon}[\mathbf{F}^{\star} - \mathbf{f}^{\star}] \tag{71}$$

$$\mathbf{f}^{(2)} = \mathbf{f}^{\star\star} - \Delta t \Delta_{\text{conv}} \tag{72}$$

$$\mathbf{f}^{n+1} = \tfrac{1}{2}(\mathbf{f}^{n} + \mathbf{f}^{(2)}) \tag{73}$$

where Δ_{conv} is the discrete convection operator . For full second-order accuracy, Jin [20] has shown that the constants a, b, c used in the above splitting method are given by

$$a = -1, \quad b = 1, \quad c = 2 \tag{74}$$

4. LEAST SQUARES UPWINDING

4.1. Introduction

Discretization of the domain is a necessary first step in solving a partial differential equation numerically. This is called grid generation. Grid generation for complex geometries is a difficult and time-consuming task. To minimize the efforts required for grid generation, in the recent past there has been a search for grid-free methods. In this search, a method called *least squares kinetic upwind method* is developed by Ghosh and Deshpande [15, 16]. In this work, the idea of least squares upwinding is used to develop a *grid-free relaxation scheme*. Here, the scheme is grid free in the sense that it works on points obtained by any type of grid (structured or unstructured), a combination of grids or just on any arbitrary distribution of points.

4.2. Least squares upwinding in 1D

4.2.1. First-order accurate least squares upwinding. Consider a 1D linear convection equation

$$\frac{\partial f}{\partial t} + \lambda\frac{\partial f}{\partial x} = 0 \tag{75}$$

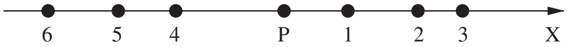

Figure 1. Arbitrary distribution of points in 1D.

To discretize the above equation, consider the computational domain consisting of an arbitrarily distributed set of points as shown in Figure 1. Let i be any neighbour point around the point of consideration, P. Taylor series expansion around P gives

$$f_i = f_P + (x_i - x_P)\left(\frac{\partial f}{\partial x}\right)_P + O[(x_i - x_P)^2] \tag{76}$$

Let the error of this approximation be denoted by e. Therefore,

$$e = f_i - f_P - (x_i - x_P)\left(\frac{\partial f}{\partial x}\right)_P \tag{77}$$

Let E be the sum of squares of the errors at all neighbouring points, i.

$$E = \sum_{i=1}^{n} e^2 \tag{78}$$

where n is the total number of neighbouring points in the stencil. Let us now introduce some notation for simplicity as

$$\Delta f_i = f_i - f_P, \quad \Delta x_i = x_i - x_P, \quad f_x = \left(\frac{\partial f}{\partial x}\right)_P \tag{79}$$

Using this notation, expressions (77) and (78) can be written as

$$e = \Delta f_i - \Delta x_i f_x \tag{80}$$

and

$$E = \sum_{i=1}^{n} (\Delta f_i - \Delta x_i f_x)^2 \tag{81}$$

Let us now minimize the sum of squares of errors with respect to the derivative, f_x.

$$\frac{\partial E}{\partial (f_x)} = 0 \quad \text{or} \quad \frac{\partial}{\partial (f_x)}\left[\sum_{i=1}^{n} (\Delta f_i - \Delta x_i f_x)^2\right] = 0 \tag{82}$$

from which we obtain

$$f_x = \left(\frac{\partial f}{\partial x}\right)_P = \frac{\sum_{i=1}^{n} \Delta f_i \Delta x_i}{\sum_{i=1}^{n} \Delta x_i^2} \tag{83}$$

Expression (83) gives an approximation for the space derivative for an arbitrary distribution of n points. But, this approximation does not give preference to any direction, and is not an upwind approximation. Therefore, this approximation will lead to an unstable scheme for the given hyperbolic equation, if coupled with forward time discretization of the time derivative, as it is like central differencing. To introduce upwinding, let us introduce *weighted least squares method.*

$$E = \sum_{i=1}^{n} w_i e^2 \tag{84}$$

If we now minimize E, we get

$$f_x = \left(\frac{\partial f}{\partial x}\right)_P = \frac{\sum_{i=1}^{n} w_i \Delta f_i \Delta x_i}{\sum_{i=1}^{n} w_i \Delta x_i^2} \tag{85}$$

Using approximation (85), we can discretize a hyperbolic equation like (75), using wave speed splitting as in C–I–R scheme, as

$$\frac{f_P^{n+1} - f_P^n}{\Delta t} + \frac{\lambda + |\lambda|}{2} \frac{\sum_{i=1}^{n} w_i \Delta f_i \Delta x_i}{\sum_{i=1}^{n} w_i \Delta x_i^2} + \frac{\lambda - |\lambda|}{2} \frac{\sum_{i=1}^{n} w_i \Delta f_i \Delta x_i}{\sum_{i=1}^{n} w_i \Delta x_i^2} = 0 \tag{86}$$

Now, we have to choose the weights w_i at each point i in such a way that the approximation is upwinding. If we substitute $w_i = 1$ everywhere, we get back the earlier formula, which is like central differencing. If we take $w_i = 1$ for upwind points and $w_i = 0$ otherwise, we can get an upwind approximation. We can do this by prescribing the weights as

$$\begin{aligned}
w_i &= 1 \quad \text{if} \quad \lambda > 0 \quad \text{and} \quad \Delta x_i < 0 \ \text{(i.e. } i \text{ is on the left side of } P)\\
w_i &= 0 \quad \text{if} \quad \lambda > 0 \quad \text{and} \quad \Delta x_i > 0 \ \text{(i.e. } i \text{ is on the right side of } P)\\
w_i &= 1 \quad \text{if} \quad \lambda < 0 \quad \text{and} \quad \Delta x_i > 0\\
w_i &= 0 \quad \text{if} \quad \lambda < 0 \quad \text{and} \quad \Delta x_i < 0
\end{aligned} \tag{87}$$

This procedure ensures that only the points upwind of P contribute to the derivative approximation. This procedure is equivalent to taking the least squares approximation for the derivative without the weights (83), but with only the points left to the point P if $\lambda > 0$ and only the points right to the point P if $\lambda < 0$.

4.2.2. Second-order accurate least squares upwinding. To achieve second-order accuracy on an arbitrary distribution of points, the two-step procedure introduced by Ghosh and Deshpande [16] is used. This procedure is explained below. Consider the Taylor series expansion as

$$f_i = f_P + \left(\frac{\partial f}{\partial x}\right)_P \Delta x_i + \left(\frac{\partial^2 f}{\partial x^2}\right)_P \frac{(\Delta x_i)^2}{2} + O(\Delta x_i^3) \tag{88}$$

or

$$\Delta f_i = \left(\frac{\partial f}{\partial x}\right)_P \Delta x_i + \left(\frac{\partial^2 f}{\partial x^2}\right)_P \frac{(\Delta x_i)^2}{2} + O(\Delta x_i^3) \tag{89}$$

Differentiating the above equation with respect to x, the following equation is obtained:

$$\frac{\partial}{\partial x}(\Delta f_i) = \Delta x_i \left(\frac{\partial^2 f}{\partial x^2}\right)_P + O(\Delta x_i^2) \tag{90}$$

The above expression can be used to eliminate the $(\partial^2 f/\partial x^2)_P$ term in the Taylor expansion, and the least squares minimization then gives a second-order accurate expression for the derivative on an arbitrary distribution of points. For this purpose, let us define

$$\Delta \tilde{f}_i = \Delta f_i - \frac{\Delta x_i}{2} \frac{\partial}{\partial x}(\Delta f_i) \tag{91}$$

Substituting (89) and (90) in (91), we get

$$\begin{aligned} \Delta \tilde{f}_i = \Delta x_i \left(\frac{\partial f}{\partial x}\right)_P + \frac{\Delta x_i^2}{2}\left(\frac{\partial^2 f}{\partial x^2}\right)_P + O(\Delta x_i^3) \\ - \frac{\Delta x_i}{2}\left[\Delta x_i \left(\frac{\partial^2 f}{\partial x^2}\right)_P + O(\Delta x_i^2)\right] \end{aligned} \tag{92}$$

The second-order terms get cancelled and we get

$$\Delta \tilde{f}_i = \Delta x_i \left(\frac{\partial f}{\partial x}\right)_P + O(\Delta x_i^3) \tag{93}$$

Note that the truncation part in Equation (93) is of $O(\Delta x_i^3)$. Now, redefining the error as

$$e = \Delta \tilde{f}_i - \Delta x_i \left(\frac{\partial f}{\partial x}\right)_P \tag{94}$$

the sum of the squares of errors as

$$E = \sum_{i=1}^{n} (\Delta \tilde{f}_i - \Delta x_i f_x)^2 \tag{95}$$

and minimizing E w.r.t. f_x, the approximation for the derivative can be obtained as

$$\left(\frac{\partial f}{\partial x}\right)_P = \frac{\sum_{i=1}^{n} \Delta x_i \Delta \tilde{f}_i}{\sum_{i=1}^{n} \Delta x_i^2} \tag{96}$$

where

$$\Delta \tilde{f}_i = \Delta f_i - \frac{\Delta x_i}{2}\left[\left(\frac{\partial f}{\partial x}\right)_i - \left(\frac{\partial f}{\partial x}\right)_P\right] \tag{97}$$

Since the error e is defined with a truncation term $O[\Delta x_i^3]$, the above expression is second-order accurate. The derivatives $(f_x)_P$ and $(f_x)_i$ can be evaluated using the expression for the first-order accurate derivative (83). The above two-step procedure yields a second-order accurate derivative on an arbitrary distribution of points, *without increasing the points in the stencil*. Upwinding can be done in the same way as explained in the previous section. To suppress the spurious wiggles in the results obtained with the second-order accurate scheme, min–max limiters (see References [11, 16] for details) are used.

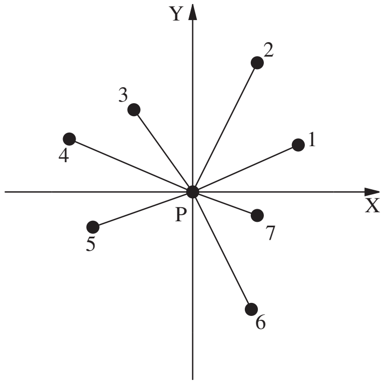

Figure 2. Arbitrary distribution of points in 2D.

4.3. Least squares upwinding in 2D

4.3.1. First-order accurate least squares upwinding. Consider an arbitrary distribution of points as shown in Figure 2. Taylor series expansion gives, for a set of points $i = 1, 2, \ldots, n$ around P,

$$f_i = f_P + (x_i - x_P)\left(\frac{\partial f}{\partial x}\right)_P + (y_i - y_P)\left(\frac{\partial f}{\partial y}\right)_P + O[(x_i - x_P)^2, (y_i - y_P)^2] \tag{98}$$

The error for this 2D case is defined by

$$e = \Delta f_i - \Delta x_i f_x - \Delta y_i f_y \tag{99}$$

The sum of squares of the error is given by

$$E = \sum_{i=1}^{n} (\Delta f_i - \Delta x_i f_x - \Delta y_i f_y)^2 \tag{100}$$

Minimizing E with respect to both f_x and f_y, the expressions for derivatives can be obtained as

$$f_x = \frac{\sum \Delta y_i^2 \sum \Delta x_i \Delta f_i - \sum \Delta x_i \Delta y_i \sum \Delta y_i \Delta f_i}{\sum \Delta x_i^2 \sum \Delta y_i^2 - (\sum \Delta x_i \Delta y_i)^2} \tag{101}$$

$$f_y = \frac{\sum \Delta x_i^2 \sum \Delta y_i \Delta f_i - \sum \Delta x_i \Delta y_i \sum \Delta x_i \Delta f_i}{\sum \Delta x_i^2 \sum \Delta y_i^2 - (\sum \Delta x_i \Delta y_i)^2} \tag{102}$$

In the above two expressions $\sum$ stands for $\sum_{i=1}^{n}$. The derivations of (101) and (102) are given in Reference [11]. The above two expressions ((101) and (102)) give approximations for space derivatives for an arbitrary distributions of n points, with respect to x and y, respectively. But, these approximations do not give preference to any direction, and they are not upwind approximations. It can be shown that these expressions give central difference like approximations for the derivatives, hence they will lead to an unstable scheme. To introduce upwinding let us first consider linear convection equation in 2D as

$$\frac{\partial f}{\partial t} + \lambda_1 \frac{\partial f}{\partial x} + \lambda_2 \frac{\partial f}{\partial y} = 0 \tag{103}$$

Here, f is a function that propagates with a constant wave speed whose components in x and y directions are λ_1 and λ_2, respectively. To develop an upwind scheme for (103), let us re-write (103) with C–I–R splitting as

$$\frac{\partial f}{\partial t} + \frac{\lambda_1 + |\lambda_1|}{2}\frac{\partial f}{\partial x} + \frac{\lambda_1 - |\lambda_1|}{2}\frac{\partial f}{\partial x} + \frac{\lambda_2 + |\lambda_2|}{2}\frac{\partial f}{\partial y} + \frac{\lambda_2 - |\lambda_2|}{2}\frac{\partial f}{\partial y} = 0 \tag{104}$$

Following the same procedure introduced in Section 4.2.1, weighted least squares approximations for the derivatives can be obtained as

$$f_x = \frac{\sum w_i \Delta y_i^2 \sum w_i \Delta x_i \Delta f_i - \sum w_i \Delta x_i \Delta y_i \sum w_i \Delta y_i \Delta f_i}{\sum w_i \Delta x_i^2 \sum w_i \Delta y_i^2 - (\sum w_i \Delta x_i \Delta y_i)^2} \tag{105}$$

$$f_y = \frac{\sum w_i \Delta x_i^2 \sum w_i \Delta y_i \Delta f_i - \sum w_i \Delta x_i \Delta y_i \sum w_i \Delta x_i \Delta f_i}{\sum w_i \Delta x_i^2 \sum w_i \Delta y_i^2 - (\sum w_i \Delta x_i \Delta y_i)^2} \tag{106}$$

Here, $\sum$ stands for $\sum_{i=1}^{n}$. The weights w_i are chosen in such a way that the above expressions for the derivatives give upwind approximations. In essence, only upwind points are considered while evaluating the derivatives. To put it in another way w_i is taken as 1 if i is an upwind point and w_i is taken as 0 if i is not an upwind point. This criterion for (104) leads to the following set of weights:

$$w_i = \begin{cases} 1 & \text{for } \Delta x_i < 0 \quad \text{and} \quad \lambda_1 > 0 \\ 1 & \text{for } \Delta x_i > 0 \quad \text{and} \quad \lambda_1 < 0 \\ 1 & \text{for } \Delta y_i < 0 \quad \text{and} \quad \lambda_2 > 0 \\ 1 & \text{for } \Delta y_i > 0 \quad \text{and} \quad \lambda_2 < 0 \\ 0 & \text{otherwise} \end{cases} \tag{107}$$

Another way of explaining the least squares upwinding is by splitting stencil based on the location of each neighbour. Here, two types of splitting the stencil are explained, the first one is x–y splitting and the other is quadrant splitting [16]. Now, let us consider x–y splitting. In this method the full set of neighbours for the point P can be divided into 4 subsets as points to the left side of y-axis, points to the right side of y-axis, point on the upper side of x-axis and points on the lower side of the x-axis.

Let $N(P)$ be the set of all neighbouring nodes to point P in the stencil, defined as

$$N(P) = \{i, \ 1 \leqslant i \leqslant n\} \tag{108}$$

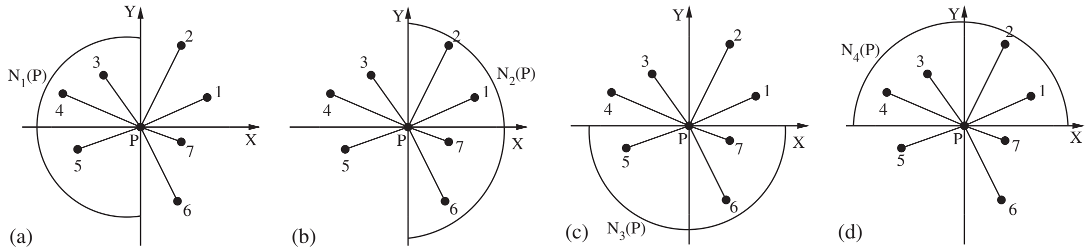

Figure 3. Split stencils for x–y splitting.

Now, dividing the set $N(P)$ into 4 subsets, based on the location as explained above, we get

$$\begin{aligned} N_1(P) &= \{i,\ \ i \in N(P) \ \text{ and } \ \Delta x_i < 0\} \\ N_2(P) &= \{i,\ \ i \in N(P) \ \text{ and } \ \Delta x_i > 0\} \\ N_3(P) &= \{i,\ \ i \in N(P) \ \text{ and } \ \Delta y_i < 0\} \\ N_4(P) &= \{i,\ \ i \in N(P) \ \text{ and } \ \Delta y_i > 0\} \end{aligned} \tag{109}$$

The above four sub-stencils are shown pictorially in Figure 3. Using (105), (106) and (109), (104) can be written as

$$\begin{aligned} &\frac{\partial f}{\partial t} + \frac{\lambda_1 + |\lambda_1|}{2}\left(\frac{\sum \Delta y_i^2 \sum \Delta x_i \Delta f_i - \sum \Delta x_i \Delta y_i \sum \Delta y_i \Delta f_i}{\sum \Delta x_i^2 \sum \Delta y_i^2 - (\sum \Delta x_i \Delta y_i)^2}\right)_{N_1(P)} \\ &\quad + \frac{\lambda_1 - |\lambda_1|}{2}\left(\frac{\sum \Delta y_i^2 \sum \Delta x_i \Delta f_i - \sum \Delta x_i \Delta y_i \sum \Delta y_i \Delta f_i}{\sum \Delta x_i^2 \sum \Delta y_i^2 - (\sum \Delta x_i \Delta y_i)^2}\right)_{N_2(P)} \\ &\quad + \frac{\lambda_2 + |\lambda_2|}{2}\left(\frac{\sum \Delta x_i^2 \sum \Delta y_i \Delta f_i - \sum \Delta x_i \Delta y_i \sum \Delta x_i \Delta f_i}{\sum \Delta x_i^2 \sum \Delta y_i^2 - (\sum \Delta x_i \Delta y_i)^2}\right)_{N_3(P)} \\ &\quad + \frac{\lambda_2 - |\lambda_2|}{2}\left(\frac{\sum \Delta x_i^2 \sum \Delta y_i \Delta f_i - \sum \Delta x_i \Delta y_i \sum \Delta x_i \Delta f_i}{\sum \Delta x_i^2 \sum \Delta y_i^2 - (\sum \Delta x_i \Delta y_i)^2}\right)_{N_4(P)} = 0 \end{aligned} \tag{110}$$

The subscripts in (110) indicate that the derivative is calculated using those particular stencils. Using the weights as shown in (107) is equivalent to using the x–y split stencils as in (110). Another way of splitting called *quadrant splitting* is also possible, in which upwinding is done by considering the informations coming from each quadrant separately. For details the reader is referred to References [11, 16]. Second-order accuracy in 2D is achieved in the same way as is described for the 1D case, by a two-step procedure on a compact stencil. Min–max limiters (see References [11, 16]) are used to suppress the spurious oscillations in the second-order accurate solution.

5. GRID-FREE RELAXATION SCHEME

In Section 2, the discrete kinetic system as a relaxation system is introduced for hyperbolic equations. In Section 4 the least squares upwinding is presented. These two basic ideas are used as building blocks for the new grid-free relaxation scheme. The discrete kinetic system approximates scalar or vector conservation laws by a set of linear advection equations with relaxation terms as source terms. The discrete kinetic system for a given conservation law is solved by splitting method. For a first-order accurate grid-free relaxation scheme simple operator splitting is used along with first-order accurate least squares upwinding. To achieve second-order accuracy Jin's modified Strang splitting method, which is discussed in Section 2, is used along with second-order accurate least squares upwinding. In this section first-order accurate and second-order accurate grid-free relaxation schemes are presented based on discrete kinetic approximation, in one and two space dimensions. A new solid-wall boundary condition is also presented in this section.

5.1. First-order accurate grid-free relaxation scheme in 1D

Consider discrete kinetic system in 1D:

$$\frac{\partial \mathbf{f}}{\partial t} + \Lambda \frac{\partial \mathbf{f}}{\partial x} = \frac{1}{\varepsilon}[\mathbf{F} - \mathbf{f}] \tag{111}$$

The above system, in its form, is same for a scalar conservation law or for vector conservation laws, differing only in the definitions of **f** and **F**. The details are presented in Section 2. Here, the numerical scheme is given in a general form such that it is applicable for a scalar conservation law or for vector conservation laws, with suitable definitions for **f** and **F**. As mentioned in Section 3.3, using simple operator splitting, (111) can be written as

$$\textit{Step}\ \text{A}: \quad \frac{\partial \mathbf{f}}{\partial t} = S^{(t)}\,(\text{relaxation step}) \tag{112}$$

$$\textit{Step}\ \text{B}: \quad \frac{\partial \mathbf{f}}{\partial t} = C^{(t)}\,(\text{convection step}) \tag{113}$$

After discretization, the relaxation step can be written as

$$\frac{\mathbf{f}^* - \mathbf{f}^n}{\Delta t} = \frac{1}{\varepsilon}[\mathbf{F}^* - \mathbf{f}^*] \tag{114}$$

Since the local Maxwellian F is a function of the conserved variables, at first glance, it seems inevitable to use a non-linear algebraic solver. However, using the moment relations $\mathbf{u} = P\mathbf{f} = P\mathbf{F}$, the above equation yields $\mathbf{u}^* = \mathbf{u}^n$. Thus, during the relaxation step, the conserved quantities are unchanged, and therefore, we have, $\mathbf{F}^* = \mathbf{F}^n$ (see Reference [23]). Hence, the above equation can be solved in an explicit way, without non-linear algebraic solvers. Now, (114) can be re-written as

$$\frac{\mathbf{f}^* - \mathbf{f}^n}{\Delta t} = \frac{1}{\varepsilon}[\mathbf{F}^n - \mathbf{f}^*] \tag{115}$$

After some re-arrangement, the update formula is given by

$$\mathbf{f}^* = \frac{1}{\left(1+\dfrac{\Delta t}{\varepsilon}\right)}\left[\mathbf{f}^n + \left(\frac{\Delta t}{\varepsilon}\right)\mathbf{F}^n\right] \tag{116}$$

After discretization the convection step can be given by

$$\frac{\mathbf{f}^{n+1}-\mathbf{f}^*}{\Delta t} + \Lambda\frac{\partial \mathbf{f}^*}{\partial x} = 0 \tag{117}$$

or

$$\mathbf{f}^{n+1} = \mathbf{f}^* + \Delta t\Lambda\frac{\partial \mathbf{f}^*}{\partial x} \tag{118}$$

where $\Lambda(\partial \mathbf{f}^*/\partial x)$ is calculated by using least squares upwinding, as described in Section 4.2.1. Hence, Equations (116) and (118) together give the first-order accurate update formula for the grid-free relaxation scheme in 1D.

5.2. Second-order accurate grid-free relaxation scheme in 1D

As mentioned in Section 3.5, even the Strang's splitting reduces to formal first-order accuracy in the stiff limit, so to achieve formal second-order accuracy the Jin's modified TVD Runge–Kutta method is used. For the discrete kinetic system (111), the second-order grid-free relaxation scheme is given by (69) to (73) where Δ_{conv} is the discrete convection operator. In the present case second-order accurate two-step least squares upwind approximation is used for this convection operator (presented in Section 4.2.2). Using the fact that the conserved quantities are unchanged during the collision step, $\mathbf{F}^*$ is calculated as

$$\mathbf{F}^* = \mathbf{F}^n \tag{119}$$

and $\mathbf{F}^{**}$ is calculated after the second step, i.e. after calculating $\mathbf{f}^{(1)}$. In 2D, a similar procedure is followed, using the 2D discrete Boltzmann equation, the simple splitting or the Strang's splitting method along with least squares upwinding presented in Section 4.3.

5.3. Stability analysis for grid-free relaxation scheme

In the grid-free relaxation scheme the discrete kinetic system or the diagonal form of relaxation system is solved by splitting method. In the splitting method, the relaxation step is solved by an implicit method, whereas the convection step (consisting of just a linear advection equation) is solved by an explicit method. So, it is sufficient to choose the convection step, i.e. a linear advection equation, for stability analysis, since the time step is restricted by the convection step alone. The following stability analysis is based on Reference [17]. Consider a linear advection equation in 1D, given by

$$\frac{\partial f}{\partial t} + \lambda\frac{\partial f}{\partial x} = 0 \tag{120}$$

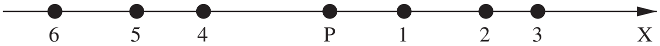

Figure 4. Arbitrary distribution of points in 1D, $i=1,2,3,\ldots,N$.

where λ is a constant wave speed. Consider the arbitrary distribution of points shown in Figure 4. Now, let us replace the space derivative in (120) by the least squares derivative to obtain

$$\left.\frac{\mathrm{d}f}{\mathrm{d}t}\right|_P = -\lambda\frac{\sum_{i=1}^N w_i\Delta x_i\Delta f_i}{\sum_{i=1}^N w_i\Delta x_i^2} \tag{121}$$

where $\Delta x_i = x_i - x_P$ and $\Delta f_i = f_i - f_P$. The weights w_i are chosen in such a way that the upwinding property is satisfied (as explained in Section 4). Let us define

$$a_i = -\frac{\lambda w_i\Delta x_i}{\sum_{i=1}^N \Delta x_i^2} \tag{122}$$

$$a_i > 0 \text{ because } w_i\Delta x_i < 0 \text{ for } \lambda > 0 \text{ and } w_i\Delta x_i > 0 \text{ for } \lambda < 0 \text{ from (87)} \tag{123}$$

Now, Equation (121) can be re-written as

$$\left.\frac{\mathrm{d}f}{\mathrm{d}t}\right|_P = \sum_{i=1}^N a_i(f_i - f_P) \tag{124}$$

Since the coefficients a_i are positive, the local maximum cannot increase and the local minimum cannot decrease, i.e. the scheme is local extremum diminishing (LED) [24]. Using the explicit Euler time stepping (124) can be written as

$$f_P^{n+1} = f_P^n + \Delta t\sum_{i=1}^N a_i(f_i^n - f_P^n) \tag{125}$$

or

$$f_P^{n+1} = \left(1 - \Delta t\sum_{i=1}^N a_i\right) f_P^n + \Delta t\sum_{i=1}^N a_i f_i^n \tag{126}$$

or

$$f_P^{n+1} = \sum_{i=P,\, i=1}^N \beta_i f_i^n \tag{127}$$

where

$$\begin{aligned}\beta_i &= 1 - \Delta t\sum_{i=1}^N a_i \quad \text{for } i = P\\ \beta_i &= \Delta t\sum_{i=1}^N a_i \quad \text{for } i \neq P\end{aligned} \tag{128}$$

The LED condition demands that the following criterion is satisfied:

$$\min_i f_i^n \leqslant f_i^{n+1} \leqslant \max_i f_i^n \text{ and, therefore, } \|f^{n+1}\|_{L_\infty} \leqslant \|f^n\|_{L_\infty} \tag{129}$$

We, therefore, obtain

$$\beta_i \geqslant 0 \quad \text{and} \quad \sum_{i=P,\, i=1}^{N} \beta_i = 1 \tag{130}$$

The second condition on β is the consistency condition, for a constant to be a solution of (127). To satisfy (130) the following condition on Δt can be obtained:

$$\begin{gathered} \text{because } a_i \geqslant 0 \quad 0 \leqslant \beta_i \leqslant 1 \quad \text{for } i = P \\ \Delta t \leqslant \frac{1}{\sum_{i=1}^{N} a_i} \quad \text{for } i \neq P \end{gathered} \tag{131}$$

The above CFL-like condition gives the time step restriction for the scheme to be stable, for an arbitrary distribution of points.

5.4. A note on flux conservation for the grid-free relaxation scheme

It is well-known that flux conservation is important to capture the location of discontinuities like shocks and contacts accurately, in the framework of finite volume methods. Shocks and contact discontinuities are common features in compressible flows and hence the issue of flux conservation becomes important. The grid-free (or meshless) methods do not belong to the framework of finite volume methods and the issue of flux conservation is not studied in detail yet for these newly developed schemes. One major difficulty in studying the issue of flux conservation for grid-free schemes is that the neighbourhood connectivity of a point consists of overlapping sets of quadrilaterals, triangles (in 2D), hexahedrals or tetrahedrals (in 3D), which is quite unlike the situation in the case of finite volume methods. Deshpande *et al.* [25] have made preliminary studies on the issue of flux conservation for the least-squares-based kinetic schemes. A similar analysis is applicable for the present scheme. The numerical results shown in the next section for scalar and vector conservation equations, compared with exact solutions and established numerical solutions, provide strong evidence that the flux conservation is satisfied.

5.5. A new solid-wall boundary condition

The wall boundary condition used in the present work is based on the simple physical principle that the component of velocity normal to the wall should be zero. The basic idea of the new solid-wall boundary condition is explained in this section with the aid of Figure 5.

Let P be the point at which the wall boundary condition is to be applied. Let α be the angle made by the tangent with x-axis, in the counter-clockwise direction. For the boundary condition to be imposed at the point P, the state variables (p, ρ, u and v) are required. These are extrapolated from inside the domain, using the Taylor series expansion and the least squares expressions for evaluating the derivatives in it. The least squares expressions are used without any upwind weights. Let u_1 and v_1 be the components of the velocity along x and y co-ordinates, respectively, which are obtained by first-order accurate least squares

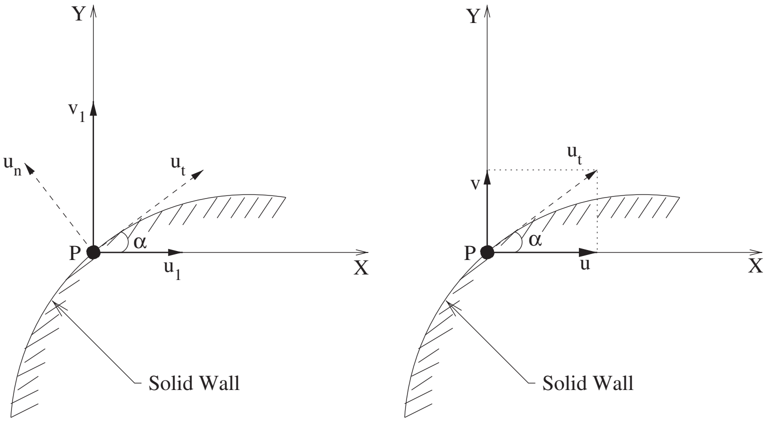

Figure 5. Solid-wall boundary condition.

extrapolation as explained above. From u_1 and v_1 the tangential and normal components of the velocity u_t and u_n are calculated as (refer to Figure 5(a))

$$\begin{aligned} u_t &= u\cos\alpha + v\sin\alpha \\ u_n &= -u\sin\alpha + v\cos\alpha \end{aligned} \tag{132}$$

Now, the wall boundary condition is enforced by killing the normal component of the velocity i.e. forcing u_n to zero. Then, the velocities are transformed back into the x and y coordinates as (refer to Figure 5(b)). The inverse transformation is given by

$$\begin{aligned} u &= u_t\cos\alpha - u_n\sin\alpha \\ v &= u_t\sin\alpha + u_n\cos\alpha \end{aligned} \tag{133}$$

Enforcing the normal velocity u_n to be zero, we obtain

$$\begin{aligned} u &= u_t\cos\alpha \\ v &= u_t\sin\alpha \end{aligned} \tag{134}$$

6. RESULTS AND DISCUSSION

The new *grid-free relaxation scheme* is used to solve some standard test problems for inviscid Burgers' equation and Euler equations in 1D and 2D. The test problems in 1D are solved both on uniform distribution of points and arbitrary distribution of points, where the arbitrary distribution of points are obtained by a random number generator. In case of 2D the test problems are solved on uniform distribution of points and also on points obtained from an unstructured mesh generator called *Delaundo* [26]. The test cases for the Burgers' equation are taken from References [4, 27]. For the test cases in 2D for Euler equations, the points in the flow domain are adapted based on the solution, to explore the power of the grid-free method. For adaptation bi-dimensional anisotropic mesh generation and adaptation software (BAMG) [28] is used. Results for the adapted cases are also presented in this section.

When ε goes to zero the relaxation system 3, reduces to the original hyperbolic equations. But in the numerical study ε is not actually zero and a finite value has to be assumed. Hence, the effect of ε on the solution was studied for values ranging from 10^{-4} to 10^{-45}. It is found that there is no change in the solution when ε is in the range of 10^{-8} to 10^{-45}. Hence, for all the test cases discussed in this section the numerical value of ε is taken as 10^{-8}. This is consistent with the experiments done by Jin and Xin [8] on the numerical value of ε in the framework of finite volume methods.

6.1. *Inviscid Burgers' equation in 1D*

Test case 1: In one dimension, the inviscid Burgers' equation is

$$\frac{\partial u}{\partial t}+\frac{\partial}{\partial x}\left(\frac{u^2}{2}\right)=0 \tag{135}$$

The initial conditions for this test case are given by

$$u=\begin{cases}1 & \text{for } |x|<\frac{1}{3}\\ -1 & \text{for } \frac{1}{3}<|x|\leqslant 1\end{cases}$$

Exact solution:

$$u_{\text{ex}}(x,t)=\begin{cases}-1 & \text{for} \quad -\infty \quad <x<b_1\\ -1+2\dfrac{x-b_1}{b_2-b_1} & \text{for} \quad b_1 \quad <x<b_2\\ 1 & \text{for} \quad b_1 \quad <x<b_{\text{shock}}\\ -1 & \text{for} \quad b_{\text{shock}} <x<\infty\end{cases}$$

where

$$b_1=-\tfrac{1}{3}-t, \quad b_2=-\tfrac{1}{3}+t, \quad b_{\text{shock}}=\tfrac{1}{3}$$

These conditions for Burgers' equation describe a square wave. Periodic boundary conditions are applied at the boundaries. The problem consists of a jump from zero to one at $x=-1/3$ which creates an expansion fan while the jump from one to zero at $x=1/3$ creates a shock. Exact solution and the numerical solutions are obtained at $u(x,0.3)$, with 200 points in the domain. The first- and second-order accurate numerical solutions are obtained both on uniform distribution of points and arbitrary distribution of points, and are shown in Figures 6 and 7. We can see that the scheme has no problem at the sonic point (where the value of u jumps from negative to positive) and no unphysical expansion shocks are present.

6.2. *Inviscid Burgers' equation in 2D*

Test case 2: The inviscid Burgers' equation considered here is given by

$$\frac{\partial u}{\partial t}+\frac{\partial}{\partial x}\left(\frac{u^2}{2}\right)+\frac{\partial u}{\partial y}=0$$

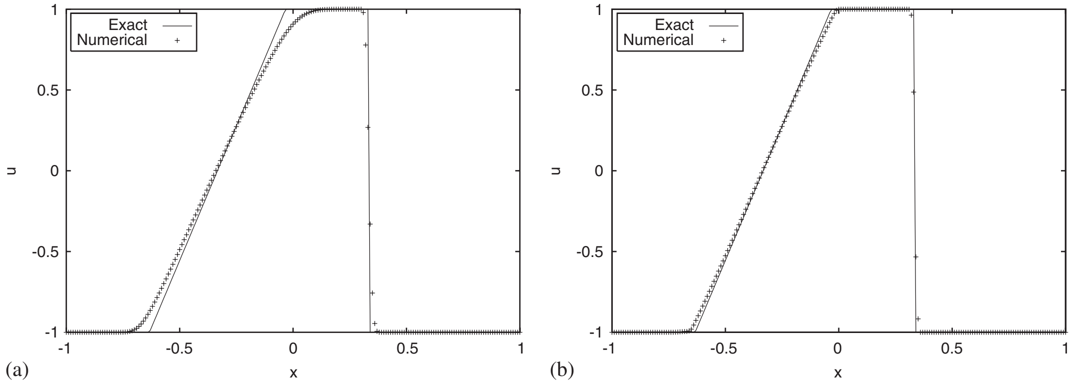

Figure 6. Solutions for inviscid Burgers' equation on uniform distribution of points, for test case 1: (a) first-order accurate solution; and (b) second-order accurate solution.

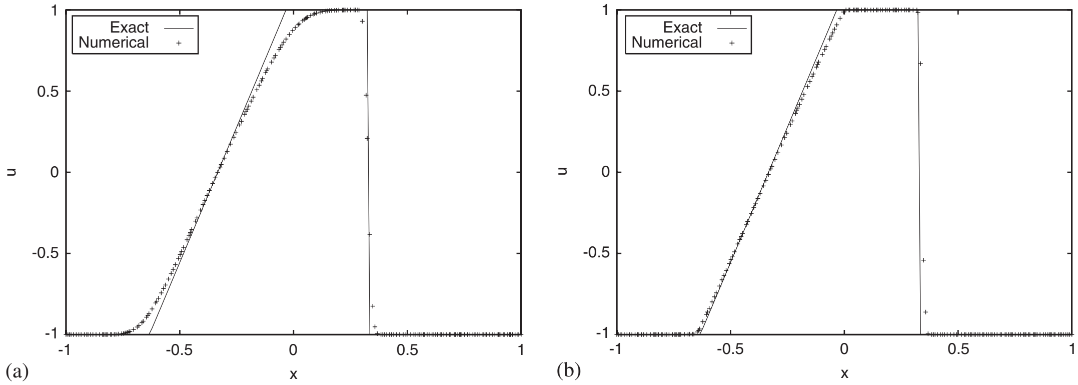

Figure 7. Solutions for inviscid Burgers' equation on arbitrary distribution of points, for test case 1: (a) first-order accurate solution; and (b) second-order accurate solution.

Boundary conditions for the above non linear problem over a square $[0,1]\times[0,1]$, are given by

$$u(0,y)=1.5 \quad \text{for } 0<y<1$$
$$u(1,y)=-0.5 \quad \text{for } 0<y<1$$
$$u(x,0)=1.5-2x \quad \text{for } 0<x<1$$

The exact solution is given as shown in Figure 8(b), in which region A and region B are separated by an oblique shock originating at $(x,y)=(0.75,0.5)$ and in region C there is a smooth variation from left to right.

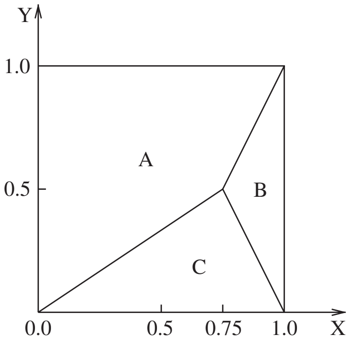

Figure 8. Exact solutions for the 2D inviscid Burgers' equation for test case 2.

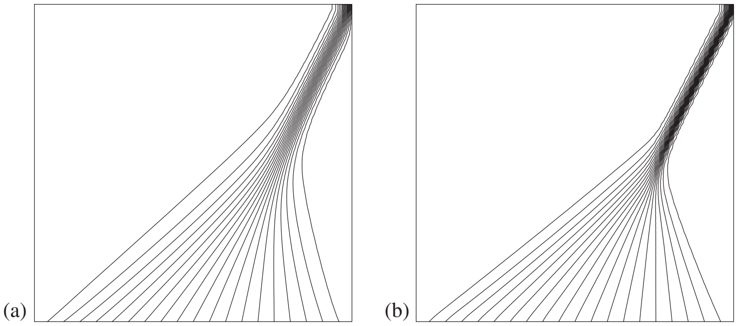

Figure 9. Solutions for 2D inviscid Burgers' equation on uniform distribution of points, for test case 2: (a) first-order accurate solution; and (b) second-order accurate solution.

The exact solution $u_{\text{ex}}(x,y)$ is given by

$$u_{\text{ex}}(x,y) = 1.5 \quad \text{if } (x,y) \text{ is in region A}$$

$$u_{\text{ex}}(x,y) = -0.5 \quad \text{if } (x,y) \text{ is in region B}$$

$$u_{\text{ex}}(x,y) = \frac{1.5-2x}{1-2y} \quad \text{if } (x,y) \text{ is in region C}$$

Numerical experiments are done using the first- and second-order accurate versions of the grid-free relaxation scheme both on uniform distribution of 64×64 points and on a distribution of 4825 points obtained from the triangular mesh generator. Figures 9(a) and (b) show the first- and second-order accurate solutions on the uniform distribution points, where as Figures 10(a) and (b) show the first- and second-order accurate solutions on the points obtained from the triangular mesh generator. For this test case and the previous one, on the top boundary, Neumann boundary condition is applied in the least squares framework, the details of which are available in Reference [12].

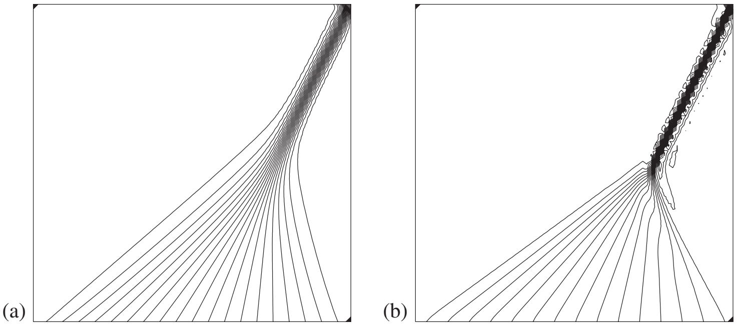

Figure 10. Solutions for 2D inviscid Burgers' equation on distribution of points obtained by the unstructured mesh generator, for test case 2: (a) first-order accurate solution; and (b) second-order accurate solution.

6.3. 1D Euler equations

Test case 3: 1D shock tube problem (Sod's test case).

The initial conditions are given by

$$\begin{aligned} \rho_{\mathrm{L}} &= 1.0 \quad \rho_{\mathrm{R}} = 0.125 \\ u_{\mathrm{L}} &= 0.0 \quad u_{\mathrm{R}} = 0.0 \\ p_{\mathrm{L}} &= 1.0 \quad p_{\mathrm{R}} = 0.1 \end{aligned} \tag{136}$$

where the subscripts L and R indicate the left and right sides of the diaphragm, respectively.

This problem is solved on a uniform distribution of 200 points and also on a random distribution of 200 points. The arbitrary distribution of points are obtained from a random number generator. Both first- and second-order accurate versions of the grid-free relaxation scheme are tested on both uniform and arbitrary distribution of points and the results are presented here. Figure 11 shows the first-order solution on uniform distribution of points and the solution is very dissipative. The second-order scheme improves the solution and is shown in Figure 12. Similarly, Figures 13 and 14 show the first- and second-order solutions, respectively, on an arbitrary distribution of points.

6.4. 2D Euler equations

Test case 4: Shock reflection problem.

A schematic of the *regular shock reflection* problem is shown in Figure 15. In Figure 15, AB is a solid wall, and an incident shock at an angle β hits the solid wall and gets reflected. The flow is fully supersonic in this case. Along the left boundary AD an inflow boundary condition is applied. Along the top boundary CD the post shock values are imposed. This way the shock is introduced at the corner D. Along the right boundary BC the supersonic outflow boundary condition is applied, i.e. the state variables are extrapolated from inside. Along the solid wall AB, wall boundary condition, which is described in the previous section

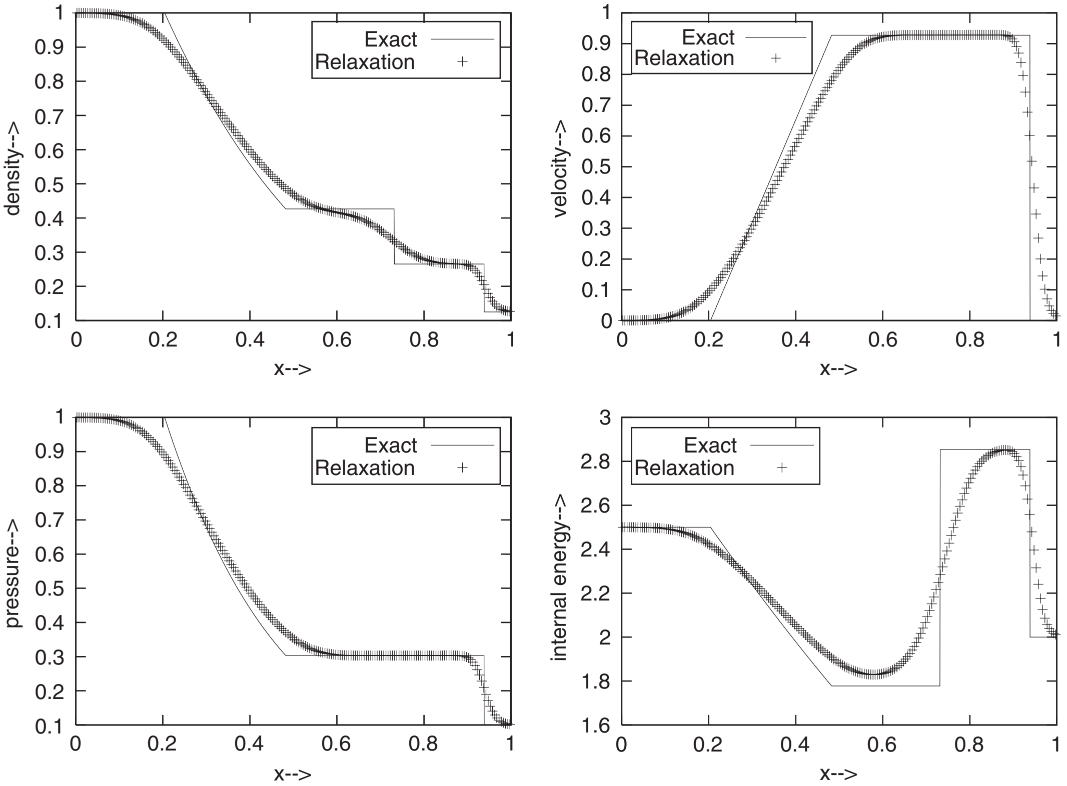

Figure 11. First-order accurate solution for shock tube problem on uniform distribution of points, for Sod's test case.

is applied. The numerical values considered for this test case are given below.

	Inflow	Post shock
ρ	1.0	1.69997
u	2.9	2.61934
v	0.0	−0.50633
p	1/1.4	1.52819

(137)

This problem is solved using first- and second-order accurate versions of the grid-free relaxation scheme on uniform distribution of points and also on a distribution of points obtained from the triangular mesh generator. A domain of 3×1 is chosen for this problem. Three different sizes $(30 \times 10, 60 \times 20$ and $120 \times 40)$ of uniform distribution of points are used for testing the scheme. Similarly, three different sizes $(391, 1452$ and $5667)$ of distribution of points obtained from the triangular mesh generator are also used for testing the scheme. But, only some of the results are presented here (see Reference [11] for full details). The contour plots of pressure on two different distributions of points are shown in Figures 16 and 17.

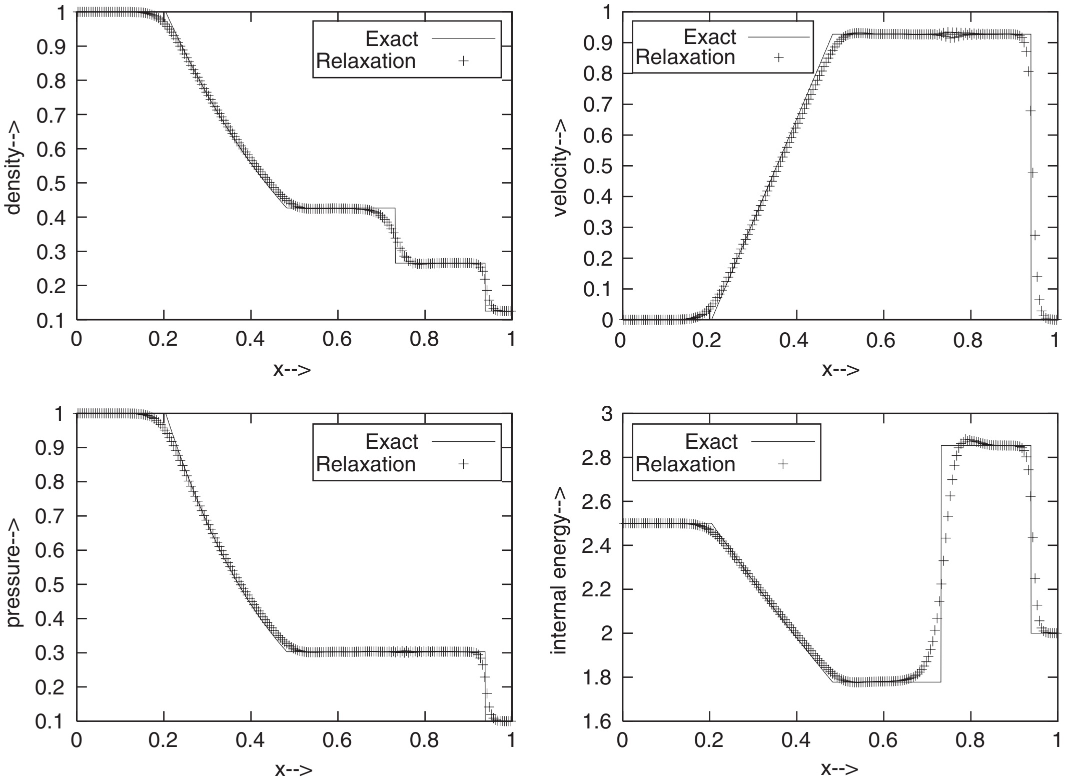

Figure 12. Second-order accurate solution for shock tube problem on uniform distribution of points, for Sod's test case.

Adaptation: To demonstrate the power of the grid-free scheme, adaptation is done for the shock reflection problem and also for the next test case, i.e. internal flow over a ramp in a channel. For adaptation the bi-dimensional anisotropic mesh generator and adaptation software (BAMG), developed by INRIA [28], which is freely available in the web is used. Figures 18 and 19 show the distribution of points and solutions, respectively, at different levels of adaptation, for the shock reflection problem. We can see a lot of improvement in the solution at each level of adaptation. The residue history for the adapted solution is shown in Figure 20.

Test case 5: Internal flow over a ramp in a channel

$$\text{Inflow Mach number} = 2.0$$

$$\text{Ramp angle} = 15^\circ$$

Internal flow over a ramp in a channel has a shock, two reflected shocks, an expansion fan and interaction of a shock wave and an expansion fan. The bottom and the top boundaries are considered as solid walls and the wall boundary condition is applied along these two boundaries. Along the left boundary inflow boundary condition is applied. Along the right boundary supersonic boundary condition is applied, i.e. the state variables are extrapolated

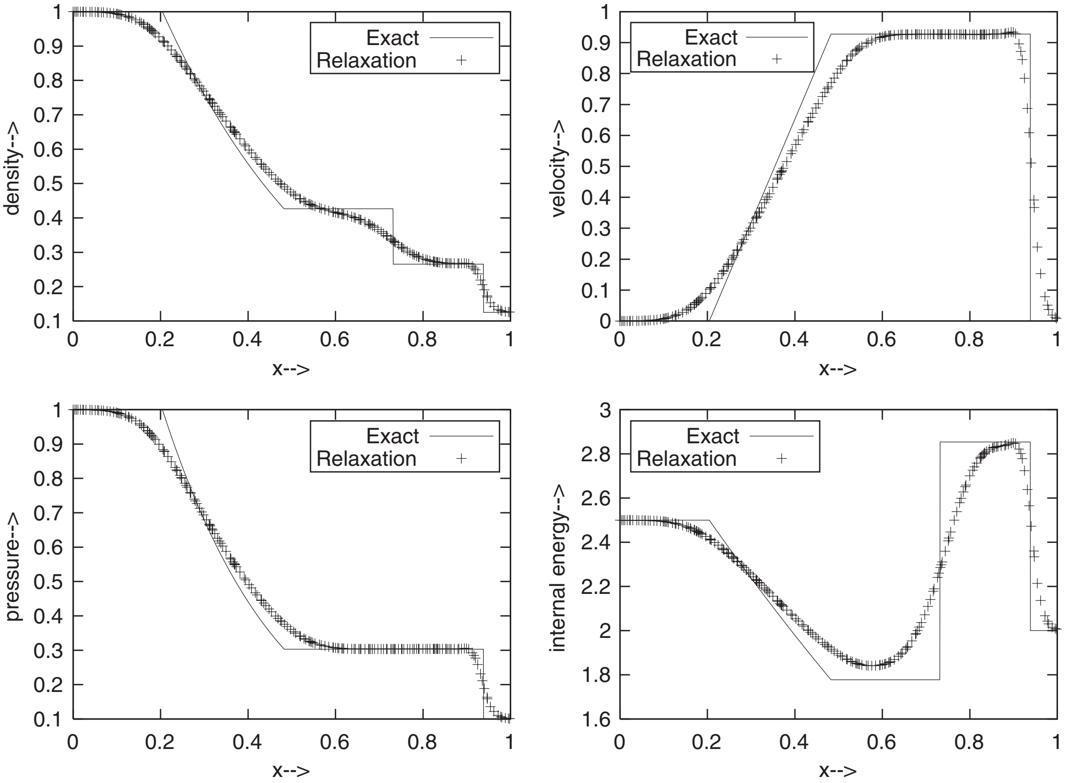

Figure 13. First-order accurate solution for shock tube problem on arbitrary distribution of points, for Sod's test case.

from inside. For this problem second-order scheme is used and the solution is obtained on four distributions of points with 3952, 7868, 15 329 and 31 138 points, respectively, all being obtained from the triangular mesh generator Delaundo [26]. Pressure contours for the second-order solution on the four distributions of points are given in Figure 21. Adaptation is done for this case also and the adapted distribution of points and solutions are given in Figures 22 and 23. In this case in each level of adaptation there is a significant improvement in the solution. Residue history for the adapted case is shown in Figure 24.

6.5. Cost and accuracy of the grid-free relaxation scheme

The grid-free relaxation scheme updates more number of variable than the usual schemes due to the introduction of an additional equation in the relaxation system and introduction of artificial variables. However, a relaxed scheme can be obtained by summing up the characteristic variables and discrete Maxwellians, by substituting $\varepsilon = 0$. This strategy has not been used in the current study but it has been observed and used in the later studies. For a simple 1D case the scheme has been compared with Roe's scheme and it is found that the present scheme takes double the amount of time that Roe's scheme takes. This is expected as the number of

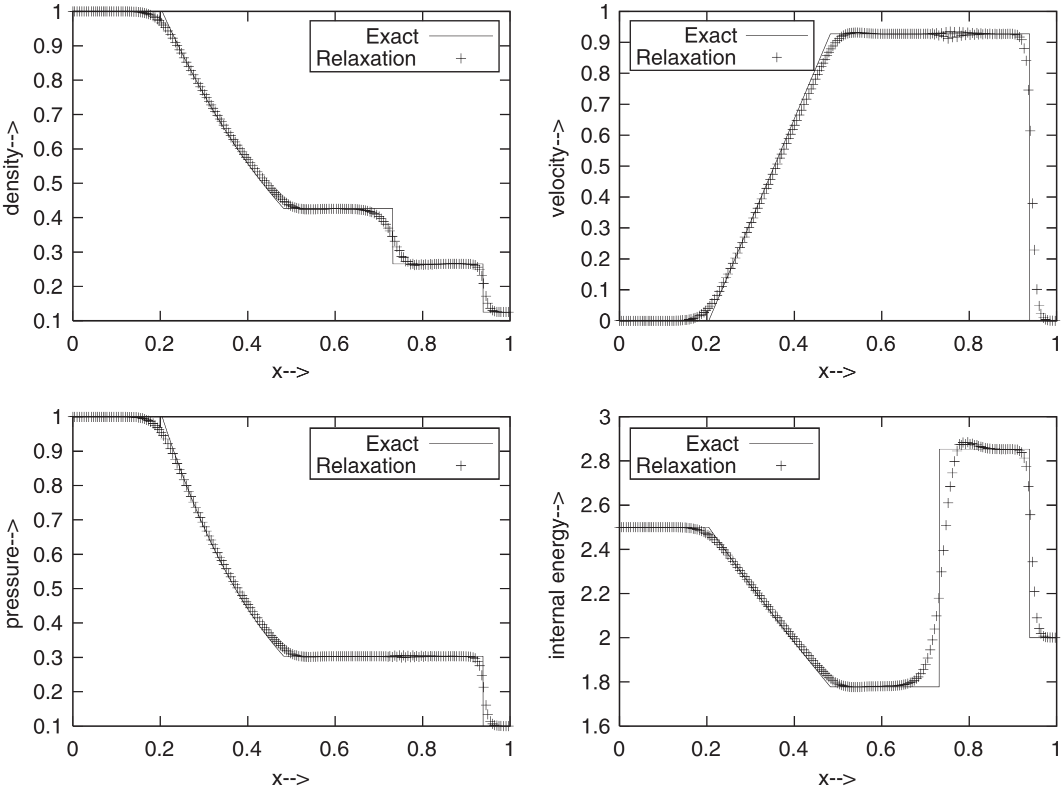

Figure 14. Second-order accurate solution for shock tube problem on arbitrary distribution of points, for Sod's test case.

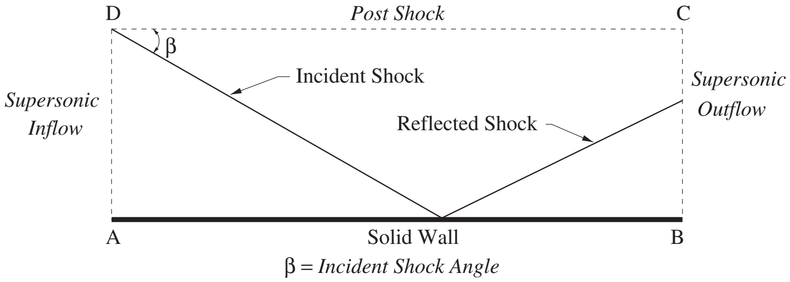

Figure 15. Schematic for the shock reflection problem.

variables updated in the current scheme is almost double compared to Roe's scheme. But for a relaxed scheme the number of variables updated will be equal to that of a Roe's scheme and hence the cost should be almost equal to that of a Roe's scheme. The relaxed scheme in the framework of grid-free relaxation scheme will be used in future studies. The real advantage

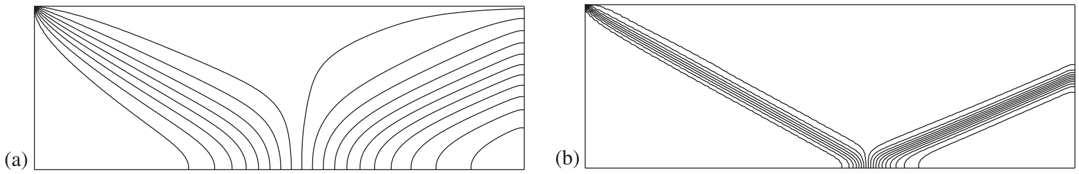

Figure 16. Solutions for the shock reflection problem on uniform distribution of 120 × 40 points, for test case 4: (a) first-order accurate solution; and (b) second-order accurate solution.

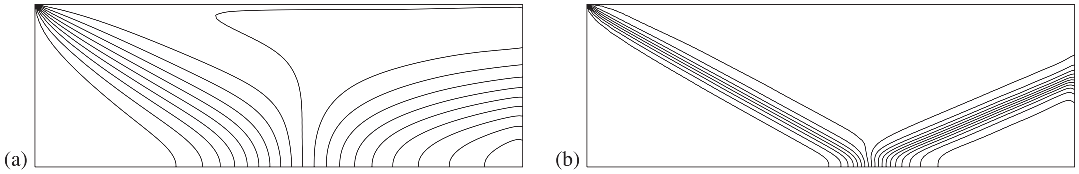

Figure 17. Solutions for the shock reflection problem on 5667 points obtained from a triangular mesh generator, for test case 4: (a) first-order accurate solution; and (b) second-order accurate solution.

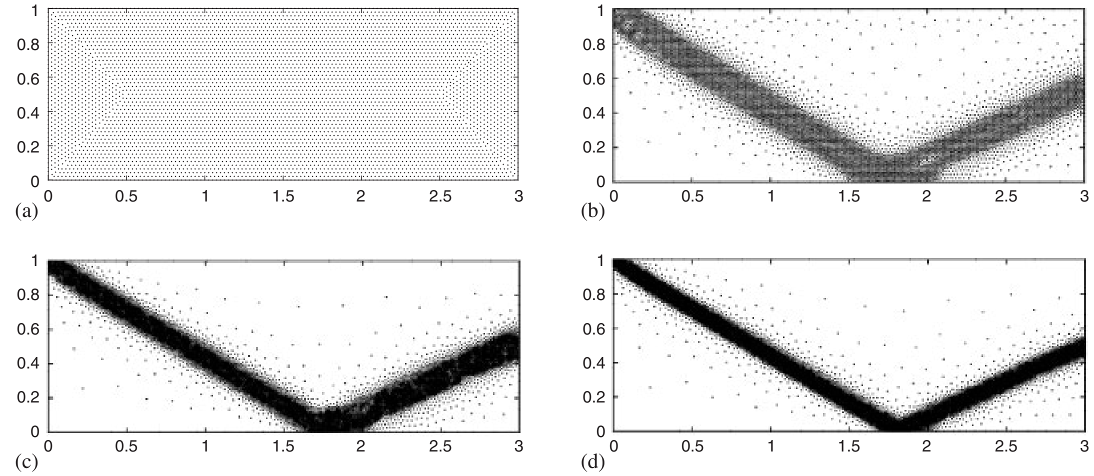

Figure 18. Distribution of points for the shock reflection problem at different levels of adaptation: (a) initial distribution of 5667 points; (b) 5740 points after first level; (c) 10 000 points after second level; and (d) 12 000 points after third level.

of the present scheme lies in the reduction of time and effort that would be otherwise spent in generating meshes for complicated geometries.

First-order accurate results in the above test cases show high dissipation. This is expected as the first-order accurate scheme is equivalent to Lax–Fredrichs scheme, which is known to have the highest dissipation. However, the focus of the current study is to demonstrate the capability of the newly developed grid-free scheme within the framework of simple

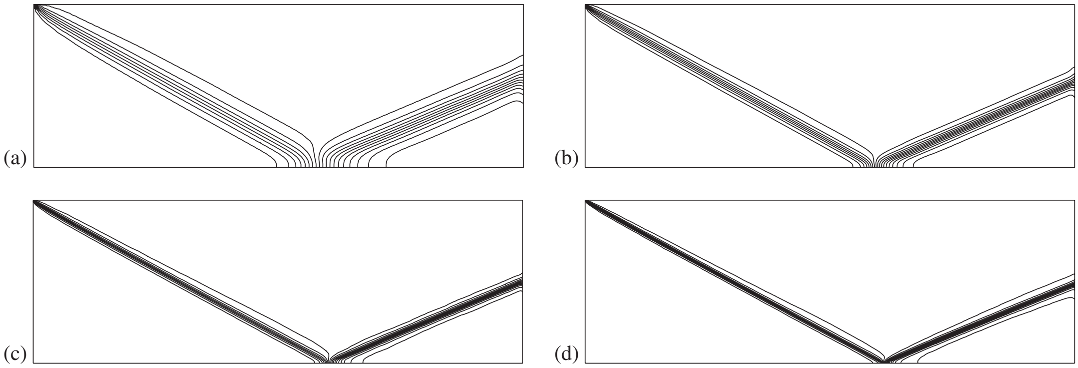

Figure 19. Solutions for the shock reflection problem, at different levels of adaptation: (a) initial solution; (b) after first level of adaptation; (c) after second level of adaptation; and (d) after third level of adaptation.

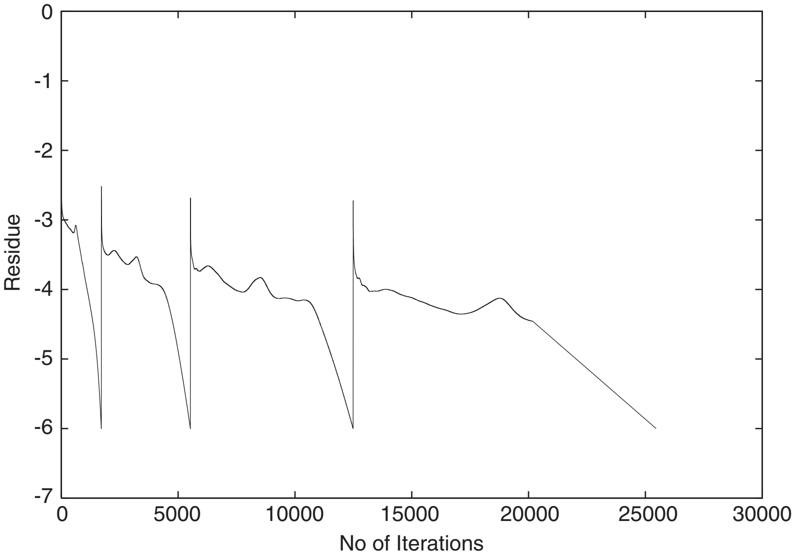

Figure 20. Residue history for the adapted solution for the shock reflection problem.

relaxation systems. A low-dissipation method in the framework of relaxation schemes is developed by Raghurama Rao and Balakrishna [29] in which the discontinuities are capture exactly by enforcing Rankine–Hugoniot condition at the discrete level. In another study by Anand Tripathy and Raghurama Rao [30, 31] a relaxation scheme is developed to solve Navier–Stokes equations. Modifying the grid-free relaxation scheme to reduce the dissipation and to solve Navier–Stokes equations, based on the concepts of the above-mentioned references, are the subjects of a future study.

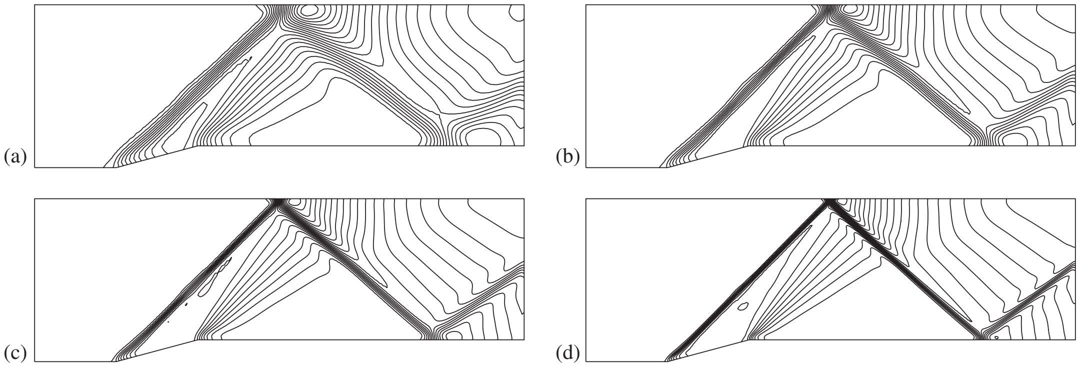

Figure 21. Pressure contours for the internal flow over a ramp in a channel: (a) solution on 3952 points; (b) solution on 7868 points; (c) solution on 15 329 points; and (d) solution on 31 138 points.

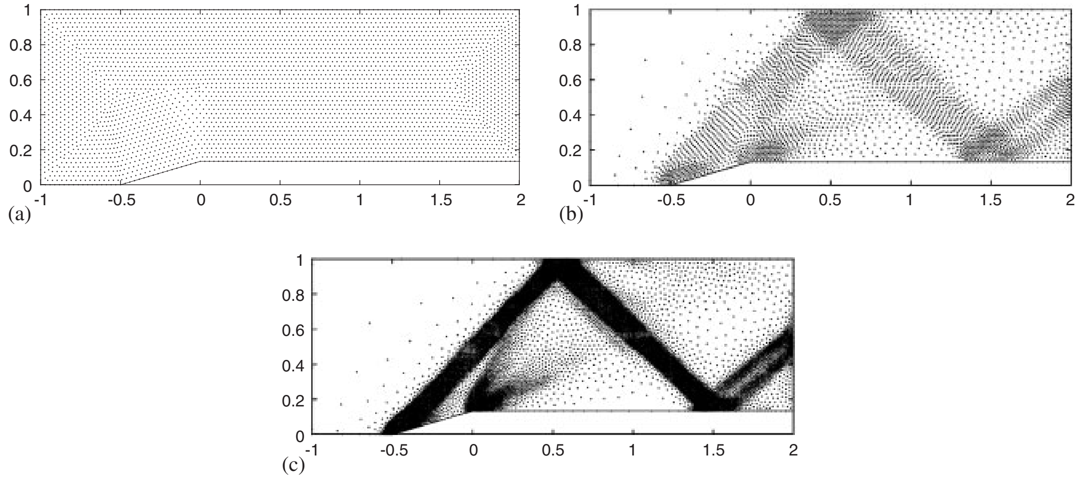

Figure 22. Distribution of points for the internal flow over a ramp, at different levels of adaptation: (a) initial distribution of 3952 points; (b) 3674 points after first level of adaptation; and (c) 10 841 points after second level of adaptation.

7. CONCLUSIONS

A *new grid-free upwind relaxation scheme* is developed to simulate inviscid compressible flows numerically. The scheme is grid free in the sense that it works on any arbitrary distribution of points. All that the solver needs is the coordinates of a set of neighbouring nodes around each node in the domain. Apart from being grid free, another basic advantage of the scheme is that it avoids complicated flux splitting schemes and time-consuming Riemann solvers, which often get into troubles. Efforts required to generate grids for complex geometries can be reduced, as the scheme is grid free.

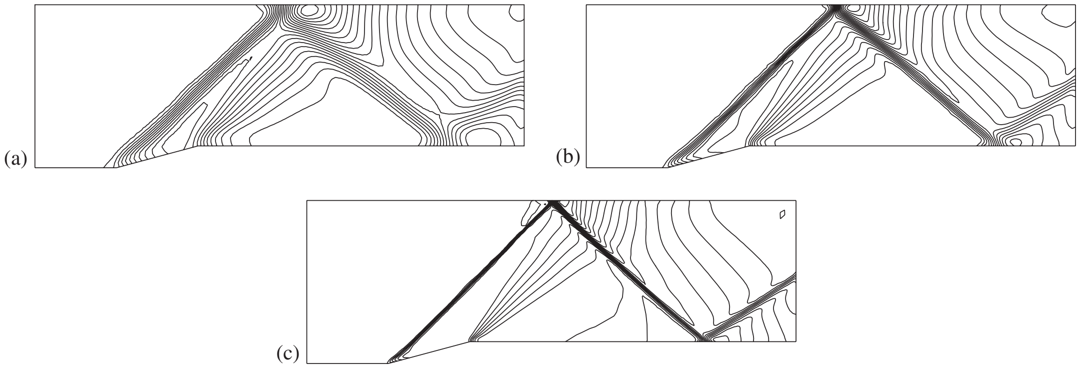

Figure 23. Solutions for the internal flow over a ramp, at different levels of adaptation: (a) initial solution; (b) after first level of adaptation; and (c) after second level of adaptation.

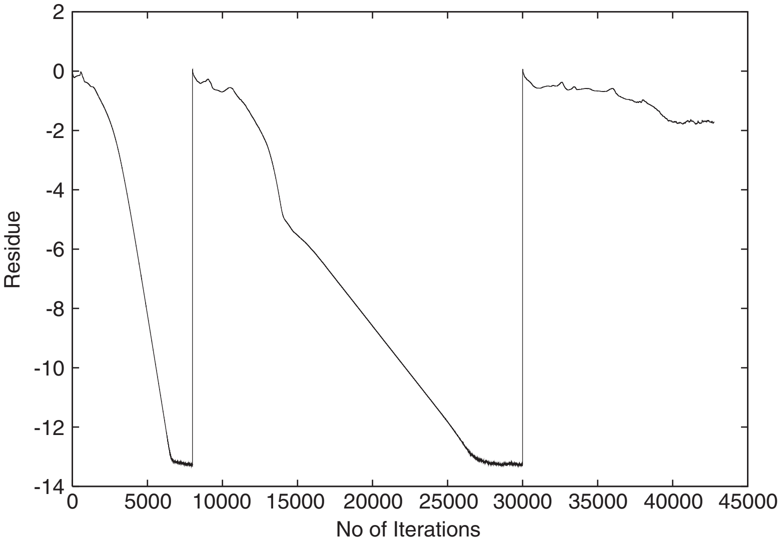

Figure 24. Residue history for the adaptation case for the internal flow over ramp.

The first- and second-order accurate versions of the scheme are used to solve scalar and vector conservation laws for different test problems in one and two space dimensions. To show the grid-free nature of the scheme in 1D, the scheme is tested on uniform distribution of points as well as arbitrary distribution of points, which are obtained from a random number generator. In 2D the scheme is tested on a distribution of points obtained from a structured mesh as well as from an unstructured mesh and the results demonstrate the usefulness of the scheme in capturing the flow features efficiently. To explore the power of the grid-free scheme for adaptation, solution-based adaptation is done for some test cases in 2D. The results demonstrate that the scheme is quite suitable for adaptation. A new solid-wall boundary condition is developed and its usefulness is demonstrated through application to the above test problems.

REFERENCES

1. Hirsch C. Numerical computation of internal and external flows. *Fundamentals of Numerical Discretization*, vol. 1. Wiley: New York, 1988.
2. Hirsch C. Numerical computation of internal and external flows. *Computational Methods for Inviscid and Viscous Flows*, vol. 2. Wiley: New York, 1990.
3. Toro EF. *Riemann Solvers and Numerical Methods for Fluid Dynamics*: *A Practical Introduction* (2nd edn). Springer: Berlin, 1999.
4. Laney CB. *Computational Gas Dynamics*. Cambridge University Press: Cambridge, MA, 1998.
5. Deshpande SM. Kinetic flux splitting schemes. In *Computational Fluid Dynamics Review 1995*: *A State-of-the-Art Reference to the Latest Developments in CFD*, Hafez MM, Oshima AK (eds). Wiley: Chichester, 1995.
6. Godlewski E, Raviart PA. *Numerical Approximations of Hyperbolic Systems of Conservation Laws*. Applied Mathematics Series, vol. 118. Springer: Berlin, 1996.
7. Quirk JJ. A contribution to the great Riemann solver debate. *International Journal for Numerical Methods in Fluids* 1994; **18**:555–574.
8. Jin S, Xin Z. The relaxation schemes for systems of conservation laws in arbitrary space dimensions. *Communications in Pure and Applied Mathematics* 1995; **48**:235–276.
9. Balasubramanyam S, Raghurama Rao SV. A new grid-free relaxation scheme for Euler equations. *Fluid Mechanics Report No*: *2001 FM 07*, Department of Aerospace Engineering, IISc, Bangalore, India.
10. Balasubramanyam S, Raghurama Rao SV. A new grid-free relaxation scheme for Euler equations. *4th Annual CFD Symposium*, Bangalore, 10–11 August 2001.
11. Balasubramanyam S. A new grid-free upwind relaxation scheme for compressible flows. *M.Sc.* (*Engg.*) *Thesis*, Department of Aerospace Engineering, Indian Institute of Science, Bangalore, India, 2002.
12. Balasubramanyam S, Raghurama Rao SV. A new grid-free relaxation scheme for Euler equations. *Ninth Asian Congress of Fluid Mechanics*, Isfahan, Iran, May 27–31, 2002.
13. Raghurama Rao SV. New numerical schemes based on relaxation systems for conservation laws. *AGTM Report 249*, Arbeitsgruppe Technomathematik, Fachbereich Mathematik, University of Kaiserslautern, Germany, May 2002.
14. Balasubramanyam S, Raghurama Rao SV. A new grid-free relaxation scheme for conservation laws. In *Computational Fluid Dynamics 2002*, Armfield S, Morgan R, Srinivas K (eds), *Proceedings of the International Conference on Computational Fluid Dynamics-2*. Springer: Berlin, 2003; 259–264.
15. Deshpande SM, Ghosh AK, Mandal JC. Least squares weak upwind method for Euler equations. *Fluid Mechanics Report*, *89 FM 4*, Department of Aerospace Engineering, Indian Institute of Science, Bangalore, India, 1989.
16. Ghosh AK, Deshpande SM. Least squares kinetic upwind method for inviscid compressible flows. *AIAA Paper 95-1735*, 1995.
17. Jin S. Asymptotic and numerical approximations to multiscale hyperbolic and transport equations, in press.
18. Natalini R. A discrete kinetic approximation of entropy solutions to multidimensional scalar conservation laws. *Journal of Differential Equations* 1999; **148**:292–317.
19. Droillet DA, Natalini R. Discrete kinetic schemes for multidimensional systems of conservation laws. *SIAM Journal on Numerical Analysis* 2000; **37**(6):1973–2004.
20. Jin S. Runge–Kutta methods for hyperbolic conservation laws with stiff relaxation terms. *Journal of Computational Physics* 1995; **122**:51–67.
21. Leveque RJ. Nonlinear conservation laws and finite volume methods for astrophysical fluid flow. *Computational Methods for Astrophysical Fluid Flow*. Springer: Berlin, 1998.
22. Strang G. On the construction and comparison of difference schemes. *SIAM Journal on Numerical Analysis* 1968; **5**:506–517.
23. Coron F, Perthame B. Numerical passage from kinetic to fluid equations. *SIAM Journal on Numerical Analysis* 1991; **28**:26–42.
24. Jameson A. Analysis and design of numerical schemes for gas dynamics 1 artificial diffusion, upwind biasing, limiters and their effect on accuracy and multigrid convergence. *International Journal of Computational Fluid Dynamics* 1995; **4**:171–218.
25. Deshpande SM, Kulkarni PS, Ghosh AK. New developments in kinetic schemes. *Computers and Mathematics with Applications* 1998; **35**(1–2):75–93.
26. Delaundo. An unstructured 2D mesh generation software, http://www.cerfacs.fr/ ~muller/delaundo.html

27. Spekreijse. Multigrid solutions of monotone second order discretizations of hyperbolic conservation laws. *Mathematics of Computation* 1987; **49**:135–155.
28. BAMG, Bidimensional Anisotropic Mesh Generator, INRIA, http://www-rocq.inria.fr
29. Raghurama Rao SV, Balakrishna K. An accurate shock capturing algorithm with a relaxation system for hyperbolic conservation laws. *AIAA Paper No. AIAA-2003-4115*.
30. Tripathy AB. A relaxation scheme for viscous compressible flows. *M.E. Thesis*, Department of Mechanical Engineering, M.S. University of Baroda, Gujarat, India.
31. Tripathy AB, Raghurama Rao SV. A relaxation scheme for viscous compressible flows. *6th Annual AeSI CFD Symposium*, Aeronautical Society of India, Bangalore, August 2003.