BERICHTE
DER
ARBEITSGRUPPE TECHNOMATHEMATIK

FORSCHUNG – AUSBILDUNG – WEITERBILDUNG

New Numerical Schemes based on Relaxation Systems for Conservation Laws

S.V. Raghurama Rao

Bericht 249 Mai 2002

FACHBEREICH MATHEMATIK

New Numerical Schemes based on Relaxation Systems for Conservation Laws

S.V. Raghurama Rao
Visiting Scientist [1]

1 Relaxation Schemes for Hyperbolic Conservation Laws

1.1 Introduction

For the numerical solution of hyperbolic conservation laws, upwind methods have gained popularity in the last two decades. The upwind methods developed for solving Euler equations of gas dynamics can be broadly classified as

1. Riemann solvers (exact or approximate),
2. Flux Vector Splitting methods and
3. Kinetic theory based methods.

Details of the Riemann solvers and Flux Vector Splitting methods are available in the text books by Hirsch [1, 2], Toro [3] and Laney [4]. For a review of *Kinetic Schemes*, also known as *Boltzmann Schemes*, the reader is referred to the review article by Deshpande [5] and also the book by Godlewski and Raviart [6]. Out of these upwind methods, the Riemann solvers have been very popular due to their higher accuracy, which is a consequence of lower numerical dissipation, compared to other schemes. However, the low numerical dissipation also resulted in several problems which made these schemes lose the a part of the property of *robustness* which was initially attributed to these methods. A long list of failings of the Riemann solvers is given by Quirk [7]. Since the recognition of the failures of the Riemann solvers, there has been a renewed interest in the search for non-Riemann solver based numerical methods for solving hyperbolic conservation laws.

Recently, a fourth category of upwind methods is introduced by Jin and Xin [8], called *Relaxation Schemes*. These methods are based on replacing the original non-linear conservation laws by a set of linear equations with non-linear source terms. The interesting feature of these *Relaxation Systems* is the linearity of the convection terms. Coupled with the technique of the *Splitting method* which separates the linear convection terms and the non-linear source terms, these semi-linear Relaxation Systems provide a very simple alternative to the Riemann solvers and complicated flux splittings. The numerical methods based on this technique are termed as *Relaxation Schemes*.

[1] Assistant Professor, ARDB Centre of Excellence for Aerospace CFD, Department of Aerospace Engineering, Indian Institute of Science, Bangalore-560012, India. e-mail : raghu@aero.iisc.ernet.in

The motivation for this article is to give a simple introduction to the *Relaxation Schemes* for solving equations of compressible flows and also to present some interesting research directions the author is involved in, based on the *Relaxation Systems*.

1.2 Relaxation Systems for Non-linear Conservation Laws

Consider a scalar conservation law in one dimension

$$\frac{\partial u}{\partial t} + \frac{\partial g(u)}{\partial x} = 0 \qquad \text{with the initial condition } u(x, t = 0) = u_0(x). \tag{1}$$

Here the flux $g(u)$ is a non-linear function of the dependent variable u. With $g(u) = \frac{u^2}{2}$, we recover the inviscid Burgers equation. The main difficulty in solving this equation numerically is the non-linearity of the flux $g(u)$. Jin and Xin [8] dealt with this problem of non-linearity by introducing a new variable v, which is not an explicit function of the dependent variable u and provided the following system of equations.

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

Here, λ is a positive constant and ϵ is a very small number approaching zero. We can rearrange the second equation of the above system (2) as

$$\epsilon\left[\frac{\partial v}{\partial t} + \lambda^2 \frac{\partial u}{\partial x}\right] = -\left[v - g(u)\right] \tag{3}$$

and as $\epsilon \to 0$, we obtain $v = g(u)$. Substituting this expression in the first equation of the *Relaxation System* (2), we recover the original non-linear conservation law (1). Therefore, in the limit $\epsilon \to 0$, solving the *Relaxation System* (2) is equivalent to solving the original conservation law (1). It is advantageous to work with the *Relaxation System* instead of the original conservation law as the convection terms are not non-linear any more. The source term is still non-linear, and this can be handled easily by the method of splitting, which will be described in the following sections. The initial conditions for the new variable v is given by

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

This initial condition avoids the development of an initial layer, as the initial state is in local equilibrium [8]. The above approach of replacing the non-linear conservation law by a semi-linear *Relaxation System* can be easily extended to vector conservation laws and to multi-dimensions. Consider a vector conservation law in one dimension, given by

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{G}(\mathbf{U})}{\partial x} = 0 \tag{5}$$

Here, $\mathbf{U}$ is the vector of conserved variables and $\mathbf{G}(\mathbf{U})$ is the flux vector, defined by

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ \rho E \end{bmatrix} \text{ and } \mathbf{G}(\mathbf{U}) = \begin{bmatrix} \rho u \\ p + \rho u^2 \\ pu + \rho u E \end{bmatrix} \tag{6}$$

where ρ is the density, u is the velocity, p is the pressure and E is the total internal energy of the fluid, defined by

$$E = \frac{p}{\rho(\gamma - 1)} + \frac{u^2}{2} \tag{7}$$

with γ being the ratio of specific heats. The above vector conservation laws are the Euler equations of gas dynamics and describe the mass, momentum and energy conservation laws for the case of an inviscid compressible fluid flow. The *Relaxation System* for the above vector conservation laws is given by

$$\begin{aligned} \frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{V}}{\partial t} &= 0 \\ \frac{\partial \mathbf{V}}{\partial t} + D\frac{\partial \mathbf{U}}{\partial x} &= -\frac{1}{\epsilon}\left[\mathbf{V} - \mathbf{G}(\mathbf{U})\right] \end{aligned} \tag{8}$$

where D is positive constant diagonal matrix, defined by

$$D = \begin{bmatrix} D_1 & 0 & 0 \\ 0 & D_2 & 0 \\ 0 & 0 & D_3 \end{bmatrix} \tag{9}$$

The positive constants λ in the *Relaxation System* for the scalar case (2) and $D_i, (i = 1, 2, 3)$ in the *Relaxation System* for the vector case (8) are chosen in such a way that the *Relaxation System* is a dissipative approximation to the original non-linear conservation laws. To understand this better, let us do a Chapman-Enskog type expansion for the *Relaxation System*.

1.3 Chapman-Enskog type expansion for the Relaxation System

In this section, a Chapman-Enskog type expansion is performed for the *Relaxation System*, following Jin and Xin [8]. We can rewrite the second equation of the *Relaxation System* (2) as

$$v = g(u) - \epsilon\left[\frac{\partial v}{\partial t} + \lambda^2 \frac{\partial u}{\partial x}\right] \tag{10}$$

which means that

$$v = g(u) + O[\epsilon] \tag{11}$$

Differentiating with respect to time, we obtain

$$\frac{\partial v}{\partial t} = \frac{\partial}{\partial t}\left[g(u)\right] + O[\epsilon] = \frac{\partial g}{\partial u}\frac{\partial u}{\partial t} + O[\epsilon] \tag{12}$$

Since the first equation of the *Relaxation System* (2) gives

$$\frac{\partial u}{\partial t} = -\frac{\partial v}{\partial x} \tag{13}$$

we can write

$$\frac{\partial v}{\partial t} = -\frac{\partial g}{\partial u}\frac{\partial v}{\partial x} + O\left[\epsilon\right] \tag{14}$$

Therefore, using (11), we can write

$$\frac{\partial v}{\partial t} = -\frac{\partial g}{\partial u}\left[\frac{\partial}{\partial x}\left\{g\left(u\right) + O\left[\epsilon\right]\right\}\right] + O\left[\epsilon\right] \tag{15}$$

$$\text{or } \frac{\partial v}{\partial t} = -\frac{\partial g}{\partial u}\left[\frac{\partial g}{\partial u}\frac{\partial u}{\partial x}\right] + O\left[\epsilon\right] = -\left(\frac{\partial g}{\partial u}\right)^2 \frac{\partial u}{\partial x} + O\left[\epsilon\right] \tag{16}$$

Substituting the above expression in (10), we get

$$v = g\left(u\right) - \epsilon\left[-\left\{\left(\frac{\partial g}{\partial u}\right)^2 \frac{\partial u}{\partial x} + O\left[\epsilon\right]\right\} + \lambda^2 \frac{\partial u}{\partial x}\right] \tag{17}$$

or

$$v = g\left(u\right) - \epsilon\left[\frac{\partial u}{\partial x}\left\{\lambda^2 - \left(\frac{\partial g}{\partial u}\right)^2\right\}\right] + O\left[\epsilon^2\right] \tag{18}$$

Substituting this expression for v in the first equation of the *Relaxation System* (2), we get

$$\frac{\partial u}{\partial t} + \frac{\partial g\left(u\right)}{\partial x} = \epsilon\frac{\partial}{\partial x}\left[\frac{\partial u}{\partial x}\left\{\lambda^2 - \left(\frac{\partial g}{\partial u}\right)^2\right\}\right] + O\left[\epsilon^2\right] \tag{19}$$

The right hand side of (19) contains a second derivative of u and hence represents a dissipation (viscous) term. The coefficient represents the coefficient of viscosity. Therefore, the *Relaxation System* provides a vanishing viscosity model to the original conservation law. For the coefficient of dissipation to be positive, the following condition should be satisfied.

$$\lambda^2 \geq \left(\frac{\partial g}{\partial u}\right)^2 \text{ or } -\lambda \leq \left(\frac{\partial g}{\partial u}\right) \leq \lambda \tag{20}$$

This is referred to as the *sub-characteristic condition.* The constant λ in the *Relaxation System* (2) should be chosen in such a way that the condition (20) is satisfied.

For the vector conservation laws (5) modeled by the *Relaxation System* (8), the Chapman–Enskog type expansion gives

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{G}(\mathbf{U})}{\partial x} = \epsilon\frac{\partial}{\partial x}\left[\left\{D - \left(\frac{\partial \mathbf{G}(\mathbf{U})}{\partial \mathbf{U}}\right)^2\right\}\frac{\partial \mathbf{U}}{\partial x}\right] + O(\epsilon^2) \tag{21}$$

For the *Relaxation System* (8) to be dissipative, the following condition should be satisfied.

$$D - \left(\frac{\partial \mathbf{G}(\mathbf{U})}{\partial \mathbf{U}}\right)^2 \geq 0 \tag{22}$$

Based on the eigenvalues of the original conservation laws (5), *i.e.*, Euler equations, Jin and Xin [8] proposed the following two choices.

$$(i) \text{ Define D as } D = \begin{bmatrix} \lambda_1^2 & 0 & 0 \\ 0 & \lambda_2^2 & 0 \\ 0 & 0 & \lambda_3^2 \end{bmatrix}$$

$$\text{First choice } : \lambda^2 = \lambda_1^2 = \lambda_2^2 = \lambda_3^2 = max\left[|u-a|, |u|, |u+a|\right] \tag{23}$$

$$(ii) \text{ Second choice } : \lambda_1^2 = max|u-a|, \lambda_2^2 = max|u| \text{ and } \lambda_3^2 = max|u+a| \tag{24}$$

where u is the fluid velocity and a is the speed of sound. With the first choice, the diagonal matrix D can be written as

$$D = \lambda^2 I \tag{25}$$

where I is a unit matrix.

1.4 Diagonal form of the Relaxation System

The *Relaxation System* (2) can be written in matrix form as

$$\frac{\partial \mathbf{Q}}{\partial t} + A\frac{\partial \mathbf{Q}}{\partial x} = \mathbf{H} \tag{26}$$

$$\text{where } \mathbf{Q} = \begin{bmatrix} u \\ v \end{bmatrix}, \; A = \begin{bmatrix} 0 & 1 \\ \lambda^2 & 0 \end{bmatrix} \text{ and } \mathbf{H} = \begin{bmatrix} 0 \\ -\frac{1}{\epsilon}\left[v - g\left(u\right)\right] \end{bmatrix} \tag{27}$$

As the *Relaxation System* (2) is hyperbolic, so is (26) and, therefore, we can write

$$A = R\Lambda R^{-1} \text{ and consequently } \Lambda = R^{-1}AR \tag{28}$$

where R is the matrix of right eigenvectors of A, R^{-1} is its inverse and Λ is a diagonal matrix with eigenvalues of A as its elements. The expressions for R, R^{-1} and Λ are given by

$$R = \begin{bmatrix} 1 & 1 \\ -\lambda & \lambda \end{bmatrix}, \; R^{-1} = \begin{bmatrix} \frac{1}{2} & -\frac{1}{2\lambda} \\ \frac{1}{2} & \frac{1}{2\lambda} \end{bmatrix} \text{ and } \Lambda = \begin{bmatrix} -\lambda & 0 \\ 0 & \lambda \end{bmatrix} \tag{29}$$

Since the *Relaxation System* (26) is a set of coupled hyperbolic equations, we can decouple it by introducing the *characteristic variables* as

$$\mathbf{f} = R^{-1}\mathbf{Q} \text{ which gives } \mathbf{Q} = R\mathbf{f} \tag{30}$$

Therefore, we can write

$$\frac{\partial \mathbf{Q}}{\partial t} = R\frac{\partial \mathbf{f}}{\partial t} \text{ and } \frac{\partial \mathbf{Q}}{\partial x} = R\frac{\partial \mathbf{f}}{\partial x} \tag{31}$$

Substituting the above expressions in (26), we obtain

$$\frac{\partial}{\partial t} + R^{-1}AR\frac{\partial \mathbf{f}}{\partial x} = R^{-1}\mathbf{H} \tag{32}$$

Using (28), the above equation can be written as

$$\frac{\partial \mathbf{f}}{\partial t} + \Lambda\frac{\partial \mathbf{f}}{\partial x} = R^{-1}\mathbf{H} \tag{33}$$

where

$$\mathbf{f} = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} = R^{-1}\mathbf{Q} = \begin{bmatrix} \dfrac{u}{2} - \dfrac{v}{2\lambda} \\ \\ \dfrac{u}{2} + \dfrac{v}{2\lambda} \end{bmatrix} \text{ and } R^{-1}\mathbf{H} = \begin{bmatrix} \dfrac{1}{2\lambda\epsilon}\left[v - g\left(u\right)\right] \\ \\ -\dfrac{1}{2\lambda\epsilon}\left[v - g\left(u\right)\right] \end{bmatrix} \tag{34}$$

Thus, we obtain two decoupled equations as

$$\begin{aligned} \frac{\partial f_1}{\partial t} - \lambda\frac{\partial f_1}{\partial x} &= \frac{1}{2\lambda\epsilon}\left[v - g\left(u\right)\right] \\ \frac{\partial f_1}{\partial t} + \lambda\frac{\partial f_1}{\partial x} &= -\frac{1}{2\lambda\epsilon}\left[v - g\left(u\right)\right] \end{aligned} \tag{35}$$

Solving these two equations in the limit of $\epsilon \to 0$ is equivalent to solving the original non-linear conservation law (1). It is much easier to solve the above two equations than solving (1), since the convection terms in them are linear. The source terms are still non-linear, but these can be handled easily by the splitting method, which will be described in the following sections. Using (29) and (30), we obtain the expressions

$$u = f_1 + f_2 \text{ and } v = \lambda\left(f_2 - f_1\right) \tag{36}$$

using which we can recover the original variables u and v. In the case of vector conservation laws (5), the diagonal form of the *Relaxation System* leads to

$$\begin{aligned} \frac{\partial \mathbf{f}_1}{\partial t} - \lambda\frac{\partial \mathbf{f}_1}{\partial x} &= \frac{1}{2\lambda\epsilon}\left[\mathbf{V} - \mathbf{G}\left(\mathbf{U}\right)\right] \\ \frac{\partial \mathbf{f}_1}{\partial t} + \lambda\frac{\partial \mathbf{f}_1}{\partial x} &= -\frac{1}{2\lambda\epsilon}\left[\mathbf{V} - \mathbf{G}\left(\mathbf{U}\right)\right] \end{aligned} \tag{37}$$

where $\mathbf{f}_1$ and $\mathbf{f}_2$ are vectors with three components each for the 1-D case.

1.5 Diagonal form as a Discrete Kinetic System

The diagonal form of the *Relaxation System* can be interpreted as a discrete Boltzmann equation [9, 10, 11]. Let us introduce a new variable $\mathbf{F}$ as

$$\mathbf{F} = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix} = \begin{bmatrix} \dfrac{u}{2} - \dfrac{g(u)}{2\lambda} \\ \\ \dfrac{u}{2} + \dfrac{g(u)}{2\lambda} \end{bmatrix} \tag{38}$$

With these new variables, the diagonal form of the *Relaxation System* (33) can be rewritten as

$$\frac{\partial \mathbf{f}}{\partial t} + \Lambda \frac{\partial \mathbf{f}}{\partial x} = \frac{1}{\epsilon}\left[\mathbf{F} - \mathbf{f}\right] \tag{39}$$

This equation is similar to the Boltzmann equation of Kinetic Theory of Gases with a Bhatnagar-Gross-Krook (B-G-K) collision model, except that the molecular velocities are discrete ($-\lambda$ and λ) and the distribution function $\mathbf{f}$ correspondingly has two components, f_1 andf_2. The new variable $\mathbf{F}$ represents the local Maxwellian distribution. This interpretation was used by Natalini [9] and Droillet & Natalini [10] to develop *multi-dimensional Relaxation Systems* which are diagonalizable and new schmes based on them. The classical Boltzmann equation with B-G-K model in one dimension is given by

$$\frac{\partial f}{\partial t} + \xi \frac{\partial f}{\partial \mathbf{x}} = \frac{1}{t_R}\left[F - f\right] \tag{40}$$

where ξ is the molecular velocity, t_R is the relaxation time and F is the equilibrium (Maxwellian) distribution. The Euler equations can be obtained as moments of the Boltzmann equation. The 1-D local Maxwellian for such a case is given by

$$F = \frac{\rho}{I_0}\left(\frac{\beta}{\pi}\right)^{\frac{1}{2}} e^{\left[-\beta(\xi-u)^2 + \frac{I}{I_0}\right]} \tag{41}$$

where ρ is the density, D is the number of translational degrees of freedom, $\beta = \frac{1}{2RT}$, T is the temperature, I is the internal energy variable for the non-translational degrees of freedom and I_0 is the corresponding average internal energy. The moments of the distribution function lead to the macroscopic variables as

$$u = \int_0^\infty dI \int_{-\infty}^\infty d\xi \begin{bmatrix} 1 \\ \xi \\ I + \dfrac{\xi^2}{2} \end{bmatrix} f \;=\; \int_0^\infty dI \int_{-\infty}^\infty d\xi \begin{bmatrix} 1 \\ \xi \\ I + \dfrac{\xi^2}{2} \end{bmatrix} F \tag{42}$$

and

$$g(u) = \int_0^\infty dI \int_{-\infty}^\infty d\xi \begin{bmatrix} 1 \\ \xi \\ I + \dfrac{\xi^2}{2} \end{bmatrix} \xi f \;=\; \int_0^\infty dI \int_{-\infty}^\infty d\xi \begin{bmatrix} 1 \\ \xi \\ I + \dfrac{\xi^2}{2} \end{bmatrix} \xi F \tag{43}$$

The macroscopic equations (Euler equations in this case) are obtained as moments as of the Boltzmann equation by

$$\int_0^\infty dI \int_{-\infty}^{\infty} d\xi \begin{bmatrix} 1 \\ \xi \\ I + \frac{1}{2}\xi^2 \end{bmatrix} \left[\frac{\partial f}{\partial t} + \xi \frac{\partial f}{\partial x} = \frac{1}{t_R}\left[F - f\right]\right] \tag{44}$$

The corresponding expressions for the moments for the discrete Boltzmann equation are

$$u = P\mathbf{f} = P\mathbf{F} \ , \ v = P\Lambda\mathbf{f} \text{ and } g\left(u\right) = P\Lambda\mathbf{F} \text{ where } P = \left[1\ 1\right] \tag{45}$$

for the case of scalar conservation laws and

$$\mathbf{U} = P\mathbf{f} = P\mathbf{F} \ , \ \mathbf{V} = P\Lambda\mathbf{f} \text{ and } \mathbf{G}\left(\mathbf{U}\right) = P\Lambda\mathbf{F} \tag{46}$$

for the case of vector conservation laws. The macroscopic equations are obtained from the discrete Boltzmann equation by multiplying by P and $P\Lambda$ respectively. Let us multiply the discrete Boltzmann equation (39) by P to obtain

$$P\left[\frac{\partial \mathbf{f}}{\partial t} + \Lambda \frac{\partial \mathbf{f}}{\partial x}\right] = P\left[\frac{1}{\epsilon}\left[\mathbf{F} - \mathbf{f}\right]\right] \tag{47}$$

or

$$\frac{\partial\left(P\mathbf{f}\right)}{\partial t} + \frac{\partial\left(P\Lambda\mathbf{f}\right)}{\partial x} = \frac{1}{\epsilon}\left[P\mathbf{F} - P\mathbf{f}\right] \tag{48}$$

Using (45), the above equation can be rewritten as

$$\frac{\partial u}{\partial t} + \frac{\partial v}{\partial x} = 0 \tag{49}$$

which is the first equation of the *Relaxation System* (2). Similarly, multiplying the discrete Boltzmann equation (39) by $P\Lambda$, we obtain

$$\frac{\partial\left(P\Lambda\mathbf{f}\right)}{\partial t} + \frac{\partial\left(P\Lambda^2\mathbf{f}\right)}{\partial x} = \frac{1}{\epsilon}\left[P\Lambda\mathbf{F} - P\Lambda\mathbf{f}\right] \tag{50}$$

Evaluating $P\Lambda^2\mathbf{f}$ as $\lambda^2 u$ and using (45), we get

$$\frac{\partial v}{\partial t} + \lambda^2 \frac{\partial u}{\partial x} = -\frac{1}{\epsilon}\left[v - g\left(u\right)\right] \tag{51}$$

which is the second equation of the *Relaxation System* (2). The *Relaxation System* for the vector conservation laws can also recovered by a similar procedure. In comparison with the classical Boltzmann equation, we can see that recovering the moments are simpler for the *Relaxation System* and therefore the *Relaxation Schemes* will be simpler than the traditional *Kinetic Schemes* in final expressions.

1.6 Multi-dimensional Relaxation Systems

Consider a scalar conservation law in 2-D

$$\frac{\partial u}{\partial t} + \frac{\partial g_1(u)}{\partial x} + \frac{\partial g_2(u)}{\partial y} = 0 \tag{52}$$

The Relaxation System given by Jin and Xin [8] for the above equation is

$$\begin{aligned} \frac{\partial u}{\partial t} + \frac{\partial v_1}{\partial x} + \frac{\partial v_2}{\partial y} &= 0 \\ \frac{\partial v_1}{\partial t} + \lambda_1^2 \frac{\partial u}{\partial x} &= -\frac{1}{\epsilon}\left[v_1 - g_1(u)\right] \\ \frac{\partial v_2}{\partial t} + \lambda_2^2 \frac{\partial u}{\partial x} &= -\frac{1}{\epsilon}\left[v_2 - g_2(u)\right] \end{aligned} \tag{53}$$

We can write the above *Relaxation System* in matrix form as

$$\frac{\partial \mathbf{Q}}{\partial t} + A_1 \frac{\partial \mathbf{Q}}{\partial x} + A_2 \frac{\partial \mathbf{Q}}{\partial y} = H \tag{54}$$

where

$$\mathbf{Q} = \begin{bmatrix} u \\ v_1 \\ v_2 \end{bmatrix}, \; A_1 = \begin{bmatrix} 0 & 1 & 0 \\ \lambda_1^2 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \; A_2 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ \lambda_2^2 & 0 & 0 \end{bmatrix} \text{ and } H = \begin{bmatrix} 0 \\ -\frac{1}{\epsilon}\left\{v_1 - g_1(u)\right\} \\ -\frac{1}{\epsilon}\left\{v_2 - g_2(u)\right\} \end{bmatrix} \tag{55}$$

The matrices A_1 and A_2 do not commute ($A_1 A_2 \neq A_2 A_1$) and the above system is not diagonalizable. This is true in general for the *multi-dimensional Relaxation System* of Jin and Xin (see [9]). As it is preferable to work with a diagonal form, Droillet and Natalini [10] generalize the discrete Boltzmann equation in 1-D to multi-dimensions to obtain a *multi-dimensional Relaxation System* as

$$\frac{\partial \mathbf{f}}{\partial t} + \sum_{k=1}^{D} \Lambda_k \frac{\partial \mathbf{f}}{\partial x_k} = \frac{1}{\epsilon}\left[\mathbf{F} - \mathbf{f}\right] \tag{56}$$

For the multi-dimensional *diagonal Relaxation System*, the local Maxwellians are defined by [10]

$$\begin{aligned} F_{D+1} &= \frac{1}{D}\left[u + \frac{1}{\lambda}\sum_{k=1}^{D} g_k(u)\right] \\ F_i &= -\frac{1}{\lambda} g_i(u) + F_{D+1}\,, \; (i = 1, \cdots, D) \end{aligned} \tag{57}$$

Let us consider the 2-D case for which the local Maxwellians are given by

$$F = \begin{bmatrix} F_1 \\ F_2 \\ F_3 \end{bmatrix} = \begin{bmatrix} \frac{u}{3} - \frac{2}{3\lambda} g_1(u) + \frac{1}{3\lambda} g_2(u) \\ \frac{u}{3} + \frac{1}{3\lambda} g_1(u) - \frac{2}{3\lambda} g_2(u) \\ \frac{u}{3} + \frac{1}{3\lambda} g_1(u) + \frac{1}{3\lambda} g_2(u) \end{bmatrix} \tag{58}$$

Using the definitions

$$u = P\mathbf{f} = P\mathbf{F} \ , \ g_1 = P\Lambda_1 \mathbf{F} \ , \ g_2 = P\Lambda_2 \mathbf{F} \ , \ v_1 = P\Lambda_1 \mathbf{f} \text{ and } v_2 = P\Lambda_2 \mathbf{f} \tag{59}$$

we can obtain

$$P = [1 \ 1 \ 1] \ , \ \Lambda_1 = \begin{bmatrix} -\lambda & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \lambda \end{bmatrix} \ , \ \Lambda_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -\lambda & 0 \\ 0 & 0 & \lambda \end{bmatrix}$$
$$\text{and } \mathbf{f} = \begin{bmatrix} \frac{u}{3} - \frac{2}{3\lambda} v_1 + \frac{1}{3\lambda} v_2 \\ \frac{u}{3} + \frac{1}{3\lambda} v_1 - \frac{2}{3\lambda} v_2 \\ \frac{u}{3} + \frac{1}{3\lambda} v_1 + \frac{1}{3\lambda} v_2 \end{bmatrix} \tag{60}$$

Now, multiplying the 2-D discrete Boltzmann equation by P, $P\Lambda_1$ and $P\Lambda_2$, we can obtain the *2-D Relaxation System* for (52). The 2-D Boltzmann equation is

$$\frac{\partial \mathbf{f}}{\partial t} + \Lambda_1 \frac{\partial \mathbf{f}}{\partial x} + \Lambda_2 \frac{\partial \mathbf{f}}{\partial y} = \frac{1}{\epsilon} [\mathbf{F} - \mathbf{f}] \tag{61}$$

Multiplying the above equation by P, we obtain

$$\frac{\partial (P\mathbf{f})}{\partial t} + \frac{\partial (P\Lambda_1 \mathbf{f})}{\partial x} + \frac{\partial (P\Lambda_2 \mathbf{f})}{\partial y} = \frac{1}{\epsilon} [P\mathbf{F} - P\mathbf{f}] \tag{62}$$

Using (59), the above equation can be simplified to

$$\frac{\partial u}{\partial t} + \frac{\partial v_1}{\partial x} + \frac{\partial v_2}{\partial y} = 0 \tag{63}$$

Similarly, multiplying (61) by $P\Lambda_1$, we get

$$\frac{\partial (P\Lambda_1 \mathbf{f})}{\partial t} + \frac{\partial (P\Lambda_1^2 \mathbf{f})}{\partial x} + \frac{\partial (P\Lambda_1 \Lambda_2 \mathbf{f})}{\partial y} = \frac{1}{\epsilon} [P\Lambda_1 \mathbf{F} - P\Lambda_1 \mathbf{f}] \tag{64}$$

and by multiplying (61) by $P\Lambda_2$, we obtain

$$\frac{\partial\left(P\Lambda_2\mathbf{f}\right)}{\partial t}+\frac{\partial\left(P\Lambda_2\Lambda_1\mathbf{f}\right)}{\partial x}+\frac{\partial\left(P\Lambda_2^2\mathbf{f}\right)}{\partial y}=\frac{1}{\epsilon}\left[P\Lambda_2\mathbf{F}-P\Lambda_2\mathbf{f}\right] \tag{65}$$

Using (60), we can evaluate $P\Lambda_1^2\mathbf{f}$, $P\Lambda_1\Lambda_2\mathbf{f}$, $P\Lambda_2\Lambda_1\mathbf{f}$ and $P\Lambda_2^2\mathbf{f}$ as

$$P\Lambda_1^2\mathbf{f}=\lambda^2\left(f_1+f_3\right)\ ,\ P\Lambda_1\Lambda_2\mathbf{f}=P\Lambda_2\Lambda_1\mathbf{f}=\lambda^2 f_3 \text{ and } P\Lambda_2^2\mathbf{f}=\lambda^2\left(f_2+f_3\right) \tag{66}$$

Using theses expressions, the above two equations get simplified to

$$\frac{\partial v_1}{\partial t}+\frac{\partial}{\partial x}\left[\lambda^2\left(f_1+f_3\right)\right]+\frac{\partial}{\partial y}\left[\lambda^2 f_3\right]=\frac{1}{\epsilon}\left[g_1\left(u\right)-v_1\right] \tag{67}$$

$$\frac{\partial v_2}{\partial t}+\frac{\partial}{\partial x}\left[\lambda^2 f_3\right]+\frac{\partial}{\partial y}\left[\lambda^2\left(f_2+f_3\right)\right]=\frac{1}{\epsilon}\left[g_1\left(u\right)-v_1\right] \tag{68}$$

Using the definitions (59) in (63), (67) and (68), after some algebraic manipulation, we obtain the *2-D Relaxation System* as

$$\begin{aligned}\frac{\partial u}{\partial t}+\frac{\partial v_1}{\partial x}+\frac{\partial v_2}{\partial y}&=0\\ \frac{\partial v_1}{\partial t}+\frac{\lambda}{3}\frac{\partial}{\partial x}\left[2\left(\lambda u+v_1+v_2\right)-3v_1\right]+\frac{\lambda}{3}\frac{\partial}{\partial y}\left[\lambda u+v_1+v_2\right]&=\frac{1}{\epsilon}\left[g_1\left(u\right)-v_1\right]\\ \frac{\partial v_x}{\partial t}+\frac{\lambda}{3}\frac{\partial}{\partial x}\left[\lambda u+v_1+v_2\right]+\frac{\lambda}{3}\frac{\partial}{\partial y}\left[2\left(\lambda u+v_1+v_2\right)-3v_2\right]&=\frac{1}{\epsilon}\left[g_2\left(u\right)-v_2\right]\end{aligned} \tag{69}$$

This *Relaxation System* is different from the non-diagonalizable *Relaxation System* of Jin and Xin (53). The above derivation was done only to see the type of *multi-dimensional Relaxation System* we obtain from the multi-dimensional discrete Boltzmann equation. It is not necessary, however, to use the above *Relaxation System*, and it is sufficient to use the multi-dimensional discrete Boltzmann equation as a starting point to derive *Relaxation Schemes*. For the 2-D cases presented in this report the following equation is used as a starting point

$$\frac{\partial\mathbf{f}}{\partial t}+\Lambda_1\frac{\partial\mathbf{f}}{\partial x}+\Lambda_2\frac{\partial\mathbf{f}}{\partial y}=\frac{1}{\epsilon}\left[\mathbf{F}-\mathbf{f}\right] \tag{70}$$

which, when expanded, leads to the following equations.

$$\begin{aligned}\frac{\partial f_1}{\partial t}-\lambda\frac{\partial f_1}{\partial x}&=\frac{1}{\epsilon}\left[F_1-f_1\right]\\ \frac{\partial f_2}{\partial t}-\lambda\frac{\partial f_2}{\partial y}&=\frac{1}{\epsilon}\left[F_2-f_2\right]\\ \frac{\partial f_3}{\partial t}+\lambda\frac{\partial f_1}{\partial x}+\lambda\frac{\partial f_3}{\partial y}&=\frac{1}{\epsilon}\left[F_3-f_3\right]\end{aligned} \tag{71}$$

For 2-D Euler equations, the expanded form of the discrete Boltzmann equation is given by

$$\begin{aligned} \frac{\partial \mathbf{f}_1}{\partial t} - \lambda \frac{\partial \mathbf{f}_1}{\partial x} &= \frac{1}{\epsilon}\left[\mathbf{F}_1 - \mathbf{f}_1\right] \\ \frac{\partial \mathbf{f}_2}{\partial t} - \lambda \frac{\partial \mathbf{f}_2}{\partial y} &= \frac{1}{\epsilon}\left[\mathbf{F}_2 - \mathbf{f}_2\right] \\ \frac{\partial \mathbf{f}_3}{\partial t} + \lambda \frac{\partial \mathbf{f}_1}{\partial x} + \lambda \frac{\partial \mathbf{f}_3}{\partial y} &= \frac{1}{\epsilon}\left[\mathbf{F}_3 - \mathbf{f}_3\right] \end{aligned} \tag{72}$$

where $\mathbf{f}_i$ and $\mathbf{F}_i$ for each i, $(i = 1,2,3)$ are vectors with 4 components each similar to the corresponding moments $\mathbf{U}$, $\mathbf{V}_1$, $\mathbf{V}_2$, $\mathbf{G}_1$ and $\mathbf{G}_2$. The initial condition for $\mathbf{f}$ is prescribed as

$$\mathbf{f}\left(x,y,t=0\right) = \mathbf{F}\left(\mathbf{U}\left(x,y,t=0\right)\right) \tag{73}$$

The Chapman–Enskog type analysis for the above *multi–dimensional Relaxation System* to be a dissipative approximation to the original conservation laws leads to the following condition (see [10, 13] for details of the derivation).

$$\lambda \geq max\left(-A_1 - A_2, 2A_1 - A_2, -A_1 + 2A_2\right) \tag{74}$$

where

$$A_1 = \frac{\partial \mathbf{G}_1\left(\mathbf{U}\right)}{\partial \mathbf{U}} \text{ and } A_2 = \frac{\partial \mathbf{G}_2\left(\mathbf{U}\right)}{\partial \mathbf{U}} \tag{75}$$

1.7 Relaxation Schemes

Based on the model of a *Relaxation System* to the non-linear conservation laws, Jin and Xin [8] developed upwind schemes for conservation laws, termed as *Relaxation Schemes*. Consider the 1-D *Relaxation System* (2) modeling the scalar conservation law (1). For solving this *Relaxation System* which has a stiff non-linear term on the right hand side of the second equation, following Jin and Xin [8], we can use an operator splitting by which the solution procedure is split into the following two steps.

(i) Convection Step :

$$\begin{aligned} \frac{\partial u}{\partial t} + \frac{\partial v}{\partial x} &= 0 \\ \frac{\partial v}{\partial t} + \lambda^2 \frac{\partial u}{\partial x} &= 0 \end{aligned} \tag{76}$$

In this convection step, an upwind method is used. Consider a 3-point stencil as shown in Figure 1. Integrating the equations of the convection step (76) over the finite volume $[j-\frac{1}{2}, j+\frac{1}{2}]$ from time t^n to t^{n+1}, we obtain

$$\begin{aligned} u_j^{n+1} &= u_j^n - \frac{\Delta t}{\Delta x}\left[v_{j+\frac{1}{2}}^n - v_{j-\frac{1}{2}}^n\right] \\ v_j^{n+1} &= v_j^n - \lambda^2 \frac{\Delta t}{\Delta x}\left[u_{j+\frac{1}{2}}^n - u_{j-\frac{1}{2}}^n\right] \end{aligned} \tag{77}$$

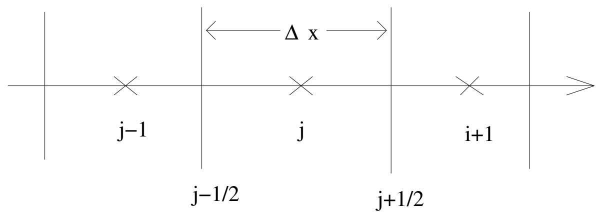

Figure 1: 3-point stencil

where the cell integral averages u_j and v_j are defined by

$$
\begin{aligned}
u_j &= \frac{1}{\Delta x}\int_{j-\frac{1}{2}}^{j+\frac{1}{2}} u\left(x\right) dx \\
v_j &= \frac{1}{\Delta x}\int_{j-\frac{1}{2}}^{j+\frac{1}{2}} v\left(x\right) dx
\end{aligned}
\tag{78}
$$

To calculate $v^n_{j\pm\frac{1}{2}}$ and $u^n_{j\pm\frac{1}{2}}$ with an upwind approximation, following Jin and Xin [8], we can use the characteristic variables of the relaxation system, f_1 and f_2 (see expression (34)) as

$$
f_{1,j+\frac{1}{2}} = f_{1,j+1} \text{ and } f_{2,j+\frac{1}{2}} = f_{2,j} \tag{79}
$$

or

$$
\frac{1}{2}u_{j+\frac{1}{2}} - \frac{1}{2\lambda}v_{j+\frac{1}{2}} = \frac{1}{2}u_{j+1} - \frac{1}{2\lambda}v_{j+1} \text{ and } \frac{1}{2}u_{j+\frac{1}{2}} + \frac{1}{2\lambda}v_{j+\frac{1}{2}} = \frac{1}{2}u_j + \frac{1}{2\lambda}v_j \tag{80}
$$

Solving the above two equations, we obtain

$$
\begin{aligned}
u_{j+\frac{1}{2}} &= \frac{1}{2}\left[u_j + u_{j+1}\right] - \frac{1}{2\lambda}\left[v_{j+1} - v_j\right] \\
v_{j+\frac{1}{2}} &= \frac{1}{2}\left[v_j + v_{j+1}\right] - \frac{\lambda}{2}\left[u_{j+1} - u_j\right]
\end{aligned}
\tag{81}
$$

Using these expressions, the final upwind scheme in the convection step can be written as

$$
\begin{aligned}
u_j^{n+1} &= u_j^n - \frac{\Delta t}{\Delta x}\left[\frac{1}{2}\left(v_{j+1} - v_{j-1}\right) - \frac{\lambda}{2}\left(u^n_{j+1} - 2u^n_j + u^n_{j-1}\right)\right] \\
v_j^{n+1} &= v_j^n - \frac{\Delta t}{\Delta x}\left[\frac{\lambda^2}{2}\left(u^n_{j+1} - u^n_{j-1}\right) - \frac{\lambda}{2}\left(v^n_{j+1} - 2v^n_j + v^n_{j-1}\right)\right]
\end{aligned}
\tag{82}
$$

(ii) Source Step :

$$
\frac{dv}{dt} = -\frac{1}{\epsilon}\left[v - g\left(u\right)\right] \tag{83}
$$

The above ordinary differential equation is solved by an implicit method which avoids the time-step being dependent on ϵ, as

$$\frac{v_j^{n+1} - v_j^n}{\Delta t} = -\frac{1}{\epsilon}\left[v^{n+1} - g\left(u^{n+1}\right)\right] \tag{84}$$

The above strategy of splitting the method into a relaxation step and a convection step involves a *splitting error* which makes the method first order accurate in time, irrespective of whatever higher order discretization is used in time and space in both the steps. To rectify this defect, *Strang splitting* is needed in which relaxation step is solved for half a time-step ($\frac{\Delta t}{2}$), followed by a convection step for full time-step (Δt) and then again by a relaxation step for half a time-step ($\frac{\Delta t}{2}$). Even this Strang splitting degenerates to formal first order accuracy in time in the limit $\epsilon \to 0$, as demonstrated by Jin [14]. As a remedy, Jin [14] introduced a *modified Strang splitting* strategy which was used in [8], along with MUSCL scheme for higher order accuracy in space. The modified Strang splitting introduced by Jin is given below.

$$\begin{aligned}
u^* &= u^n \\
v^* &= v^n + \frac{\Delta t}{\epsilon}\left[v^* - g\left(u^*\right)\right] \\
u^1 &= u^* - \Delta t \Delta_+ v^* \\
v^1 &= v^* - \Delta t \lambda^2 \Delta_+ v^* \\
u^{**} &= u^1 \\
v^{**} &= v^1 - \frac{\Delta t}{\epsilon}\left[v^{**} - g\left(u^{**}\right)\right] - \frac{2\Delta t}{\epsilon}\left[v^* - g\left(u^*\right)\right] \\
u^2 &= u^{**} - \Delta t \Delta_+ v^{**} \\
v^2 &= v^{**} - \Delta t \lambda^2 \Delta_+ v^{**} \\
u^{n+1} &= \frac{1}{2}\left(u^n + u^2\right) \\
v^{n+1} &= \frac{1}{2}\left(v^n + v^2\right)
\end{aligned} \tag{85}$$

where

$$\Delta_+ w = \frac{1}{\Delta x}\left(w_{j+\frac{1}{2}} - w_{j-\frac{1}{2}}\right) \tag{86}$$

The reader is referred to the above references for further details. A different point of view of this problem will be given in section (2.4) in this paper.

2 New Developments in Relaxation Schemes

The framework of *Relaxation Schemes* for solving non-linear conservation laws has proved to be very fruitful and led to a lot of research work by applied mathematicians. The reader is referred to the review article by Natalini [11] and the references therein for an account of this research and Leveque & Pelanti [12] for the connection between *Relaxation Schemes* and Riemann Solvers. In the following sections, some new developments in *Relaxation Schemes* that the author is involved in are presented.

2.1 A Grid-Free Relaxation Scheme

Grid generation is an important aspect of numerical simulation of fluid flows on computers. Grid generation around complex geometries is often a difficult and time-consuming task. To reduce the efforts required for the grid generation, there has been a search for numerical algorithms that are grid-free in the recent past, in the sense that they should be able to operate on any arbitrary distribution of points. The *Least Squares Kinetic Upwind Method* (LSKUM) of Ghosh and Deshpande [15] is an important development in this research. Balasubramanyam and Raghurama Rao [16, 17, 18, 13] used the idea of *Least Squares Upwinding* coupled with a *Relaxation System* to develop a *Grid-free Upwind Relaxation Scheme* for solving compressible flows. The basic idea of *Least Squares Upwinding* is explained in the following section.

2.1.1 Least Squares Upwinding for a Grid-Free Scheme

Consider a linear convection equation in one dimension

$$\frac{\partial f}{\partial t} + \lambda \frac{\partial f}{\partial x} = 0 \tag{87}$$

Least Squares Upwind method can be used to discretize the above equation on an arbitrary distribution of points. Consider the stencil of an arbitrarily distributed points as shown in the Figure 2. Taylor expansion around the point P gives, for any neighbouring point i

Figure 2: Arbitrarily distributed points in 1-D

$$f_i = f_P + (x_i - x_P)\left(\frac{\partial f}{\partial x}\right)_P + O\left[(x_i - x_P)^2\right] \tag{88}$$

Defining an error of the approximation by e, we can write

$$e = f_i - f_P - (x_i - x_P)\left(\frac{\partial f}{\partial x}\right)_P \tag{89}$$

The sum of squares of the error is given by

$$E = \sum_{i=1}^{n} e^2 \tag{90}$$

where n is the number of the neighbouring points. Let us now introduce some notations for simplicity.

$$\Delta x_i = x_i - x_P \ , \ \Delta f_i = f_i - f_P \text{ and } f_x = \left(\frac{\partial f}{\partial x}\right)_P \tag{91}$$

Therefore, we can write

$$e = \Delta f_i - \Delta x_i f_x \text{ and } E = \sum_{i=1}^{n} \left(\Delta f_i - \Delta x_i f_x\right)^2 \tag{92}$$

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

$$\frac{\partial}{\partial\left(f_x\right)} E = 0 \text{ or } \frac{\partial}{\partial\left(f_x\right)}\left[\sum_{i=1}^{n} \left(\Delta f_i - \Delta x_i f_x\right)^2\right] = 0 \tag{93}$$

from which we obtain

$$f_x = \frac{\sum_i^n \Delta f_i \Delta x_i}{\sum_i^n \Delta x_i^2} \tag{94}$$

The above expression gives the formula for calculating a derivative at a point P from the values of its neighbours i, in whatever way the neighbouring points are distributed. We can use it for discretizing the space derivatives in the equation (87). The above expression for the derivative does not give preference to any direction and, therefore, is like central differencing. To use it in an upwinding framework, let us introduce *weighted Least Squares minimization* by defining

$$E = \sum_{i=1}^{n} w_i e^2 \tag{95}$$

where w_i are the weights to be prescribed in such a way that for the upwind points the weight is unity or zero otherwise. With the above modified definition of the sum of squares of the error, the minimization with respect to the derivative leads to the following formula for the derivative.

$$f_x = \frac{\sum_i^n w_i \Delta f_i \Delta x_i}{\sum_i^n w_i \Delta x_i^2} \tag{96}$$

We can now use C-I-R type splitting in (87) to obtain

$$\frac{\partial f}{\partial t} + \frac{\lambda + |\lambda|}{2} \frac{\sum_i^n w_i \Delta f_i \Delta x_i}{\sum_i^n w_i \Delta x_i^2} + \frac{\lambda - |\lambda|}{2} \frac{\sum_i^n w_i \Delta f_i \Delta x_i}{\sum_i^n w_i \Delta x_i^2} = 0 \tag{97}$$

With weights defined by

$$\begin{array}{ll} w_i = 1 & if\ \lambda > 0 \text{ and } \Delta x_i < 0\ (i.e., i \text{ is on the left side of } P) \\ w_i = 0 & if\ \lambda > 0 \text{ and } \Delta x_i > 0\ (i.e., i \text{ is on the right side of } P) \\ w_i = 1 & if\ \lambda < 0 \text{ and } \Delta x_i > 0 \\ w_i = 0 & if\ \lambda < 0 \text{ and } \Delta x_i < 0 \end{array} \tag{98}$$

the above discretization leads to a *grid–free upwind scheme* which can operate on any arbitrary distribution of points. The above weights ensure that only the points upwind to P will contribute.

2.1.2 Second Order Accurate Least Square Upwinding

A second order accurate Least Squares Upwind method can be derived for an arbitrary distribution of points by using the two-step procedure introduced by Ghosh and Deshpande [15]. The advantage with this procedure is that the stencil size will still be compact, without increasing the number of points. Consider the Taylor Series expansion as

$$f_i = f_P + \left(\frac{\partial f}{\partial x}\right)_P \Delta x_i + \left(\frac{\partial^2 f}{\partial x^2}\right)_P \frac{(\Delta x_i)^2}{2} + O\left(\Delta x_i^3\right) \tag{99}$$

or

$$\Delta f_i = \left(\frac{\partial f}{\partial x}\right)_P \Delta x_i + \left(\frac{\partial^2 f}{\partial x^2}\right)_P \frac{(\Delta x_i)^2}{2} + O\left(\Delta x_i^3\right) \tag{100}$$

Differentiating the above equation with respect to x, we obtain

$$\frac{\partial}{\partial x}\left(\Delta f_i\right) = \Delta x_i \left(\frac{\partial^2 f}{\partial x^2}\right)_P + O\left(\Delta x_i^2\right) \tag{101}$$

We can use the above expression to eliminate the $\left(\frac{\partial^2 f}{\partial x^2}\right)_P$ term in the Taylor expansion to obtain a second order accurate expression for the derivative. Defining

$$\Delta \tilde{f}_i = \Delta f_i - \frac{\Delta x_i}{2}\frac{\partial}{\partial x}\left(\Delta f_i\right) \tag{102}$$

and substituting (100) and (101) in (102), we get

$$\begin{aligned} \Delta \tilde{f}_i &= \Delta x_i \left(\frac{\partial f}{\partial x}\right)_P + \frac{\Delta x_i^2}{2}\left(\frac{\partial^2 f}{\partial x^2}\right)_P + O\left(\Delta x_i^3\right) \\ &- \frac{\Delta x_i}{2}\left[\Delta x_i \left(\frac{\partial^2 f}{\partial x^2}\right)_P + O\left(\Delta x_i^2\right)\right] \end{aligned} \tag{103}$$

The second order terms get canceled and we get

$$\Delta\tilde{f}_i = \Delta x_i \left(\frac{\partial f}{\partial x}\right)_P + O\left(\Delta x_i^3\right) \tag{104}$$

Note that the truncation part in the above equation (104) is $O\left(\Delta x_i^3\right)$. Let us now redefine the error as

$$e = \Delta\tilde{f}_i - \Delta x_i \left(\frac{\partial f}{\partial x}\right)_P \tag{105}$$

and the sum of the squares of errors as

$$E = \sum_{i=1}^{n} \left(\Delta\tilde{f}_i - \Delta x_i f_x\right)^2 \tag{106}$$

Now minimizing E w.r.t f_x we obtain

$$\left(\frac{\partial f}{\partial x}\right)_P = \frac{\sum_{i=1}^{n} \Delta x_i \Delta\tilde{f}_i}{\sum_{i=1}^{n} \Delta x_i^2} \tag{107}$$

where

$$\Delta\tilde{f}_i = \Delta f_i - \frac{\Delta x_i}{2}\left[\left(\frac{\partial f}{\partial x}\right)_i - \left(\frac{\partial f}{\partial x}\right)_P\right] \tag{108}$$

Since the error e is defined with a truncation term $O\left[\Delta x_i^3\right]$, the above expression (107) is second order accurate. The derivatives $(f_x)_P$ and $(f_x)_i$ are evaluated using the expression for the first order accurate derivative given in the previous sub-section, *i.e.*, expressoin (96). 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 sub-section. The spurious oscillations which will appear in the second order solution can be suppressed by using a simple *minmax* limiter. The procedure involves calculating the local minima and maxima of the data and limiting the second order solution within those limits. The reader is referred to [16, 18, 13] for details.

2.1.3 Least Squares Upwinding in 2-D

Consider a point P surrounded by a set of neighbouring points distributed arbitrarily, as shown in the Figure 3. Taylor Series expansion gives, for a set of points $i = 1, 2, \cdots, n$ around P,

$$f_i = f_P + (x_i - x_P)\left(\frac{\partial f}{\partial x}\right)_P + (y_i - y_P)\left(\frac{\partial f}{\partial y}\right)_P + O\left[(x_i - x_P)^2, (y_i - y_P)^2\right] \tag{109}$$

Defining the error as

$$e = \Delta f_i - \Delta x_i f_x - \Delta y_i f_y \tag{110}$$

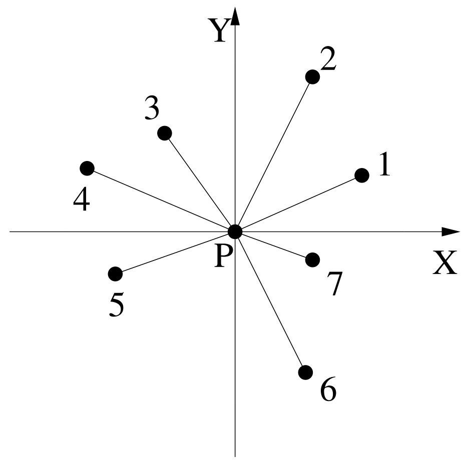

Figure 3: Arbitrary distribution of points in 2 dimensions

we obtain the sum of squares of the error as

$$E = \sum_{i=1}^{n} \left(\Delta f_i - \Delta x_i f_x - \Delta y_i f_y\right)^2 \tag{111}$$

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

$$f_x = \frac{\sum \Delta y_i^2 \sum \Delta x_i \Delta f_i - \sum \Delta x_i \Delta y_i \sum \Delta y_i \Delta f_i}{\sum \Delta x_i^2 \sum \Delta y_i^2 - \left(\sum \Delta x_i \Delta y_i\right)^2} \tag{112}$$

$$f_y = \frac{\sum \Delta x_i^2 \sum \Delta y_i \Delta f_i - \sum \Delta x_i \Delta y_i \sum \Delta x_i \Delta f_i}{\sum \Delta x_i^2 \sum \Delta y_i^2 - \left(\sum \Delta x_i \Delta y_i\right)^2} \tag{113}$$

where $\sum$ stands for $\sum_{i=1}^{n}$. The above two expressions (112) and (113) give approximations for space derivatives for an arbitrary distributions of n points, w.r.t. x and y respectively. Consider a 2-D linear convection equation as

$$\frac{\partial f}{\partial t} + \lambda_1 \frac{\partial f}{\partial x} + \lambda_2 \frac{\partial f}{\partial y} = 0 \tag{114}$$

We can introduce C-I-R splitting in the above equation as

$$\frac{\partial f}{\partial t} + \frac{\lambda_1 + |\lambda_1|}{2}\frac{\partial f}{\partial x} + \frac{\lambda_1 - |\lambda_1|}{2}\frac{\partial f}{\partial x} + \frac{\lambda_2 + |\lambda_2|}{2}\frac{\partial f}{\partial y} + \frac{\lambda_2 - |\lambda_2|}{2}\frac{\partial f}{\partial y} = 0 \tag{115}$$

Following a similar procedure as in 1-D case, weighted least squares approximations for the derivatives can be obtained as given below.

$$f_x = \frac{\sum w_i \Delta y_i^2 \sum w_i \Delta x_i \Delta f_i - \sum w_i \Delta x_i \Delta y_i \sum w_i \Delta y_i \Delta f_i}{\sum w_i \Delta x_i^2 \sum w_i \Delta y_i^2 - \left(\sum w_i \Delta x_i \Delta y_i\right)^2} \tag{116}$$

$$f_y = \frac{\sum w_i \Delta x_i^2 \sum w_i \Delta y_i \Delta f_i - \sum w_i \Delta x_i \Delta y_i \sum w_i \Delta x_i \Delta f_i}{\sum w_i \Delta x_i^2 \sum w_i \Delta y_i^2 - \left(\sum w_i \Delta x_i \Delta y_i\right)^2} \tag{117}$$

where $\sum$ stands for $\sum_{i=1}^{n}$. The weights w_i are chosen in such a way that 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 (115) leads to the following set of weights.

$$w_i = \begin{cases} 1 & \text{for } \Delta x_i < 0 \text{ and } \lambda_1 > 0 \\ 1 & \text{for } \Delta x_i > 0 \text{ and } \lambda_1 < 0 \\ 1 & \text{for } \Delta y_i < 0 \text{ and } \lambda_2 > 0 \\ 1 & \text{for } \Delta y_i > 0 \text{ and } \lambda_2 < 0 \\ 0 & \text{otherwise} \end{cases} \tag{118}$$

Without introducing the weights, the upwinding can also be enforced by dividing the stencil appropriately. For example, 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, points on the top side of x - axis and points on the bottom side of the x - axis. Let $N(P)$ be the set of all neighbouring nodes to point P, defined as

$$N(P) = \{i,\ 1 \leq i \leq n\} \tag{119}$$

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

$$\begin{aligned} N_1(P) &= \{i,\ i \in N(P) \text{ and } \Delta x_i < 0\} \\ N_2(P) &= \{i,\ i \in N(P) \text{ and } \Delta x_i > 0\} \\ N_3(P) &= \{i,\ i \in N(P) \text{ and } \Delta y_i < 0\} \\ N_4(P) &= \{i,\ i \in N(P) \text{ and } \Delta y_i > 0\} \end{aligned} \tag{120}$$

The above four sub-stencils are shown pictorially in the Figure (4). Using (116), (117) and (120), (115) can be written as

$$\begin{aligned} \frac{\partial f}{\partial t} &+ \frac{\lambda_1 + |\lambda_1|}{2} \left(\frac{\sum \Delta y_i^2 \sum \Delta x_i \Delta f_i - \sum \Delta x_i \Delta y_i \sum \Delta y_i \Delta f_i}{\sum \Delta x_i^2 \sum \Delta y_i^2 - \left(\sum \Delta x_i \Delta y_i\right)^2} \right)_{N_1(P)} \\ &+ \frac{\lambda_1 - |\lambda_1|}{2} \left(\frac{\sum \Delta y_i^2 \sum \Delta x_i \Delta f_i - \sum \Delta x_i \Delta y_i \sum \Delta y_i \Delta f_i}{\sum \Delta x_i^2 \sum \Delta y_i^2 - \left(\sum \Delta x_i \Delta y_i\right)^2} \right)_{N_2(P)} \\ &+ \frac{\lambda_2 + |\lambda_2|}{2} \left(\frac{\sum \Delta x_i^2 \sum \Delta y_i \Delta f_i - \sum \Delta x_i \Delta y_i \sum \Delta x_i \Delta f_i}{\sum \Delta x_i^2 \sum \Delta y_i^2 - \left(\sum \Delta x_i \Delta y_i\right)^2} \right)_{N_3(P)} \\ &+ \frac{\lambda_2 - |\lambda_2|}{2} \left(\frac{\sum \Delta x_i^2 \sum \Delta y_i \Delta f_i - \sum \Delta x_i \Delta y_i \sum \Delta x_i \Delta f_i}{\sum \Delta x_i^2 \sum \Delta y_i^2 - \left(\sum \Delta x_i \Delta y_i\right)^2} \right)_{N_4(P)} \\ &= 0 \end{aligned} \tag{121}$$

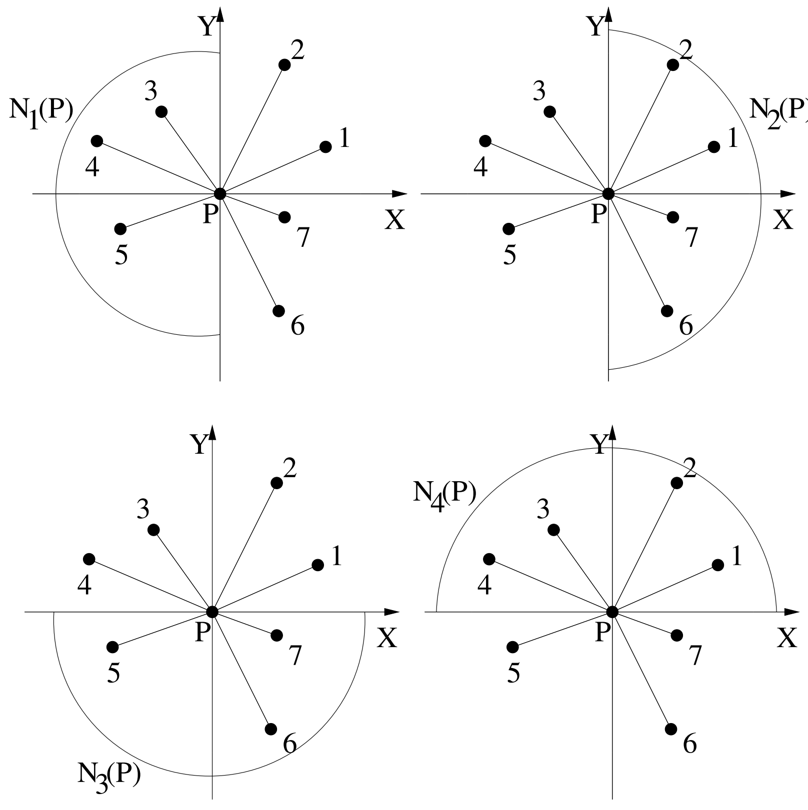

Figure 4: Stencil splitting for upwinding (x-y splitting)

The subscripts in (121) indicate the particular stencil with which the derivative is calculated. Using the weights as shown in (118) is equivalent to using the x - y split stencils as in (121). The above mentioned procedure is termed as *x-y splitting* [15]. One more way of splitting, called as *quadrant splitting* is also possible, in which the information coming from each quadrant is treated separately for upwinding. The reader is referred to [15, 13] for details. The second order accuracy in 2-D is achieved by a two-step procedure on a compact stencil, as in the 1-D case and a similar minmax limiter is used, the details of which are available in the above mentioned references.

2.1.4 Grid-free Relaxation Scheme and Results for Bench-mark Problems

The Least Squares Upwind method is applied to the *Relaxation System* for inviscid (non-linear) Burgers equation and Euler equations to obtain a *Grid-free Upwind Relaxation Scheme* and is tested on some bench-mark test problems in 1-D and 2-D.

1-D Burgers equation : This test case is for the inviscid Burgers equation

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x}\left(\frac{u^2}{2}\right) = 0 \tag{122}$$

with the following initial condition.

$$u = \begin{cases} 1 \text{ for } |x| < \frac{1}{2} \\ -1 \text{ for } \frac{1}{2} < |x| \leq 1 \end{cases} \tag{123}$$

The exact solution and the numerical solution obtained with the *Grid-free Relaxation Scheme* with both uniform distribution of 200 points and an arbitrary distribution of 200 points (obtained by random number generation) are plotted for $u(x, 0.3)$ in the Figures 5 and 6.

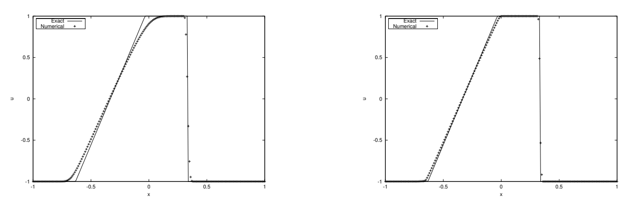

Figure 5: First order and second order accurate solutions with Grid-free Relaxation Scheme on a uniform grid

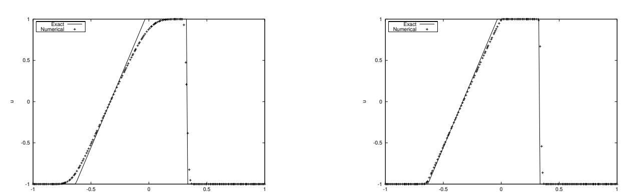

Figure 6: First order and second order accurate solutions with Grid-free Relaxation Scheme on an arbitrarily distributed points

2-D Burgers equation : The 2-D Burgers equation, given by

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x}\left[u^2\right] + \frac{\partial u}{\partial y} = 0 \tag{124}$$

is solved for the following two test cases.
Test case 1 : The initial conditions are :

$$\begin{aligned} u(0, y) &= 1 && \text{for } 0 < y < 1 \\ u(1, y) &= -1 && \text{for } 0 < y < 1 \\ u(x, 0) &= 1 - 2x && \text{for } 0 < x < 1 \end{aligned} \tag{125}$$

This test case models a normal shock and an expansion fan. The solutions obtained with the *Grid-free Relaxation Scheme* on a distribution of points obtained by a triangular mesh generator are shown in the Figure 7.

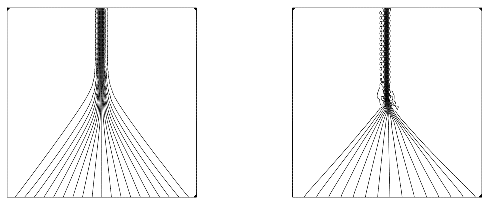

Figure 7: First order and second order accurate solutions with Grid-free Relaxation Scheme on a distribution of points obtained from a triangular mesh generator

Test case 2 : The initial conditions are :

$$\begin{aligned} u(0,y) &= 1.5 && \text{for } 0 < y < 1 \\ u(1,y) &= -0.5 && \text{for } 0 < y < 1 \\ u(x,0) &= 1.5 - 2x && \text{for } 0 < x < 1 \end{aligned} \tag{126}$$

This test case models an oblique shock and an expansion fan. The solutions obtained with the *Grid-free Relaxation Scheme* on a distribution of points obtained by a triangular mesh generator are shown in the Figure 8.

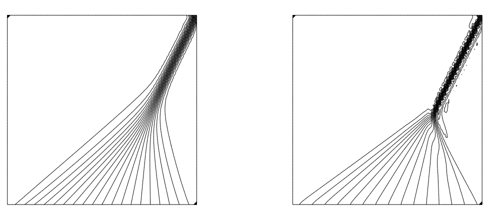

Figure 8: First order and second order accurate solutions with Grid-free Relaxation Scheme on a distribution of points obtained from a triangular mesh generator

Euler equations :
1-D shock tube test case : The Sod's test case is solved with the following initial conditions.

$$\begin{array}{ll} \rho_L = 1.0 & \rho_R = 0.125 \\ u_L = 0.0 & u_R = 0.0 \\ p_L = 1.0 & p_R = 0.1 \end{array} \tag{127}$$

The results obtained on a uniform distribution and a random distributoin of 200 points are shown in Figures 9 and 10.

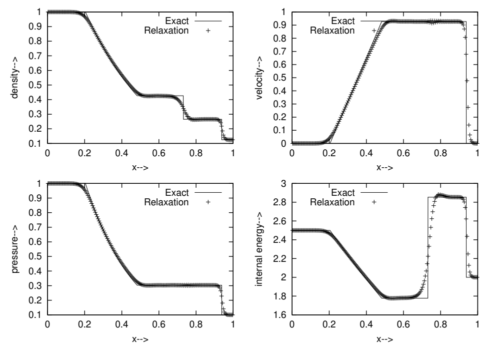

Figure 9: Second order accurate solutions with Grid-free Relaxation Scheme on a uniform distribution of points

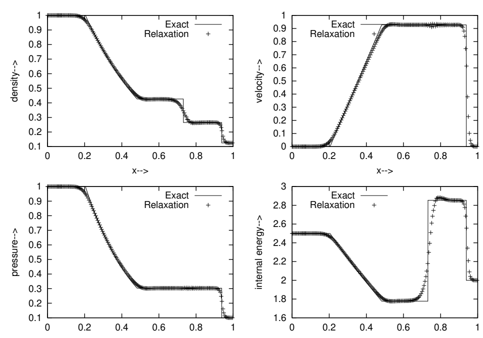

Figure 10: Second order accurate solutions with Grid-free Relaxation Scheme on a distribution of points obtained from a random number generator

Test cases for 2-D Euler equations : The results obtained with the *Grid-free Relaxation Scheme* for the case of a shock reflection problem and an internal flow in a channel around a ramp are presented in Figures 11 to 13. For the ramp case, the point distribution is obtained from a triangular mesh generator. The framework of the present *Grid-free Relaxation Scheme*, is easily amenable to mesh adaptation, as only the local connectivity for a point changes and the Least Squares Upwinding adapts to it easily. Some results are also shown here with mesh adaptation, in Figures 12 and 13. The reader is referred to [18, 13] for more results with this scheme. This new idea of developing a grid-free scheme based on *Relaxation Systems* for non-linear conservation laws has proved to be quite fruitful and the idea appears to have potential for further research.

Figure 11: Second order accurate solutions with Grid-free Relaxation Scheme for the shock reflection problem on (i) 1452 points and (ii) 5667 points obtained from a triangular mesh generator

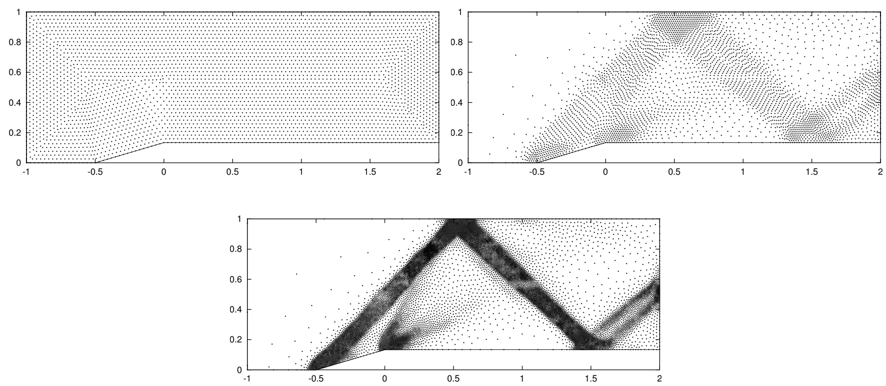

Figure 12: Mesh adaptation with second order accurate Grid-free Relaxation Scheme for the supersonic flow over a ramp in a channel : (i) Initial distribution : 3952 points, (ii) First adaptation : 3674 points, (iii) Second adaptation : 10841 points

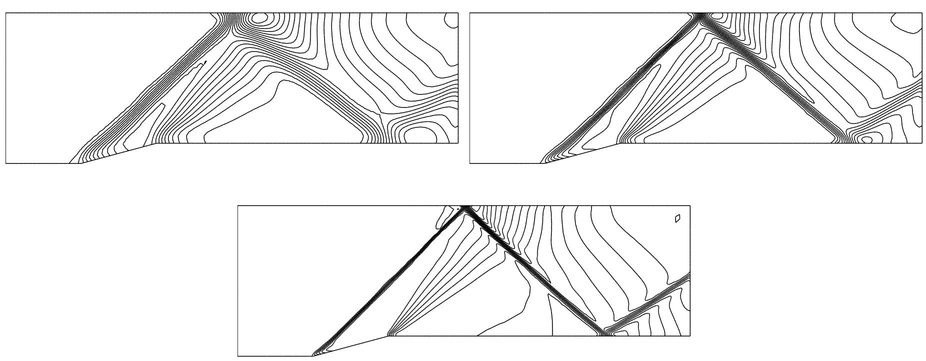

Figure 13: Pressure contours with second order accurate Grid-free Relaxation Scheme for the supersonic flow over a ramp in a channel with different levels of mesh adaptation

2.2 An Unsplit Relaxation Scheme

Achieving second order accuracy with the Relaxation Schemes is not so straightforward, as discussed in section (1.7). The basic difficulty comes from the *splitting method* and with the *stiff source term*. In another context of *Boltzmann schemes* (also termed as *Kinetic Schemes*), which use the concepts of Kinetic theory for developing upwind methods for Euler or Navier–Stokes equations of gas dynamics, Prendergast and Kun Xu [19] developed a Boltzmann scheme which is based on an integral solution of the Boltzmann equation with the B-G-K model for the collision term, without using operator splitting. This method was later improved and extended in various ways by Kun Xu [20, 21] and Kim *et al.* [22]. Using a similar strategy, Raghurama Rao and Khosla [23, 24] developed an *Unsplit Relaxation Scheme* for solving hyperbolic conservation laws. The basic advantage of this new *Relaxation Scheme* is that achieving second order accuracy is no longer as involved as in the *Relaxation Schemes* of Jin and Xin [8] and the problem of stiffness is overcome in a simple way for solving the conservation equations. The basic idea of the method is described in this section.

Consider the non-linear Burgers equation

$$\frac{\partial u}{\partial t} + \frac{\partial g\left(u\right)}{\partial x} = 0 \text{ where } g\left(u\right) = \frac{u^2}{2} \tag{128}$$

The *Relaxation system* for the above scalar conservation law is given by

$$\begin{aligned} \frac{\partial u}{\partial t} + \frac{\partial v}{\partial x} &= 0 \\ \frac{\partial v}{\partial t} + \lambda^2 \frac{\partial u}{\partial x} &= -\frac{1}{\epsilon}\left[v - g\left(u\right)\right] \end{aligned} \tag{129}$$

Let us now formulate a finite volume method for the first equation of (129) as

$$\overline{u}_j^{n+1} = \overline{u}_j^n - \frac{1}{\Delta x}\int_0^{\Delta t}\left(v_{j+\frac{1}{2}} - v_{j-\frac{1}{2}}\right)dt \tag{130}$$

where $\overline{u}_j$ is the cell integral average of $u\left(x\right)$ in a finite volume $\left[j - \frac{1}{2}, j + \frac{1}{2}\right]$. To use the above equation to update u, we require to calculate the fluxes $v_{j\pm\frac{1}{2}}$ at the cell interfaces. Since the variable v can be obtained from the distribution function $\mathbf{f}$ as $v = P\Lambda\mathbf{f}$, we can us use the integral solution of the following Discrete Boltzmann equation (which is nothing but the diagonal form of the *Relaxation System*(129)) to obtain $v_{j\pm\frac{1}{2}}$. The discrete Boltzmann equation is given by

$$\frac{\partial \mathbf{f}}{\partial t} + \Lambda\frac{\partial \mathbf{f}}{\partial x} = \frac{1}{\epsilon}\left[\mathbf{F} - \mathbf{f}\right] \tag{131}$$

where

$$\mathbf{f} = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} = \begin{bmatrix} \dfrac{u}{2} - \dfrac{v}{2\lambda} \\ \dfrac{u}{2} + \dfrac{v}{2\lambda} \end{bmatrix} \text{ and } \mathbf{F} = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix} = \begin{bmatrix} \dfrac{u}{2} - \dfrac{g\left(u\right)}{2\lambda} \\ \dfrac{u}{2} + \dfrac{g\left(u\right)}{2\lambda} \end{bmatrix} \tag{132}$$

If we can obtain $f_{j\pm\frac{1}{2}}$ at the cell interfaces from the solution of the above Discrete Boltzmann equation, we can use it to evaluate $v_{j\pm\frac{1}{2}}$ as the expression for recovering v from f is given by

$$v = P\Lambda\mathbf{f} = \lambda\left(f_2 - f_1\right) \tag{133}$$

First order accurate Unsplit Relaxation Scheme :
Consider a piece-wise constant approximation, as shown in the Figure 14. Let us consider the

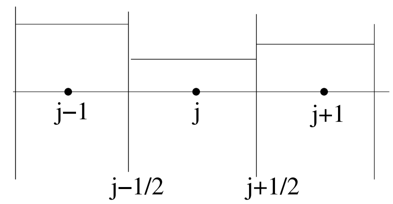

Figure 14: Piece-wise constant approximation

first equation of (131).

$$\frac{\partial f_1}{\partial t} - \lambda \frac{\partial f_1}{\partial x} = \frac{1}{\epsilon} \left[F_1 - f_1 \right] \tag{134}$$

The integral solution for the above equation, at the cell interface $x_{j+\frac{1}{2}}$, can be written as

$$f_1 \left(x_{j+\frac{1}{2}}, t \right) = \frac{1}{\epsilon} \int_0^t F_1 \left(x_{j+\frac{1}{2}} + \lambda \left(t - t' \right), t' \right) e^{-\frac{t-t'}{\epsilon}} dt' + f_1 \left(x_{j+\frac{1}{2}} + \lambda t, 0 \right) e^{-\frac{t}{\epsilon}} \tag{135}$$

Similarly, the second equation (131) is

$$\frac{\partial f_2}{\partial t} + \lambda \frac{\partial f_2}{\partial x} = \frac{1}{\epsilon} \left[F_2 - f_2 \right] \tag{136}$$

and its integral solution is

$$f_2 \left(x_{j+\frac{1}{2}}, t \right) = \frac{1}{\epsilon} \int_0^t F_2 \left(x_{j+\frac{1}{2}} - \lambda \left(t - t' \right), t' \right) e^{-\frac{t-t'}{\epsilon}} dt' + f_2 \left(x_{j+\frac{1}{2}} - \lambda t, 0 \right) e^{-\frac{t}{\epsilon}} \tag{137}$$

In the expression (135), there are two quantities which need to be approximated to obtain the complete solution, namely, the initial non-equilibrium distribution at the cell interface, $f_1 \left(x_{j+\frac{1}{2}} + \lambda t, 0 \right)$ and the equilibrium distribution at the cell interface, $F_1 \left(x_{j+\frac{1}{2}} + \lambda \left(t - t' \right), t' \right)$. The equilibrium distribution can be assumed to be smoother than the non-equilibrium distribution and, therefore, can be expanded in Taylor series at the cell interfaces as

$$F_1 \left(x, t \right) = F_{1,j+\frac{1}{2}} \left(1 + \overline{a} x + \overline{A} t \right) \tag{138}$$

For obtaining a first order accurate scheme, it is sufficient to retain only the first term and write

$$F_1 \left(x, t \right) = F_{1,j+\frac{1}{2}} \tag{139}$$

Similarly,

$$F_2 \left(x, t \right) = F_{2,j+\frac{1}{2}} \tag{140}$$

Substituting the above expressions in the integral solution (135) and (137), we obtain

$$f_1 \left(x_{j+\frac{1}{2}}, t \right) = F_1 \left(x_{j+\frac{1}{2}} \right) \left[1 - e^{-\frac{t}{\epsilon}} \right] + f_1 \left(x_{j+\frac{1}{2}} \right) e^{-\frac{t}{\epsilon}} \tag{141}$$

and

$$f_2\left(x_{j+\frac{1}{2}},t\right) = F_2\left(x_{j+\frac{1}{2}}\right)\left[1-e^{-\frac{t}{\epsilon}}\right] + f_2\left(x_{j+\frac{1}{2}}\right)e^{-\frac{t}{\epsilon}} \tag{142}$$

For approximating the initial non-equilibrium distribution at the cell interface, the possibilities of discontinuities is allowed and approximation is done by upwinding as follows.

$$f_1\left(x_{j+\frac{1}{2}}+\lambda t,0\right) = F_{1,j+1}\left(1+a_r x\right)\ \left(\text{for } x > x_{j+\frac{1}{2}}\right) \tag{143}$$

since the sign of λ is negative and the information at the cell interface comes from the right side. Similarly,

$$f_2\left(x_{j+\frac{1}{2}}-\lambda t,0\right) = F_{2,j}\left(1+a_l x\right)\ \left(\text{for } x < x_{j+\frac{1}{2}}\right) \tag{144}$$

since the sign of λ is positive and the information at the cell interface comes from the left side. For first order accuracy, we can retain only the first terms on the right hand side of the above two equations to obtain

$$f_1\left(x_{j+\frac{1}{2}}+\lambda t,0\right) = F_{1,j+1}\ \left(\text{for } x > x_{j+\frac{1}{2}}\right) \tag{145}$$

and

$$f_2\left(x_{j+\frac{1}{2}}-\lambda t,0\right) = F_{2,j}\ \left(\text{for } x < x_{j+\frac{1}{2}}\right) \tag{146}$$

The equilibrium distribution is defined by

$$\mathbf{F}_{j+\frac{1}{2}} = \begin{bmatrix} F_{1,j+\frac{1}{2}} \\ F_{2,j+\frac{1}{2}} \end{bmatrix} = \begin{bmatrix} \dfrac{1}{2}u_{j+\frac{1}{2}} - \dfrac{1}{2\lambda}g\left(u_{j+\frac{1}{2}}\right) \\ \\ \dfrac{1}{2}u_{j+\frac{1}{2}} + \dfrac{1}{2\lambda}g\left(u_{j+\frac{1}{2}}\right) \end{bmatrix} \tag{147}$$

By definition,

$$u_{j+\frac{1}{2}} = P\mathbf{f}_{j+\frac{1}{2}} = f_{1,j+\frac{1}{2}} + f_{2,j+\frac{1}{2}} \tag{148}$$

Since we have already approximated $\mathbf{f}$ in (145) and (146), we can use them in the above expressions to obtain $\mathbf{F}_{j+\frac{1}{2}}$. Having approximated both the initial non-equilibrium and equilibrium distributions at the interface, we can now substitute these expressions in the solutions of the integral equations (141) and (142), which, in turn, can be used in (130) and (133).

This method does not depend on splitting, and therefore, does not have the difficulties of the splitting schemes in achieving second order accuracy. Similar to the strategy used by Kun Xu [20] for the relaxation time in the BGK scheme, the following expression is used for ϵ.

$$\epsilon = C_1\Delta t + \Delta t min\left(1,C\right) \tag{149}$$

where

$$C = C_2\frac{u_R - u_L}{u_R + u_L} \tag{150}$$

where the subscripts L and R represent the cells to the left and right side of a surface. Here, C_1 and C_2 are constants. The best ranges for these constants, fixed by experimentation, are

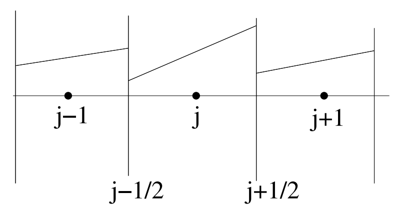

Figure 15: Piece-wise linear approximation

0.5 to 2.0 for C_1 and 0.5 to 1.0 for C_2.

Second order accurate Unsplit relaxation Scheme

For obtaining a second order accurate Unsplit Relaxation Scheme, the slopes a_r and a_l in expressions (145) and (146) are retained and evaluated using slope limiters to suppress spurious oscillations. Similarly, the $\overline{a}$ and $\overline{A}$ in (138) are retained and evaluated using the slopes with limiters together with the constraint that the moments of the non-equilibrium and equilibrium distributions should lead to the same conserved quantities, *i.e.*, $P\mathbf{f} = P\mathbf{F}$. For full details, the reader is referred to [23] and [24]. The expressions for the slopes are given by

$$a_l = \frac{1}{u_j} L\left(\Delta u_{j+\frac{1}{2}}, \Delta u_{j-\frac{1}{2}}\right) \ , \ a_r = \frac{1}{u_{j+1}} L\left(\Delta u_{j+\frac{3}{2}}, \Delta u_{j+\frac{1}{2}}\right) \text{ and } \overline{a} = \frac{1}{u_{j+\frac{1}{2}}} L\left(\Delta u_{j+\frac{3}{2}}, \Delta u_{j-\frac{1}{2}}\right) \tag{151}$$

where $\Delta u_{j+\frac{1}{2}} = u_{j+1} - u_j$ and L is any limiter. In this work, a minmod limiter is used. $\overline{A}$ is determined from the constraint $P\mathbf{f} = P\mathbf{F}$ [23, 24]. The results obtained by this Unsplit Relaxation Scheme for 2-D Burgers equation and Euler equations in 1-D and 2-D are shown in the Figures 16, 17 and 18.

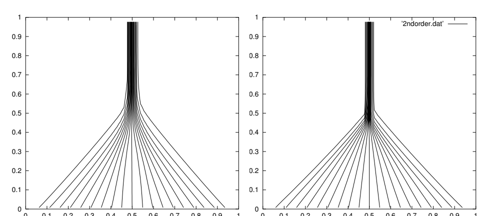

Figure 16: Results with first order accurate and second order accurate Unsplit Relaxation Scheme for the 2-D Burgers equation

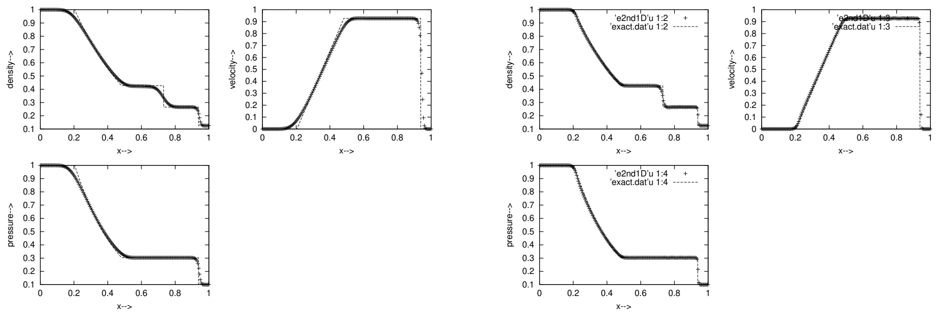

Figure 17: Results with first order accurate and second order accurate Unsplit Relaxation Scheme for the 1-D Euler equations (shock tube test case)

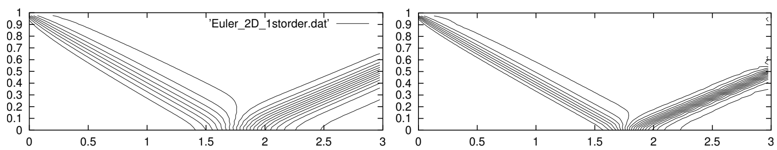

Figure 18: Results with the first order accurate and second order accurate Unsplit Relaxation Scheme for 2-D Euler equations for the shock reflection problem on a 60x20 grid

2.3 Method of Interpolation with a Relaxation Approximation for Conservation Laws (MIRACL)

In this section, *a simple algorithm for solving conservation laws is presented, based on the exact solution of the discrete Boltzmann equation with a suitable interpolation*. Such a simple framework is possible because of the availability of the *Relaxation Approximation*. Consider a non-linear conservation law

$$\frac{\partial u}{\partial t} + \frac{\partial g\left(u\right)}{\partial x} = 0 \text{ where } g\left(u\right) = \frac{1}{2}u^2 \tag{152}$$

The discrete Boltzmann equation as a *Relaxation Approximation* for the above non-linear conservation law is given by

$$\frac{\partial \mathbf{f}}{\partial t} + \Lambda \frac{\partial \mathbf{f}}{\partial x} = \frac{1}{\epsilon}\left[\mathbf{F} - \mathbf{f}\right] \tag{153}$$

Let us use the splitting method and separate the convection and relaxation parts as

$$\text{Relaxation Step } : \ \frac{d\mathbf{f}}{dt} = \frac{1}{\epsilon}\left[\mathbf{F} - \mathbf{f}\right] \tag{154}$$

$$\text{and Convection Step } : \ \frac{\partial \mathbf{f}}{\partial t} + \Lambda \frac{\partial \mathbf{f}}{\partial x} = 0 \tag{155}$$

The relaxation step can be solved by a simple implicit method as

$$\frac{\mathbf{f}_j^{n+1} - \mathbf{f}_j^n}{\Delta t} = \frac{1}{\epsilon}\left[\mathbf{F}^{n+1} - \mathbf{f}^{n+1}\right] \tag{156}$$

Since the equilibrium distribution $\mathbf{F}$ is a function of $\mathbf{f}$ (being a function of u and $g(u)$), at first glance, non-linear iterative solvers seem inevitable. However, using the principle that the moments of both the distribution $\mathbf{f}$ and the Maxwellian $\mathbf{F}$ yield the same conservative variable u as

$$u = P\mathbf{f} = P\mathbf{F} \tag{157}$$

the above equation yields $u^{n+1} = u^n$. Thus, the conservative variable remains unchanged during the relaxation step. Therefore, we can write

$$\mathbf{F}^{n+1} = \mathbf{F}^n \tag{158}$$

and the relaxation step now can be rewritten as an explicit expression for $\mathbf{f}^{n+1}$. Therefore, no non-linear iterative solvers are required [14]. The rearranged relaxation step is given by

$$\mathbf{f}^{n+1} = \frac{1}{\left[1 + \frac{\Delta t}{\epsilon}\right]}\left[\mathbf{f}^n - \frac{\Delta t}{\epsilon}\mathbf{F}^n\right] \tag{159}$$

The convection step is solved by using the exact solution of the discrete Boltzmann equation (without the relaxation term) with a suitable interpolation. For second order accuracy, Jin's modified Strang splitting procedure is used [14]. The details of solving the Discrete Boltzmann equation without the relaxation term is given in the following subsection. This scheme is named as *Method of Interpolation with Relaxation Approximation for Conservation Laws (MIRACL).*

2.3.1 Solving the convection equation using characteristics and interpolation

Consider the linear convection equation

$$\frac{\partial f_1}{\partial t} - \lambda\frac{\partial f_1}{\partial x} = 0 \tag{160}$$

The exact solution of the above equation is given by

$$f_1(x, t + \Delta t) = f_1(x + \lambda\Delta t, t) \tag{161}$$

Consider a 3-point stencil as shown below. The foot of the characteristic (point P) falls between the points j and $j+1$ on the 3-point stencil. Since the values of the variable f_1 are available only at the grid points, we can obtain the value at P by interpolating from the values of the variable f_1 at the grid points. A linear interpolation between x_j and x_{j+1} gives

$$f_{1,j}^{n+1} = f_{1,j}^n\frac{x_j + \lambda\Delta t - x_{j+1}}{x_j - x_{j+1}} + f_{1,j+1}^n\frac{x_j - x_j + \lambda\Delta t}{x_j - x_{j+1}} \tag{162}$$

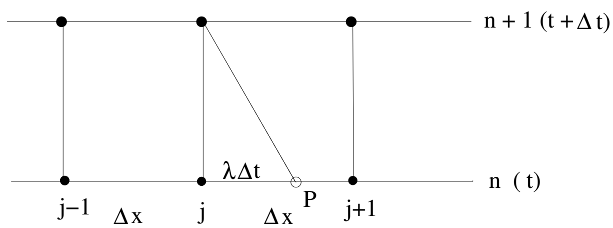

Figure 19: 3-point stencil

$$\text{or } f_{1,j}^{n+1} = f_{1,j}^{n} + \frac{\lambda \Delta t}{\Delta x} \left(f_{1,j+1} - f_{1,j} \right) \tag{163}$$

Therefore, we recover the first order upwind method when we use the exact solution for the linear convection equation with linear interpolation. Similarly, when we use the exact solution with a quadratic interpolation on the 3-point stencil, we can recover the Lax-Wendroff scheme. The exact solution of the linear convection equation with quadratic interpolation on an upwind stencil, with two points on the left or two points on the right of the point P, will lead to the Beam-Warming second order upwinding method in 1-D. In two dimensions, the exact solution with linear interpolation will lead to the first order upwind method; the exact solution with quadratic interpolation on a symmetric and compact 9-point stencil will lead to the 2-step Lax-Wendroff method. If we use an upwind stencil for quadratic interpolation with exact solution, we recover a totally new scheme. Thus, this framework of tracing the foot of the characteristics coupled with a suitable interpolation is a general framework from which some of the existing schemes for linear convection equation can be recovered as special cases. Another interesting feature with this approach is that *no discretization of the derivatives is required* and this makes the approach different from the traditional Finite Difference, Finite Volume and Finite Element methods. Eventhough this approach was known earlier, this could not be used to develop numerical methods for non-linear vector conservation laws because the exact solutions were not available except in some simple cases. With the *Relaxation Approximation*, it is now possible to use the above framework, as the non-linear vector conservation laws are converted to linear scalar convection equations with source terms (which can be separated by the splitting method). Another advantage of this approach is that the resulting numerical methods are *genuinely multi-dimensional*, as information coming from all neighbouring points is taken care of and no dimensional splitting is assumed. This new method (MIRACL) is tested on scalar conservation laws in 1-D and 2-D and the results are encouraging [25]. Some results are shown in Figures 20 and 21. From the figure 20, we can see that the MIRACL captures the oblique shock more crisply than an upwind relaxation scheme, as it is *genuinely multidimensional*. The results with the second order accurate MIRACL on a symmetric stencil are better than those on an asymmetric stencil. This method is currently being applied to vector conservation laws[26].

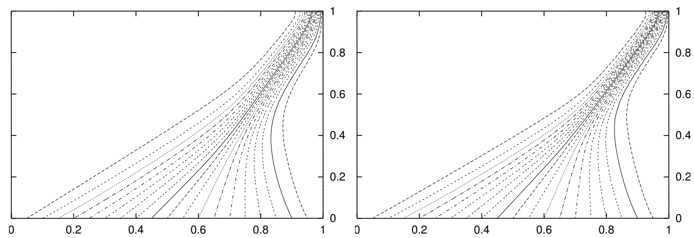

Figure 20: Comparison of the first order accurate upwind relaxation scheme and the first order accurate MIRACL for 2-D Burgers equation

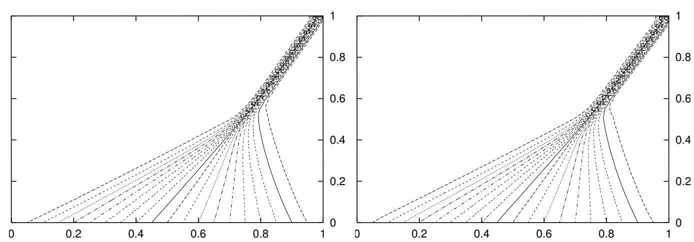

Figure 21: Results with second order accurate MIRACL on symmetric and upwind stencils for 2-D Burgers equation

2.4 A Kinetic Relaxation Scheme

In the Relaxation Scheme proposed by Jin and Xin, achieving second order accuracy is not straightforward and is rather complicated, as shown in section (1.7), due the presence of the stiff relaxation parameter in the source term. If our interest is only to solve the Euler or Navier-Stokes equations, it is not essential to retain the stiff parameter and a simple relaxed scheme can be obtained which is practically as good as a second order accurate method, though not formally. Such a method is presented in this section. This method resembles the *Kinetic Schemes* in spirit and, therefore, is termed as a *Kinetic Relaxation Scheme.*

Consider the non-linear Burgers equation

$$\frac{\partial u}{\partial t} + \frac{\partial g(u)}{\partial x} = 0 \text{ where } g(u) = \frac{u^2}{2} \tag{164}$$

The Relaxation system for the above scalar conservation law is given by

$$\begin{aligned} \frac{\partial u}{\partial t} + \frac{\partial v}{\partial x} &= 0 \\ \frac{\partial v}{\partial t} + \lambda^2 \frac{\partial u}{\partial x} &= -\frac{1}{\epsilon}\left[v - g(u)\right] \end{aligned} \tag{165}$$

The finite volume method applied to the above first equation of the *Relaxation System* leads to

$$\overline{u}_j^{n+1} = \overline{u}_j^n - \frac{1}{\Delta x}\int_0^{\Delta t}\left(v_{j+\frac{1}{2}} - v_{j-\frac{1}{2}}\right) dt \tag{166}$$

where $\overline{u}_j$ is the cell integral average of $u(x)$ in a finite volume $\left[j-\frac{1}{2}, j+\frac{1}{2}\right]$. To use the above equation to update u, we require to calculate the fluxes $v_{j\pm\frac{1}{2}}$ at the cell interfaces. Since $v_{j+\frac{1}{2}} = P\Lambda \mathbf{f}_{\mathbf{j}+\frac{\mathbf{1}}{\mathbf{2}}}$, we can use the Discrete Boltzmann equation to obtain $f_{j+\frac{1}{2}}$ and use it to obtain the fluxes $v_{j\pm\frac{1}{2}}$. The Discrete Boltzmann equation is given by

$$\frac{\partial \mathbf{f}}{\partial t} + \Lambda \frac{\partial \mathbf{f}}{\partial x} = \frac{1}{\epsilon}\left[\mathbf{F} - \mathbf{f}\right] \tag{167}$$

where

$$\mathbf{f} = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} = \begin{bmatrix} \dfrac{u}{2} - \dfrac{v}{2\lambda} \\ \dfrac{u}{2} + \dfrac{v}{2\lambda} \end{bmatrix} \text{ and } \mathbf{F} = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix} = \begin{bmatrix} \dfrac{u}{2} - \dfrac{g(u)}{2\lambda} \\ \dfrac{u}{2} + \dfrac{g(u)}{2\lambda} \end{bmatrix} \tag{168}$$

If we can obtain $f_{j\pm\frac{1}{2}}$ at the cell interfaces from the solution of the above Discrete Boltzmann equation, we can use it to evaluate $v_{j\pm\frac{1}{2}}$ as the expression for recovering v from f is given by

$$v = P\Lambda \mathbf{f} = \lambda\left(f_2 - f_1\right) \tag{169}$$

Let us consider the first equation of (167).

$$\frac{\partial f_1}{\partial t} - \lambda \frac{\partial f_1}{\partial x} = \frac{1}{\epsilon}\left[F_1 - f_1\right] \tag{170}$$

Using a splitting method, we can split the above into two steps :

$$\begin{aligned} \text{Relaxation Step} &: \quad \frac{df_1}{dt} = \frac{1}{\epsilon}\left[F_1 - f_1\right] \\ \text{Convection Step} &: \quad \frac{\partial f_1}{\partial t} + \Lambda \frac{\partial f_1}{\partial x} = 0 \end{aligned} \tag{171}$$

For the relaxation step, instead of the usual limit $\epsilon \to 0$, we can force $\epsilon = 0$ if our interest is only in solving Euler or Navier–Stokes equations. In such a case the relaxation parameter does not correspond to any physical process and the stiffness introduced by the mathematical formulation can be relaxed. The solution of the relaxation step is given by

$$f_1 = \left[f_1(t=0) - F_1\right] e^{-\frac{t}{\epsilon}} + F_1 \tag{172}$$

and with $\epsilon = 0$, we obtain

$$f_1 = F_1 \tag{173}$$

Similarly, for the second component, f_2, we obtain, for $\epsilon = 0$,

$$f_2 = F_2 \tag{174}$$

Therefore, we have a very simple algorithm of two steps :

$$\begin{aligned} \text{Relaxation Step} &: \quad \mathbf{f} = \mathbf{F} \\ \text{Convection Step} &: \quad \frac{\partial \mathbf{f}}{\partial t} + \Lambda \frac{\mathbf{f}}{\partial x} = 0 \end{aligned} \tag{175}$$

Thus, the discrete distribution function $\mathbf{f}$ is relaxed to the equilibrium distribution or the Maxwellian at the end of every convection time-step instantaneously. This is analogous to the philosophy of *Kinetic Schemes*. Therefore, we just have to solve linear convection equations, apart from forcing the discrete distributions to Maxwellians at the end of every time-step. Note that the relaxation parameter has disappeared and correspondingly all the stiffness related problems are also removed. But, the method is formally first order accurate, even if we solve the convection equation with second order accuracy in time and space. Leveque [27] has shown in another context that the splitting method, eventhough formally only first order accurate in time, is practically as good as a Strang splitting method in achieving second order accuracy in time. For the sake of emphasizing this point in the context of the new *Relaxation Scheme* presented in this section, Leveque's argument [27] is repeated in the following paragraphs.

Consider an unsplit method, which can be written for general operators A and B as

$$\frac{\partial f}{\partial t} = (A + B) f \tag{176}$$

In the case of the Boltzmann equation, A can be the relaxation (or collision) operator and B can be the convection operator. Assuming that the operators A and B do not explicitly depend on t (for simplicity), we can write

$$\frac{\partial^2 f}{\partial t^2} = (A + B) \frac{\partial f}{\partial t} = (A + B)^2 f \tag{177}$$

Proceeding in a similar way, we can obtain

$$\frac{\partial^n f}{\partial t^n} = (A + B)^n f \tag{178}$$

Using Taylor series expansion, we can write

$$\begin{aligned} f(x, t + \Delta t) &= f(x, t) + \Delta t \left(\frac{\partial f}{\partial t} \right) + \frac{\Delta t^2}{2} \left(\frac{\partial f}{\partial t} \right)^2 + \dots \\ &= f(x, t) + \Delta t (A + B) f(x, t) + \frac{\Delta t^2}{2} (A + B)^2 f(x, t) + \dots \\ &= \left(I + \Delta t (A + B) + \frac{\Delta t^2}{2} (A + B)^2 + \dots \right) f(x, t) \end{aligned} \tag{179}$$

which could be written in a compact notation as

$$f(x, t + \Delta t) = e^{\Delta t (A + B)} f(x, t) \tag{180}$$

Now, consider the splitting method which is given by

$$\frac{\partial f}{\partial t} = Af \text{ and } \frac{\partial f}{\partial t} = Bf \tag{181}$$

Using Taylor series expansion, we can write

$$\begin{aligned}
f^{(A)}(x, t+\Delta t) &= e^{\Delta t A} f(x,t) \\
f^{(A+B)}(x, t+\Delta t) &= e^{\Delta t B} f^{(A)}(x, t+\Delta t) = e^{\Delta t B} e^{\Delta t A} f(x,t) \\
&= \left(I + \Delta t B + \frac{\Delta t^2}{2} B^2 + \ldots \right) \left(I + \Delta t A + \frac{\Delta t^2}{2} A^2 + \ldots \right) f(x,t) \\
&= \left(I + \Delta t (A+B) + \frac{\Delta t^2}{2} \left(A^2 + 2BA + B^2 \right) + \ldots \right) f(x,t)
\end{aligned} \tag{182}$$

The splitting error can be obtained by subtracting the above final solution obtained by the splitting method (182) from the solution obtained from an unsplit method (179) as

$$\begin{aligned}
\text{Splitting error} &= f(x, t+\Delta t) - f^{(A+B)}(x, t+\Delta t) \\
&= e^{\Delta t (A+B)} f(x,t) - e^{\Delta t B} e^{\Delta t A} f(x,t) \\
&= \left(\frac{\Delta t^2}{2} \left[(A+B)^2 - \left(A^2 + 2BA + B^2 \right) \right] \right) f(x,t) + O\left[\Delta t^3\right] \\
&= \left(\frac{\Delta t^2}{2} \left[AB - BA \right] \right) f(x,t) + O\left[\Delta t^3\right]
\end{aligned} \tag{183}$$

Therefore, for the splitting error to be $O[\Delta t^3]$, the operators A and B should commute, *i.e.*, $AB = BA$. If the operators do not commute, as is often the case, the splitting error makes the method formally first order accurate in time, no matter how accurately we solve the step A and step B. The remedy for this defect is suggested by Strang [28]. The Strang splitting is given by 3 steps : (i) first operator (A) for half a time-step, (ii) second operator (B) for full time-step, followed by (iii) first operator (A) again for half a time-step.

$$\begin{aligned}
f^{Strang}(x, t+\Delta t) &= e^{\frac{\Delta t}{2} A} e^{\Delta t B} e^{\frac{\Delta t}{2} A} f(x,t) \\
&= \left(I + \left(\frac{\Delta t}{2} A \right) + \frac{1}{2} \left(\frac{\Delta t}{2} A \right)^2 + \ldots \right) \left(I + \Delta t B + \frac{1}{2} \left(\Delta t B \right)^2 + \ldots \right) \\
&\quad \left(I + \left(\frac{\Delta t}{2} A \right) + \frac{1}{2} \left(\frac{\Delta t}{2} A \right)^2 + \ldots \right) f(x,t) \\
&= \left(I + \Delta t (A+B) + \frac{\Delta t^2}{2} \left(A^2 + AB + BA + B^2 \right) \right) f(x,t) + O\left[\Delta t^3\right]
\end{aligned} \tag{184}$$

The splitting error for the Strang splitting is given by

$$\begin{aligned}
\text{Splitting error} &= f(x, t+\Delta t) - f^{Strang}(x, t+\Delta t) \\
&= e^{\Delta t (A+B)} f(x,t) - e^{\frac{\Delta t}{2} A} e^{\Delta t B} e^{\frac{\Delta t}{2} B} f(x,t)
\end{aligned} \tag{185}$$

Using the expressions (179) and (184), we can see that

$$\text{Splitting error} = f(x, t + \Delta t) - f^{Strang}(x, t + \Delta t) = O\left[\Delta t^3\right] \tag{186}$$

and hence the Strang splitting is formally second order accurate in time. Let us now see how the splitting method fares in comparison with the Strang splitting method. After n time-steps, the expression for the solution with the Strang splitting is given by

$$f(x, t + n\Delta t) = e^{\frac{\Delta t}{2}A} e^{\Delta t B} e^{\frac{\Delta t}{2}A} \;\; e^{\frac{\Delta t}{2}A} e^{\Delta t B} e^{\frac{\Delta t}{2}A} \;\; \ldots e^{\frac{\Delta t}{2}A} e^{\Delta t B} e^{\frac{\Delta t}{2}A} \; f(x, t) \tag{187}$$

We can combine the half time-step operator A in the end of one iteration with the half time-step operator A in the beginning of the next iteration as

$$e^{\frac{\Delta t}{2}A} e^{\frac{\Delta t}{2}A} = e^{\Delta t A} \tag{188}$$

Therefore, the Strang splitting expression becomes

$$f(x, t + n\Delta t) = e^{\frac{\Delta t}{2}A} e^{\Delta t B} e^{\Delta t A} e^{\Delta t B} e^{\Delta t A} \ldots e^{\Delta t B} e^{\frac{\Delta t}{2}A} f(x, t) \tag{189}$$

Comparing this with the expression for the splitting method given by

$$f(x, t + n\Delta t) = e^{\Delta t A} e^{\Delta t B} e^{\Delta t A} e^{\Delta t B} \ldots e^{\Delta t A} e^{\Delta t B} f(x, t) \tag{190}$$

we can see that the Strang splitting method differs from the ordinary splitting method only in the first and last terms, that is, only in the fact that the method starts and ends with a half time-step of operator A, rather than starting and ending with a full time-step and ending with B. Thus, eventhough the ordinary splitting method is formally only first order accurate, in reality, the solution will be as good as a second order accurate solution in time, except for the fact that the solution is evaluated at a slightly wrong time, approximately off by a time-step [27]. The new *Relaxation Scheme* developed in this section uses just the ordinary splitting method and is thus simpler.

The Kinetic Relaxation Scheme thus consists of just two linear convection equations to be solved

$$\frac{\partial f_1}{\partial t} - \lambda \frac{\partial f_1}{\partial x} = 0 \text{ and } \frac{\partial f_2}{\partial t} + \lambda \frac{\partial f_2}{\partial x} = 0 \tag{191}$$

with the constraints

$$f_1^n = F_1^n \text{ and } f_2^n = F_2^n \tag{192}$$

at the beginning of every time-step. Let us analyse the first order accurate Kinetic Relaxation Scheme further. Using a simple upwind method on a 3-point stencil, we can write

$$f_{1,j}^{n+1} = f_{1,j}^n + \frac{\lambda \Delta t}{\Delta x}\left[f_{1,j+1}^n - f_{1,j}^n\right] \tag{193}$$

Using the constraint that the distribution function is relaxed to an equilibrium distribution at the beginning of the time-step, we obtain

$$f_{1,j}^{n+1} = F_{1,j}^{n} + \frac{\lambda \Delta t}{\Delta x}\left[F_{1,j+1}^{n} - F_{1,j}^{n}\right] \tag{194}$$

Similarly, for the second component, we get

$$f_{2,j}^{n+1} = F_{2,j}^{n} - \frac{\lambda \Delta t}{\Delta x}\left[F_{2,j}^{n} - F_{2,j-1}^{n}\right] \tag{195}$$

Using the definition $u = P\mathbf{f} = f_1 + f_2$, we can write

$$u_j^{n+1} = f_{1,j}^{n+1} + f_{2,j}^{n+1} = F_{1,j}^{n} + \frac{\lambda \Delta t}{\Delta x}\left[F_{1,j+1}^{n} - F_{1,j}^{n}\right] + F_{2,j}^{n} - \frac{\lambda \Delta t}{\Delta x}\left[F_{2,j}^{n} - F_{2,j-1}^{n}\right] \tag{196}$$

or

$$u_j^{n+1} = \left(F_{1,j}^{n} + F_{2,j}^{n}\right) - \frac{\Delta t}{\Delta x}\left[\lambda\left(-F_{1,j+1}^{n} + F_{1,j}^{n} + F_{2,j}^{n} - F_{2,j-1}^{n}\right)\right] \tag{197}$$

or

$$u_j^{n+1} = u_j^{n} - \frac{\Delta t}{\Delta x}\left[\lambda\left(-F_{1,j+1}^{n} + F_{1,j}^{n} + F_{2,j}^{n} - F_{2,j-1}^{n}\right)\right] \tag{198}$$

Using the definitions $F_1 = \frac{u}{2} - \frac{g(u)}{2\lambda}$ and $F_2 = \frac{u}{2} + \frac{g(u)}{2\lambda}$ we obtain

$$u_j^{n+1} = u_j^{n} - \frac{\Delta t}{\Delta x}\left[\lambda\left(-\frac{u_{j+1}^{n}}{2} + \frac{g\left(u_{j+1}^{n}\right)}{2\lambda} + \frac{u_j^n}{2} - \frac{g\left(u_j^n\right)}{2\lambda} + \frac{u_j^n}{2} + \frac{g\left(u_j^n\right)}{2\lambda} - \frac{u_{j-1}^{n}}{2} - \frac{g\left(u_{j-1}^{n}\right)}{2\lambda}\right)\right] \tag{199}$$

which, after rearrangement, becomes

$$u_j^{n+1} = u_j^{n} + \frac{\lambda \Delta t}{2\Delta x}\left[u_{j+1}^{n} - 2u_j^n + u_{j-1}^{n}\right] - \frac{\Delta t}{2\Delta x}\left[g\left(u_{j+1}^{n}\right) - g\left(u_{j-1}^{n}\right)\right] \tag{200}$$

Thus, the Kinetic Relaxation Scheme reduces to the Relaxed Scheme of Jin and Xin [8]. The scheme so obtained is simple and is faster as the number of variables is reduced compared to the relaxation schemes. But, it is preferable to work at the level of the discrete Boltzmann equation, as the *Relaxaion System* is diagonalizable and all the research experience gained from the Kinetic Schemes can be useful in this framework for further research. This method seems interesting and appears promising for further study. The second order accurate scheme can be derived in a similar manner by considering a piece-wise linear approximation.

2.5 A Relaxation Scheme for Parabolic Equations

The extension of *Relaxation Schemes* to to viscous flows is not straightforward, as the simple minded approach of formulating a *Relaxation System* for parabolic equations by matching the viscous terms obtained by Chapman–Enskog type expansion with the actual viscous terms results in a *Relaxation System* with the stiff relaxation parameter in the convection terms also, apart from being in the source term. Jin, Pareschi and Toscani [30] formulated a *Relaxation Scheme* for parabolic equations by overcoming this problem with some readjustments in the *Relaxation System*. In this section, another approach is presented for developing a *Relaxation Scheme* for parabolic equations. The analysis presented in this section is based on the analysis presented by Arora [29].

Consider a viscous Burgers in equation in 1-D, given by

$$\frac{\partial u}{\partial t} + \frac{\partial g_i\left(u\right)}{\partial x} = \nu \frac{\partial^2 u}{\partial x^2} \tag{201}$$

where $g_i(u) = \dfrac{1}{2}u^2$ is the inviscid flux. We can rewrite the above equation (201) as

$$\frac{\partial u}{\partial t} + \frac{\partial g}{\partial x} = 0 \text{ where } g = g_i - \nu \frac{\partial u}{\partial x} \tag{202}$$

Relaxation approximation for (202) can be written as

$$\frac{\partial u}{\partial t} + \frac{\partial v}{\partial x} = 0 \tag{203}$$

$$\frac{\partial v}{\partial t} = -\frac{1}{\epsilon}\left(v - v_{eq}\right) \tag{204}$$

where $v_{eq} = g$. Rewriting, we obtain

$$\frac{\partial u}{\partial t} + \frac{\partial v}{\partial x} = 0 \tag{205}$$

$$\frac{\partial v}{\partial t} + \frac{\nu}{\epsilon}\left(\frac{\partial u}{\partial x}\right) = -\frac{1}{\epsilon}\left(v - g_i\right) \tag{206}$$

Let us replace $\dfrac{\nu}{\epsilon}\left(\dfrac{\partial u}{\partial x}\right)$ in above equation by $\dfrac{\partial w(u)}{\partial x}$ to obtain

$$\frac{\partial v}{\partial t} + \frac{\partial w(u)}{\partial x} = -\frac{1}{\epsilon}\left(v - g_i\right) \tag{207}$$

Let us now do a Chapman–Enskog type expansion of the Relaxation system (205), (207) and compare the so obtained viscous term with the actual viscous term in the Burgers equation. Let us use the substitutions $x = \overline{x}\epsilon$ and $t = \overline{t}\epsilon$ in (205) and (207) to eliminate ϵ temporarily.

$$\frac{\partial u}{\partial \overline{t}} + \frac{\partial v}{\partial \overline{x}} = 0 \tag{208}$$

$$\frac{\partial v}{\partial \overline{t}} + \frac{\partial (w\left(u\right))}{\partial \overline{x}} = -\left(v - g_i\right) \tag{209}$$

Let

$$v = g_i + \phi \tag{210}$$

where ϕ is small quantity and its derivatives are even smaller. Therefore, using (210), we obtain

$$\frac{\partial v}{\partial \bar{t}} = \frac{\partial g_i}{\partial \bar{t}} + \frac{\partial \phi}{\partial \bar{t}} \approx \frac{\partial g_i}{\partial \bar{t}} = \left(\frac{\partial g_i}{\partial u}\right)\left(\frac{\partial u}{\partial \bar{t}}\right) \tag{211}$$

$$\frac{\partial v}{\partial \overline{x}} = \frac{\partial g_i}{\partial \overline{x}} + \frac{\partial \phi}{\partial \overline{x}} \approx \frac{\partial g_i}{\partial \overline{x}} = \left(\frac{\partial g_i}{\partial u}\right)\left(\frac{\partial u}{\partial \overline{x}}\right) \tag{212}$$

From (210), we have $\phi = v - g_i$, and using (209), we can write:

$$\phi = -\frac{\partial v}{\partial \bar{t}} - \frac{\partial \left(w\left(u\right)\right)}{\partial \overline{x}} \tag{213}$$

Substituting (212) in (208), we get

$$\frac{\partial u}{\partial \bar{t}} = -\left(\frac{\partial g_i}{\partial u}\right)\left(\frac{\partial u}{\partial \overline{x}}\right) \tag{214}$$

From (214) and (211), we obtain

$$\frac{\partial v}{\partial \bar{t}} = \left(\frac{\partial g_i}{\partial u}\right)\left(-\frac{\partial g_i}{\partial u}\frac{\partial u}{\partial \overline{x}}\right) = -\left(\frac{\partial g_i}{\partial u}\right)^2 \frac{\partial u}{\partial \overline{x}} \tag{215}$$

Substituting (215) in (213) results in

$$\phi = \frac{\partial u}{\partial \overline{x}}\left[\left(\frac{\partial g_i}{\partial u}\right)^2 - \frac{\partial w(u)}{\partial u}\right] \tag{216}$$

Substituting (216) in (210), we get

$$v = g_i + \frac{\partial u}{\partial \overline{x}}\left[\left(\frac{\partial g_i}{\partial u}\right)^2 - \frac{\partial w(u)}{\partial u}\right] \tag{217}$$

Substituting (217) in (208), we get

$$\frac{\partial u}{\partial \bar{t}} + \frac{\partial g_i}{\partial \overline{x}} = \frac{\partial^2 u}{\partial \overline{x}^2}\left[\frac{\partial w(u)}{\partial u} - \left(\frac{\partial g_i}{\partial u}\right)^2\right] \tag{218}$$

Transforming $\overline{x}$, $\bar{t}$ back to x, t in (218),we get

$$\frac{\partial u}{\partial t} + \frac{\partial g_i}{\partial x} = \epsilon\frac{\partial^2 u}{\partial x^2}\left[\frac{\partial w(u)}{\partial u} - \left(\frac{\partial g_i}{\partial u}\right)^2\right] \tag{219}$$

Hence, the Chapman-Enskog type expansion of Relaxation system (205), (207) leads to a parabolic equation (219). For the *Relaxation System* (205), (207) to represent to viscous Burgers equation (201), we need to match the viscous term of the parabolic equation (219) to the vicous term in the Burgers equation (201) as

$$\nu \frac{\partial^2 u}{\partial x^2} = \epsilon \frac{\partial^2 u}{\partial x^2} \left[\frac{\partial w(u)}{\partial u} - \left(\frac{\partial g_i}{\partial u} \right)^2 \right] \tag{220}$$

Rearranging, (220) gives

$$\frac{\partial w}{\partial u} = \frac{\nu}{\epsilon} + \left(\frac{\partial g_i}{\partial u} \right)^2 = \frac{\nu}{\epsilon} + u^2 \tag{221}$$

Integrating (221) w.r.t. u, we get

$$w(u) = \int \left(\frac{\nu}{\epsilon} + u^2 \right) du = \left(\frac{\nu}{\epsilon} u + \frac{1}{3} u^3 \right) \tag{222}$$

Substituting (222) in (207), we obtain

$$\frac{\partial v}{\partial t} + \left(\frac{\nu}{\epsilon} + u^2 \right) \frac{\partial u}{\partial x} = -\frac{1}{\epsilon} (v - g_i) \tag{223}$$

Equations (205) and (223) together constitute the Relaxation System for (201). As we can see, the convection term also contains the stiff parameter ϵ, apart from the source term and this creates an additional difficulty. To overcome this problem, we can rearrange the above *Relaxation System* as

$$\frac{\partial u}{\partial t} + \frac{\partial v}{\partial x} = 0 \tag{224}$$

$$\frac{\partial v}{\partial t} + \underbrace{u^2}_{\left(\frac{\partial g_i}{\partial u} \right)^2} \frac{\partial u}{\partial x} = -\frac{1}{\epsilon} \left(v - \underbrace{g_i}_{Inviscid\ Flux} + \underbrace{\nu \frac{\partial u}{\partial x}}_{Viscous\ Flux} \right) \tag{225}$$

or (225) can be written as

$$\frac{\partial v}{\partial t} + \left(\frac{\partial g_i}{\partial u} \right)^2 \frac{\partial u}{\partial x} = -\frac{1}{\epsilon} (v - g) \tag{226}$$

Where $g = \left(g_i - \nu \dfrac{\partial u}{\partial x} \right)$. The equations (224) & (226) constitute a relaxation approximation for the viscous Burgers equation. The Relaxation system (224) and (226) has the same structure as that we obtained for a 1D hyperbolic equation. Using this modification of the flux, we can write this system in diagonalized form as

$$\begin{aligned}\frac{\partial f_1}{\partial t} - \lambda\frac{\partial f_1}{\partial x} &= \frac{1}{2\epsilon\lambda}\left\{v - g\left(u, \frac{\partial u}{\partial x}\right)\right\} \\ \frac{\partial f_2}{\partial t} + \lambda\frac{\partial f_2}{\partial x} &= -\frac{1}{2\epsilon\lambda}\left\{v - g\left(u, \frac{\partial u}{\partial x}\right)\right\}\end{aligned} \tag{227}$$

where f_1 and f_2 are such that

$$u = f_1 + f_2 \text{ and } v = \lambda\left(f_2 - f_1\right) \tag{228}$$

Solving the above two equations in the limit of $\epsilon \to 0$ will be equivalent to solving the 1D viscous burgers equation. This formulation is similar to the relaxation system for hyperbolic equations and we can use the same numerical techniques used to solve relaxation system for hyperbolic equations. The only difference will be in the definition of the local equilibrium, which will now involve the derivatives of the original variable u also, *i.e.*, $v = g\left(u, \frac{\partial u}{\partial x}\right)$. The decoupled *Relaxation System* can be represented as following Discrete Kinetic System

$$\begin{aligned}\frac{\partial f_1}{\partial t} - \lambda\frac{\partial f_1}{\partial x} &= \frac{1}{\epsilon}\left(F_1 - f_1\right) \\ \frac{\partial f_2}{\partial t} + \lambda\frac{\partial f_2}{\partial x} &= \frac{1}{\epsilon}\left(F_2 - f_2\right)\end{aligned} \tag{229}$$

or, equivalently,

$$\frac{\partial \mathbf{f}}{\partial t} + \Lambda\frac{\partial \mathbf{f}}{\partial x} = \frac{1}{\epsilon}\left(\mathbf{F} - \mathbf{f}\right) \tag{230}$$

where

$$\mathbf{f} = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}, \mathbf{F} = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix} \text{ and } \Lambda = \begin{bmatrix} -\lambda & 0 \\ 0 & \lambda \end{bmatrix} \tag{231}$$

Discrete local Maxwellian $\mathbf{F}$ (with components F_1 and F_2) in this case is defined by

$$F_1 = \frac{1}{2}u - \frac{1}{2\lambda}\left[g\left(u, \frac{\partial u}{\partial x}\right)\right] \text{ and } F_2 = \frac{1}{2}u + \frac{1}{2\lambda}\left[g\left(u, \frac{\partial u}{\partial x}\right)\right] \tag{232}$$

The motivation for this strategy came from the application of Kinetic Schemes for Navier–Stokes equations, in which a Chapman–Enskog distribution is used to derive Navier–Stokes equations from the Boltzmann equation [31]. The modified equilibrium distribution in this section is similar to the Chapman–Enskog distribution which is a perturbation over the Maxwellian and contains the shear stress tensor and heat flux vector terms as additions to the expression of the Maxwellian.

The first order and second order accurate versions of this *Relaxation Scheme* is applied to two test cases for the 1-D viscous Burgers equation and the results are shown in figures (22) and (23). The input parameters for the two test cases are as follows.

- Test Case 1 : $\Delta x = 0.05$, $\nu = 0.05$, $\Delta t = 0.0025$, $t_final = 4.0$ and $\epsilon = 1.0 \times 10^{-8}$.
- Test Case 2 : $\Delta x = 0.1$, $\nu = 0.2$, $\Delta t = 0.01$, $t_final = 8.0$ and $\epsilon = 1.0 \times 10^{-8}$.
- The initial conditions for both the test cases are :
  $u = 2.0$ for $-\infty \leq x \leq 0$ and $u = 1.0$ for $0 \leq x \leq \infty$.

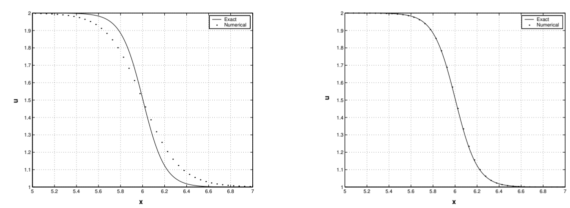

Figure 22: Results for the Viscous Burgers equation test case I

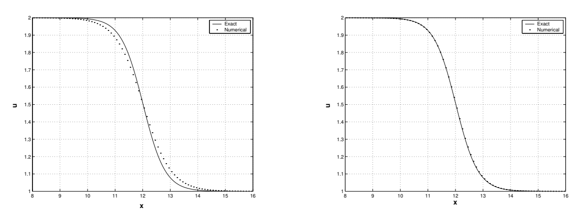

Figure 23: Results for the Viscous Burgers equation test case II

This method is currently being applied to Navier–Stokes equations [32].

3 Conclusions

The *Relaxation Schemes* present an interesting alternative to the existing upwind methods like Riemann solvers that often get into troubles and complicated flux splittings. They are simpler compared to the existing approaches and are attractive for further research as there is a lot of potential in this framework to be exploited. Some new *Relaxation Schemes* are presented in this report for solving hyperbolic and parabolic conservation laws numerically. One possible future research direction is to make these schemes more competitive by reducing the numerical dissipation in them.

Acknowledgments

The author is thankful to Prof. Helmut Neunzert for arranging a visiting position in May, 2002, to Dr. Michael Junk and Mr. G. Venkiteswaran for all the nice arrangements for stay and especially to Dr. Michael Junk for some iteresting discussions.

References

[1] C. Hirsch, Numerical Computation of Internal and External Flows, Vol. 1: Fundamentals of Numerical Discretization, John Wiley & Sons, 1988.

[2] C. Hirsch, Numerical Computation of Internal and External Flows, Vol. 2: Computational Methods for Inviscid and Viscous Flows, John Wiley & Sons, 1990.

[3] E.F. Toro, Riemann Solvers and Numerical Methods for Fluid Dynamics : A Practical Introduction, 2nd Edition, Springer, 1999.

[4] Culbert B. Laney, Computational Gasdynamics, Cambridge University Press, 1998.

[5] S.M. Deshpande, Kinetic Flux Splitting Schemes, Computational Fluid Dynamics Review 1995 : A state-of-the-art reference to the latest developments in CFD, (eds.) M.M. Hafez and K. Oshima, John Wiley & Sons, Chechester, 1995.

[6] E. Godlewski and P.A. Raviart, Numerical Approximations of hyperbolic Systems of Conservation Laws, Applied Mathematics Series 118, Springer–Verlag, 1996.

[7] Quirk, A contribution to the great Riemann solver debate, International Journal of Numerical Methods in Fluids, vol. 18, pp. 555-574, 1994.

[8] S. Jin and Z. Xin, The Relaxation Schemes for Systems of Conservation Laws in Arbitrary Space Dimensions, Communications in Pure and Applied Mathematics, vol. 48, pp. 235–276, 1995.

[9] R. Natalini, A Discrete Kinetic Approximation of Entropy Solutions to Multidimensional Scalar Conservation Laws, Journal of Differential Equations, vol. 148, pp. 292–317, 1999.

[10] D. Aregba-Droillet and R. Natalini, Discrete Kinetic Schemes for Multidimensional Systems of Conservation Laws, SIAM Journal of Numerical Analysis, vol. 37, no. 6, pp. 1973–2004, 2000.

[11] R. Natalini, Recent mathematical results on hyperbolic relaxation problems, Analysis of systems of conservation laws, Pitman Research Notes in Mathematics Series, Longman, Harlow, 1998.

[12] R.J. Leveque ad M. Pelanti, A Class of Approximate Riemann Solvers and Their Relation to Relaxation Schemes, Journal of Computational Physics, 172, pp. 572-59. 2001.

[13] S. Balasubramanyam, 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.

[14] S. Jin, Runge–Kutta Methods for Hyperbolic Conservation Laws with Stiff Relaxation Terms, Journal of Computational Physics, vol. 122, pp. 51–67, 1995.

[15] A.K. Ghosh and S.M. Deshpande, "Least Squares Kinetic Upwind Method for Inviscid Compressible Flows", AIAA paper, 95-1735, 1995.

[16] S. Balasubramanyam and S.V. Raghurama Rao, A New Grid Free Relaxation Scheme for Euler Equations, Fluid Mechanics Report No: 2001 FM 07, Dept. of Aerospace Engg., IISc, Bangalore.

[17] S. Balasubramanyam and S.V. Raghurama Rao, A New Grid Free Relaxation Scheme for Euler Equations, Proceedings of the Ninth Asian Congress of Fluid Mechanics, May 27-31 2002, Isfahan, Iran.

[18] S. Balasubramanyam and S.V. Raghurama Rao, A New Grid Free Relaxation Scheme for Conservation Laws, 2^{nd} International Conference on Computational Fluid Dynamics, July 2002 , Sydney, Australia (accepted for presentation and the full paper to be published by Springer).

[19] K.H. Prendergast and Kun Xu, Numerical Hydrodynamics from Gas-Kinetic Theory, Journal of Computational Physics, vol. 109, pp. 53–66, 1993.

[20] Kun Xu, Gas–Kinetic Schemes for Unsteady Compressible Flow Simulations, VKI Lecture Series on Computational Fluid Dynamics, 1998, von Karman Institute for Fluid Dynamics, Belgium.

[21] Kun Xu, A Gas–Kinetic BGK Scheme for the Navier–Stokes Equations and its Connection with Artificial Dissipation and Godunov Method, Journal of Computational Physics, vol. 179, pp 289-335, 2001.

[22] C. Kim, Kun Xu, L. Martinelli and A. Jameson, Analysis and Implementation of the Gas-Kinetic BGK Scheme for Computational Gas Dynamics, International Journal for Numerical Methods in Fluids, vol. 25, pp 21-49, 1997.

[23] S.V. Raghurama Rao and Sachin Khosla, A New Unsplit Relaxation Scheme for Hyperbolic Conservation Laws, Fluid Mechanics report, Department of Aerospace Engineering, Indian Institute of Science, Bangalore, India, 2002.

[24] S.V. Raghuramarao and Sachin Khosla, An Unsplit Relaxation Scheme for Hyperbolic Conservation Laws, 2^{nd} International Conference on Computational Fluid Dynamics, July 2002, Sydney, Australia (accepted for presentation and proceedings to be published by Springer).

[25] M.V. Subba Rao, A New Multidimensional Scheme for Solving Hyperbolic Equations, M.E. thesis, Department of Mechanical Engineering, Coimbatore Institute of Technology, Coimbatore, India, 2002.

[26] M.V. Subba Rao and S.V. Raghurama Rao, Method of Interpolation with a Relaxation Approximation for Conservation Laws (MIRACL), in preparation.

[27] R. Leveque, Nonlinear Conservation Laws and Finite Volume Methods for Astrophysical Fluid Flow, in Computational Methods for Astrophysical Fluid Flow, R. J. Leveque, D. Mihalas, E. Dorfi,and E. Mueller, 27th Saas-Fee Advanced Course Lecture Notes, Edited by O. Steiner and A. Gautschy, Springer-Verlag, 1998.

[28] G. Strang, On the Construction and Comparison of Difference Schemes, SIAM Journal of Numerical Analysis, No. 5, pp. 506-571, 1968.

[29] Mohit Arora, Explicit Characteristic-based High Resolution Algorithms for Hyperbolic Conservation Laws with Stiff Source Terms, PhD thesis, Department of Aerospace Engineering, University of Michigan, 1996.

[30] S. Jin, L. Pareschi and G. Toscani, Diffusive Relaxation Schemes for Multiscale Discrete Velocity Kinetic Equations, SIAM Journal of Numerical Analysis, vol. 35, no. 6, pp. 2405-2439, 1998.

[31] B. Manoj Kumar, S.V. Raghurama Rao R. Balasubramanian S.M. Deshpande, Kinetic Flux Vector Splitting for Navier–Stokes equatons based on Chapman–Enskog distribution, 97 FM 9, Fluid Mechanics reports, Department of Aerospace Engineering, Indian Institute of Science, Bangalore, India, 1997.

[32] Anish Kumar Gupta and S.V. Raghurama Rao, A Relaxation Scheme for Navier–Stokes Equations, in preparation.