A Class of the Relaxation Schemes for Two-Dimensional Euler Systems of Gas Dynamics

Mapundi K. Banda[1] and Mohammed Seaïd[2]

[1] Fachbereich Mathematik, TU Darmstadt, 64289 Darmstadt, Germany
banda@mathematik.tu-darmstadt.de
[2] Fachbereich Mathematik, TU Darmstadt, 64289 Darmstadt, Germany
seaid@mathematik.tu-darmstadt.de

Abstract. In the computation of approximate solutions to hyperbolic conservation laws, relaxation schemes have proven to be very useful. In this paper we present a new higher order relaxation scheme based on higher order nonoscillatory central space discretization and higher order time discretization without use of Riemann solvers. Numerical experiments with 2D Euler systems of gas dynamics are presented to demonstrate the remarkable accuracy of the relaxation scheme.

1 Introduction

In this paper, we further generalise and extend the relaxation schemes of S. Jin, and Z. Xin [4] to higher order accuracy to approximate solutions of compressible Euler system of equations for gas dynamics written in conservative form as:

$$\frac{\partial}{\partial t}\mathbf{U} + \frac{\partial}{\partial x}\mathbf{F}(\mathbf{U}) + \frac{\partial}{\partial y}\mathbf{G}(\mathbf{U}) = 0, \tag{1}$$

where

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ E \end{pmatrix}, \quad \mathbf{F}(\mathbf{U}) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho u v \\ u(E+p) \end{pmatrix}, \quad \mathbf{G}(\mathbf{U}) = \begin{pmatrix} \rho v \\ \rho u v \\ \rho v^2 + p \\ v(E+p) \end{pmatrix}.$$

Here ρ, u, v, p and E are the density, the x- and y-velocities, the pressure and the total energy, respectively. The equation of state $p = (\gamma - 1)\big(E - \frac{\rho}{2}(u^2+v^2)\big)$ is required, where the specific heat ratio $\gamma = 1.4$ for an ideal gas.
The eigenvalues of the Jacobian matrix $\partial\mathbf{F}/\partial\mathbf{U}$ (or $\partial\mathbf{G}/\partial\mathbf{U}$) are $\lambda_1 = u - c$, $\lambda_2 = \lambda_3 = u$ and $\lambda_4 = u + c$ (or $\mu_1 = v - c$, $\mu_2 = \mu_3 = v$ and $\mu_4 = v + c$). These are the characteristic speeds for one-dimensional gas dynamics and are needed here only for the estimation of relaxation variables. The sound speed c is defined by $c^2 = \gamma p/\rho$.

Many numerical schemes have been developed to approximate the solutions of the system (1). For instance, the Godunov methods. All these methods are easy to formulate and to implement. What we would like to present here is an alternate scheme which provides high resolution schemes at a low cost. Here the characteristic information of the flow is included but there is no need to solve (approximate) Riemann problems.

Relaxation methods make use of the characteristic variables of the system, finite speed of propagation and do not need Riemann solvers. In the same spirit they are very similar to central schemes [5]. The relaxation system proposed by Jin and Xin [4] and used henceforth for designing schemes is:

$$\begin{aligned} \frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{V}}{\partial x} + \frac{\partial \mathbf{W}}{\partial y} &= 0, \\ \frac{\partial \mathbf{V}}{\partial t} + \mathbf{A}^2 \frac{\partial \mathbf{U}}{\partial x} &= -\frac{1}{\tau}\Big(\mathbf{V} - \mathbf{F}(\mathbf{U})\Big), \\ \frac{\partial \mathbf{W}}{\partial t} + \mathbf{B}^2 \frac{\partial \mathbf{U}}{\partial y} &= -\frac{1}{\tau}\Big(\mathbf{W} - \mathbf{G}(\mathbf{U})\Big), \end{aligned} \tag{2}$$

where $\tau > 0$ is the relaxation rate. The matrices $\mathbf{A} = \mathrm{diag}\{a_1, \ldots, a_4\}$ and $\mathbf{B} = \mathrm{diag}\{b_1, \ldots, b_4\}$ are appropriate diagonal matrices. In the zero relaxation limit, $\tau \longrightarrow 0$, solution of (2) approaches a solution of the original system (1) by the local equilibrium

$$\mathbf{V} = \mathbf{F}(\mathbf{U}) \qquad \text{and} \qquad \mathbf{W} = \mathbf{G}(\mathbf{U}), \tag{3}$$

provided the subcharacteristic condition [4, 9, 7, 8], holds:

$$\frac{|\lambda_k|}{a_k} + \frac{|\mu_k|}{b_k} \leq 1, \qquad \forall \quad k = 1, \ldots, 4. \tag{4}$$

The relaxation schemes proposed in [4] are based on the finite difference discretization of (2), where, in particular, the authors consider a first order upwind and a second order MUSCL scheme, together with a second order TVD implicit-explicit time integration scheme. The resulting relaxation schemes are used in the regime $\tau << h$ or even when $\tau \longrightarrow 0$ (relaxed schemes). Here h stands for the space discretization parameter.

Notice that in (2) we construct a linear hyperbolic system with a stiff source term that approximates the original system (1) with a small dissipative correction. The main advantage of considering such a system is that one is able to solve the system (2) numerically with underresolved stable discretizations without either using Riemann solver spatially and nonlinear systems of algebraic equation solvers temporally. Moreover, the relaxation system (2) has linear characteristic variables given by

$$\mathbf{V} \pm \mathbf{A}\mathbf{U}, \qquad \text{and} \qquad \mathbf{W} \pm \mathbf{B}\mathbf{U}. \tag{5}$$

To avoid initial and boundary layer in (2), initial and boundary conditions are chosen to be consistent to the local equilibrium (3).

The paper is organized as follows: In section 2, we formulate a higher order central weighted nonoscillatory space discretization. Then, a higher order TVD time stepping procedure based on implicit-explicit Runge-Kutta methods is described in section 3. Numerical results are presented in section 5. Finally, a brief conclusion is given in section 6.

2 High Order Nonoscillatory Space Discretization

For the space discretization of the equation (2), we cover Ω with rectangular cells $C_{i,j} := [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}] \times [y_{j-\frac{1}{2}}, y_{j+\frac{1}{2}}]$ of uniform sizes Δx and Δy with $h = \max(\Delta x, \Delta y)$. The cells, $C_{i,j}$, are centred at $(x_i = i\Delta x, y_j = j\Delta y)$. We use the notation:

$$\omega_{i\pm\frac{1}{2},j}(t) := \omega(x_{i\pm\frac{1}{2}}, y_j, t), \qquad \omega_{i,j\pm\frac{1}{2}}(t) := \omega(x_i, y_{j\pm\frac{1}{2}}, t),$$

$$\text{and} \qquad \omega_{i,j}(t) := \frac{1}{\Delta x}\frac{1}{\Delta y}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}}\int_{y_{j-\frac{1}{2}}}^{y_{i+\frac{1}{2}}} \omega(x,y,t)dxdy,$$

to denote the point-values and the approximate cell-average of the function ω at $(x_{i\pm\frac{1}{2}}, y_j, t)$, $(x_i, y_{j\pm\frac{1}{2}}, t)$, and (x_i, y_j, t), respectively. We define the following difference operators

$$\mathcal{D}_x\omega_{i,j} := \frac{\omega_{i+\frac{1}{2},j} - \omega_{i-\frac{1}{2},j}}{\Delta x}, \qquad \mathcal{D}_y\omega_{i,j} := \frac{\omega_{i,j+\frac{1}{2}} - \omega_{i,j-\frac{1}{2}}}{\Delta y} \tag{6}$$

Then, the semi-discrete approximation of (2) is

$$\begin{aligned}
\frac{d\mathbf{U}_{i,j}}{dt} + \mathcal{D}_x\mathbf{V}_{i,j} + \mathcal{D}_y\mathbf{W}_{i,j} &= 0, \\
\frac{d\mathbf{V}_{i,j}}{dt} + \mathbf{A}^2\mathcal{D}_x\mathbf{U}_{i,j} &= -\frac{1}{\tau}\Big(\mathbf{V}_{i,j} - \mathbf{F}(\mathbf{U})_{i,j}\Big), \\
\frac{d\mathbf{W}_{i,j}}{dt} + \mathbf{B}^2\mathcal{D}_y\mathbf{U}_{i,j} &= -\frac{1}{\tau}\Big(\mathbf{W}_{i,j} - \mathbf{G}(\mathbf{U})_{i,j}\Big),
\end{aligned} \tag{7}$$

The approximate solution is reconstructed by a piecewise polynomial over the grid points as:

$$\omega(x,y,t) = \sum_{i,j} p_{i,j}(x,y;\omega)\chi_{i,j}(x,y), \qquad \chi_{i,j} = \mathbb{1}_{C_{i,j}}, \tag{8}$$

where $p_{i,j}$ are polynomials defined in $C_{i,j}$. The degree of $p_{i,j}$ is determined by the required order of accuracy of the method. In this paper we consider the third order CWENO reconstruction, dimension by dimension [6,3]. Thus,

$$p_{i,j}(x,y;\omega) := p_i(x;\omega) + p_j(y;\omega).$$

Here $p_i(x;\omega)$ and $p_j(y;\omega)$ are x- and y-polynomials. In the following we formulate the x-direction polynomial $p_i(x;\omega)$, the formulation of $p_j(y;\omega)$ can be done analogously.

$$p_i(x;\omega) = W_L P_L(x) + W_R P_R(x) + W_C P_C(x),$$

where

$$W_l = \frac{\alpha_l}{\sum_m \alpha_m}, \quad l,m \in \{L,R,C\}, \qquad \sum_l W_l = 1, \qquad \alpha_l = \frac{c_l}{(\varepsilon + IS_l)^2},$$

$$c_L = c_R = \frac{1}{4}, \quad c_C = \frac{1}{2}, \quad IS_L = (\omega_{i,j} - \omega_{i-1,j})^2, \quad IS_R = (\omega_{i+1,j} - \omega_{i,j})^2,$$
$$IS_C = \frac{13}{3}(\omega_{i+1,j} - 2\omega_{i,j} + \omega_{i-1,j})^2 + \frac{1}{4}(\omega_{i+1,j} - \omega_{i-1,j})^2,$$
$$P_L(x) = \frac{\omega_{i,j}}{2} + \frac{\omega_{i,j} - \omega_{i-1,j}}{\Delta x}(x - x_i), \quad P_R(x) = \frac{\omega_{i,j}}{2} + \frac{\omega_{i+1,j} - \omega_{i,j}}{\Delta x}(x - x_i),$$
$$P_C(x) = \frac{\omega_{i,j}}{2} - \frac{1}{24}(\omega_{i+1,j} - 2\omega_{i,j} + \omega_{i-1,j}) + \frac{\omega_{i+1,j} - \omega_{i-1,j}}{2\Delta x}(x - x_i) + \frac{(\omega_{i+1,j} - 2\omega_{i,j} + \omega_{i-1,j})}{(\Delta x)^2}(x - x_i)^2.$$

The constant ε guarantees that the denominator does not vanish and is empirically taken to be 10^{-6}.

With this background we can now discretize the characteristic variables (5) as follows:

$$(v + a_k u)_{i+\frac{1}{2},j} = p_i(x_{i+\frac{1}{2}}; v + a_k u), \qquad (v - a_k u)_{i+\frac{1}{2},j} = p_{i+1}(x_{i+\frac{1}{2}}; v - a_k u);$$
$$(w + b_k u)_{i,j+\frac{1}{2}} = p_j(y_{j+\frac{1}{2}}; w + b_k u), \qquad (w - b_k u)_{i,j+\frac{1}{2}} = p_{j+1}(y_{j+\frac{1}{2}}; w - b_k u).$$

Here u, v, w, a_k and b_k are the k-th ($k = 1, \ldots, 4$) components of $\mathbf{U}$, $\mathbf{V}$, $\mathbf{W}$, $\mathbf{A}$ and $\mathbf{B}$ respectively. Hence:

$$u_{i+\frac{1}{2},j} = \frac{1}{2a_k}\Big(p_i(x_{i+\frac{1}{2}}; v + a_k u) - p_{i+1}(x_{i+\frac{1}{2}}; v - a_k u)\Big),$$
$$v_{i+\frac{1}{2}}, j = \frac{1}{2}\Big(p_i(x_{i+\frac{1}{2}}; v + a_k u) + p_{i+1}(x_{i+\frac{1}{2}}; v - a_k u)\Big);$$
$$u_{i,j+\frac{1}{2}} = \frac{1}{2b_k}\Big(p_j(y_{j+\frac{1}{2}}; w + b_k u) - p_{j+1}(x_{j+\frac{1}{2}}; w - b_k u)\Big),$$
$$w_{i,j+\frac{1}{2}} = \frac{1}{2}\Big(p_j(x_{j+\frac{1}{2}}; w + b_k u) + p_{j+1}(x_{j+\frac{1}{2}}; w - b_k u)\Big).$$

Therefore, we obtain the following expressions for the numerical fluxes in (6)

$$u_{i+\frac{1}{2},j} := \frac{u_{i,j} + u_{i+1,j}}{2} - \frac{v_{i+1,j} - v_{i,j}}{2a_k} + \frac{\sigma^{x,+}_{i,j} + \sigma^{x,-}_{i+1,j}}{4a_k},$$
$$v_{i+\frac{1}{2},j} := \frac{v_{i,j} + v_{i+1,j}}{2} - a_k\frac{u_{i+1,j} - u_{i,j}}{2} + \frac{\sigma^{x,+}_{i,j} - \sigma^{x,-}_{i+1,j}}{4};$$
$$u_{i,j+\frac{1}{2}} := \frac{u_{i,j} + u_{i,j+1}}{2} - \frac{w_{i,j+1} - w_{i,j}}{2b_k} + \frac{\sigma^{y,+}_{i,j} + \sigma^{y,-}_{i,j+1}}{4b_k},$$
$$w_{i,j+\frac{1}{2}} := \frac{w_{i,j} + w_{i+1,j}}{2} - b_k\frac{u_{i,j+1} - u_{i,j}}{2} + \frac{\sigma^{y,+}_{i,j} - \sigma^{y,-}_{i,j+1}}{4},$$

where $\sigma^{x,\pm}_{i,j}$, $\sigma^{y,\pm}_{i,j}$ are defined as:

$$\sigma^{x,\pm}_{i,j} = W^{\pm}_L\Big((v \pm a_k u)_{i,j} - (v \pm a_k u)_{i-1,j}\Big) + W^{\pm}_R\Big((v \pm a_k u)_{i+1,j} - (v \pm a_k u)_{i,j}\Big)$$

$$
\begin{aligned}
&+\frac{W_C^\pm}{2}\Big((v\pm a_k u)_{i+1,j}-(v\pm a_k u)_{i-1,j}\Big)\\
&+\frac{W_C^\pm}{3}\Big((v\pm a_k u)_{i+1,j}-2(v\pm a_k u)_{i,j}+(v\pm a_k u)_{i-1,j}\Big)\\
&-\frac{W_C^\pm}{6}\Big((v\pm a_k u)_{i,j+1}-2(v\pm a_k u)_{i,j}+(v\pm a_k u)_{i,j-1}\Big),
\end{aligned}
$$

$$
\begin{aligned}
\sigma_{i,j}^{y,\pm} = {} & W_L^\pm\Big((w\pm b_k u)_{i,j}-(w\pm b_k u)_{i,j-1}\Big)+W_R^\pm\Big((w\pm b_k u)_{i,j+1}-(w\pm b_k u)_{i,j}\Big)\\
&+\frac{W_C^\pm}{2}\Big((w\pm b_k u)_{i,j+1}-(w\pm b_k u)_{i,j-1}\Big)\\
&+\frac{W_C^\pm}{3}\Big((w\pm b_k u)_{i,j+1}-2(w\pm b_k u)_{i,j}+(w\pm b_k u)_{i,j-1}\Big)\\
&-\frac{W_C^\pm}{6}\Big((w\pm b_k u)_{i+1,j}-2(w\pm b_k u)_{i,j}+(w\pm b_k u)_{i-1,j}\Big).
\end{aligned}
$$

The weight parameters $W_L^\pm$, $W_R^\pm$ and $W_C^\pm$ for $\sigma_{i,j}^{x,\pm}$ are given by

$$
W_l^\pm=\frac{\alpha_l^\pm}{\sum_k \alpha_k^\pm},\quad l,k\in\{L,R,C\},\qquad \sum_l W_l^\pm=1,\qquad \alpha_l^\pm=\frac{c_l}{(\varepsilon\pm IS_l)^2},
$$

$$
c_L=c_R=\frac{1}{4},\qquad c_C=\frac{1}{2}
$$

$$
IS_L^\pm=\Big((v\pm a_k u)_i-(v\pm a_k u)_{i-1}\Big)^2,\quad IS_R^\pm=\Big((v\pm a_k u)_{i+1}-(v\pm a_k u)_i\Big)^2
$$

$$
\begin{aligned}
IS_C^\pm = {} & \frac{13}{3}\Big((v\pm a_k u)_{i+1}-2(v\pm a_k u)_i+(v\pm a_k u)_{i-1}\Big)^2+\\
&\frac{1}{4}\Big((v\pm a_k u)_{i+1}-(v\pm a_k u)_{i-1}\Big)^2.
\end{aligned}
$$

The corresponding weight parameters for $\sigma_{i,j}^{y,\pm}$ are obtained by changing $v\pm a_k u$ to $w\pm b_k u$ in the above formulas.

We close by pointing out that in this higher order scheme we approximate $\mathbf{F}(\mathbf{U})_{i,j}$ and $\mathbf{G}(\mathbf{U})_{i,j}$ in (2) using the fourth-order Simpson quadrature rule as opposed to the Midpoint Rule which was used in the first and second order cases in [4].

3 High Order TVD Time Discretization

The semi-discrete formulation (7) can be rewritten as a system of ordinary differential equations:

$$
\frac{d\mathcal{Y}}{dt}=\mathcal{F}(\mathcal{Y})-\frac{1}{\tau}\mathcal{G}(\mathcal{Y}), \tag{9}
$$

where the time-dependent vector functions $\mathcal{Y} := \left(\mathbf{U}_{i,j}, \mathbf{V}_{i,j}, \mathbf{W}_{i,j}\right)^T$,

$$\mathcal{F}(\mathcal{Y}) := \left(-\mathcal{D}_x \mathbf{V}_{i,j} - \mathcal{D}_y \mathbf{W}_{i,j}, -\mathbf{A}^2 \mathcal{D}_x \mathbf{U}_{i,j}, -\mathbf{B}^2 \mathcal{D}_y \mathbf{U}_{i,j}\right)^T$$

and $\mathcal{G}(\mathcal{Y}) := \left(0, \mathbf{V}_{i,j} - \mathbf{F}(\mathbf{U})_{i,j}, \mathbf{W}_{i,j} - \mathbf{G}(\mathbf{U})_{i,j}\right)^T$. Due to the presence of the stiff term in (7), one can not use explicit schemes to integrate the equations (9), particularly when $\tau \longrightarrow 0$. On the other hand, integrating the equations (9) by implicit scheme, either nonlinear or linear algebraic equations have to be solved at every time step of the computational process. To find solutions of such systems is computationally very demanding. In this paper we consider an alternative approach based on Implicit-Explicit (IMEX) Runge-Kutta splittings. The non stiff stage of the splitting for $\mathcal{F}$ is straightforwardly treated by an explicit Runge-Kutta scheme, while the stiff stage for $\mathcal{G}$ is approximated by a diagonally implicit Runge-Kutta (DIRK) scheme. Compare [2, 10] for more details.

Let Δt be the time step and $\mathcal{Y}^n$ denote the approximate solution at $t = n\Delta t$. We formulate the IMEX scheme for the system (9) as:

$$\begin{aligned} K_l &= \mathcal{Y}^n + \Delta t \sum_{m=1}^{l-1} \tilde{a}_{lm} \mathcal{F}(K_m) - \frac{\Delta t}{\tau} \sum_{m=1}^{s} a_{lm} \mathcal{G}(K_m), \quad l = 1, 2, \ldots, s, \\ \mathcal{Y}^{n+1} &= \mathcal{Y}^n + \Delta t \sum_{l=1}^{s} \tilde{b}_l \mathcal{F}(K_l) - \frac{\Delta t}{\tau} \sum_{l=1}^{s} b_l \mathcal{G}(K_l). \end{aligned} \tag{10}$$

The $s \times s$ matrices $\tilde{A} = (\tilde{a}_{lm})$; $A = (a_{lm})$ and the s-vectors $\tilde{b}$; b are the standard coefficients which characterize the IMEX s-stage Runge-Kutta scheme. They are given by the usual double Butcher tables

$$\begin{array}{c|ccccc} 0 & 0 & 0 & 0 & 0 & 0 \\ \tilde{c}_2 & \tilde{a}_{21} & 0 & 0 & 0 & 0 \\ \tilde{c}_3 & \tilde{a}_{31} & \tilde{a}_{32} & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \tilde{c}_s & \tilde{a}_{s1} & \tilde{a}_{s2} & \ldots & \tilde{a}_{ss-1} & 0 \\ \hline & \tilde{b}_1 & \tilde{b}_2 & \ldots & \tilde{b}_{s-1} & \tilde{b}_s \end{array} \qquad \begin{array}{c|ccccc} 0 & 0 & 0 & 0 & 0 & 0 \\ c_2 & a_{21} & a_{22} & 0 & 0 & 0 \\ c_3 & a_{31} & a_{32} & a_{33} & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ c_s & a_{s1} & a_{s2} & \ldots & a_{ss-1} & a_{ss} \\ \hline & b_1 & b_2 & \ldots & b_{s-1} & b_s \end{array}$$

The left and right tables represent the explicit and the implicit Runge-Kutta methods. Then, the implementation of the IMEX algorithm to solve (9) is carried out in the two following steps:

1. For $l = 1, \ldots, s$,
   (a) Evaluate K_l^* as: $K_l^* = \mathcal{Y}^n + \Delta t \sum_{m=1}^{l-2} \tilde{a}_{lm} \mathcal{F}(K_m) + \Delta t \tilde{a}_{ll-1} \mathcal{F}(k_{l-1})$.
   (b) Solve for K_l: $K_l = K_l^* - \frac{\Delta t}{\tau} \sum_{m=1}^{l-1} a_{lm} \mathcal{G}(K_m) - \frac{\Delta t}{\tau} a_{ll} \mathcal{G}(K_l)$.
2. Update $\mathcal{Y}^{n+1}$ as: $\mathcal{Y}^{n+1} = \mathcal{Y}^n + \sum_{l=1}^{s} \tilde{b}_l \mathcal{F}(K_l) - \frac{\Delta t}{\tau} \sum_{l=1}^{s} b_l \mathcal{G}(K_l)$.

Recall that what we call high order relaxation schemes are composed of the high order reconstruction (8) augmented with the high order IMEX splitting (10).

Notice that, using the relaxation scheme neither linear algebraic equation nor nonlinear source terms can arise. In addition the high order relaxation scheme is stable independently of τ, so that the choice of Δt is based only on the usual CFL condition

$$\mathrm{CFL} := \max_{1\leq i,j\leq 4}\left(\frac{\Delta t}{h},\ a_i^2\frac{\Delta t}{\Delta x},\ b_j^2\frac{\Delta t}{\Delta y}\right) \leq 1. \tag{11}$$

In our numerical computation we consider the third order IMEX scheme developed in [2], the associated double Butcher tables can be represented as:

$$\begin{array}{c|ccccc}
0 & 0 & 0 & 0 & 0 & 0 \\
\frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 \\
\frac{2}{3} & \frac{11}{18} & \frac{1}{18} & 0 & 0 & 0 \\
\frac{1}{2} & \frac{5}{6} & -\frac{5}{6} & \frac{1}{2} & 0 & 0 \\
1 & \frac{1}{4} & \frac{7}{4} & \frac{3}{4} & -\frac{7}{4} & 0 \\
\hline
 & \frac{1}{4} & \frac{7}{4} & \frac{3}{4} & -\frac{7}{4} & 0
\end{array}
\qquad\qquad
\begin{array}{c|ccccc}
0 & 0 & 0 & 0 & 0 & 0 \\
\frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 & 0 \\
\frac{2}{3} & 0 & \frac{1}{6} & \frac{1}{2} & 0 & 0 \\
\frac{1}{2} & 0 & -\frac{1}{2} & \frac{1}{2} & \frac{1}{2} & 0 \\
1 & 0 & \frac{3}{2} & -\frac{3}{2} & \frac{1}{2} & \frac{1}{2} \\
\hline
 & 0 & \frac{3}{2} & -\frac{3}{2} & \frac{1}{2} & \frac{1}{2}
\end{array}$$

Obviously, at the limit ($\tau \longrightarrow 0$) the time integration procedure tends to a time integration scheme of the limit equations based on the explicit scheme given by the left table in (10).

Remark. Note that the first and second order relaxation schemes studied earlier in [4] can be viewed as (8) taking

$$P_{i,j}(x,y;\omega) = \omega_{i,j} \quad \text{and} \quad P_{i,j}(x,y;\omega) = \omega + \frac{\acute{\omega}_{i,j}}{\Delta x}(x - x_i) + \frac{\grave{\omega}_{i,j}}{\Delta y}(y - y_j),$$

respectively. Here $\acute{\omega}_{i,j}/\Delta x$ and $\grave{\omega}_{i,j}/\Delta y$ are discrete slopes in the x and y directions. The time integration procedure in [4] can be represented as (10) where the explicit and implicit tables are given by

$$\begin{array}{c|cc}
0 & 0 & 0 \\
1 & 1 & 0 \\
\hline
 & \frac{1}{2} & \frac{1}{2}
\end{array}
\qquad\qquad
\begin{array}{c|cc}
-1 & -1 & 0 \\
2 & 1 & 1 \\
\hline
 & \frac{1}{2} & \frac{1}{2}
\end{array}$$

4 Numerical Results

We present numerical experiments for several problems (1) using the scheme introduced above. We solve three model problems. All of them have been used extensively in literature to test various numerical schemes. In all our computation the computational domain Ω is divided in $Nx \times Ny$ grid points, the relaxation rate τ is fixed to 10^{-6} and the relaxation matrices $\mathbf{A}$ and $\mathbf{B}$ are chosen locally according to (4) as:

$$a_k = 2|\lambda_k|, \qquad \text{and} \qquad b_k = 2|\mu_k|, \quad k = 1, \ldots 4.$$

The following test cases are selected:

The Shock Reflection Problem. This problem was solved by Jin and Xin [4] using the second order relaxation scheme. In our computation we take the same parameters as [4]. Thus, $\Omega = [0,4] \times [0,1]$; initially the domain Ω is filled by a free-stream supersonic inflow with Mach number 2.9. The Dirichlet boundary conditions are imposed at left and upper boundaries as

$$\begin{aligned}\mathbf{U}(0,y,t) &= (1, 2.9, 0, 5.99071)^T,\\ \mathbf{U}(x,1,t) &= (1.69997, 4.45279, -0.86074, 21.30317)^T.\end{aligned}$$

The bottom boundary is a reflecting wall and the supersonic outflow condition is applied along the right boundary. The simulation is performed until $t = 5$ using $\Delta t = 0.005$. Plots of the pressure are shown in Fig. 1 using 30 equi-distributed contours. As can be seen from this figure, the reflected shock was very well captured by the relaxation scheme.

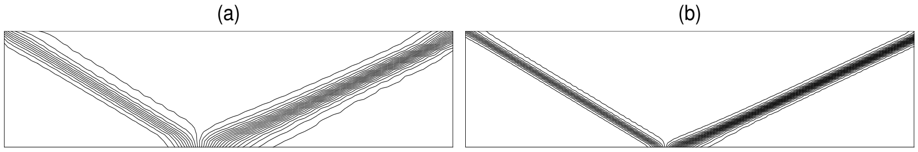

Fig. 1. The shock reflection problem. 30 equi-distributed pressure contours. (a) 60×20 grid points; (b) 120×40 grid points.

The Double-Sod Tube Shock Problem. This example is inspired by the standard 1D Sod tube shock problem [1]. Hence $\Omega = [-1,1] \times [-1,1]$ and the initial conditions are chosen as:

$$\mathbf{U}(x,y,0) = \begin{cases}(0.1, 0, 0, 0.25)^T & \text{if} \quad xy < 0,\\ (1, 0, 0, 2.5)^T & \text{otherwise.}\end{cases}$$

Homogeneous Neumann boundary condition were used, and $\Delta t = 0.001$.

In Fig. 2 we display 30 equi-distributed contour plots of the density at time $t = 0.16$. The high resolution of the new relaxation scheme is clearly visible.

The Double-Mach Reflection Problem. This test example consist of the canonical double Mach reflection problem [11]. The domain $\Omega = [0,4] \times [0,1]$. The reflecting wall lies at the bottom of the computational domain starting from $x = \frac{1}{6}$. Initially a right-moving Mach 10 shock is positioned at $x = \frac{1}{6}$, $y = 0$ and makes a 60° angle with the x-axis. For the bottom boundary, the exact post-shock condition is imposed for the part from $x = 0$ to $x = \frac{1}{6}$ and a reflective boundary condition is used for the rest. At the top boundary of the domain Ω, the flow values are set to describe the exact motion of the Mach 10 shock.

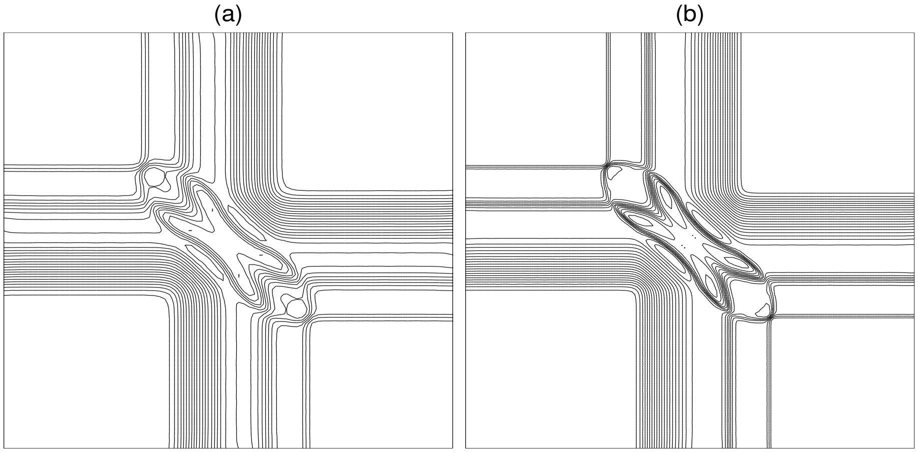

Fig. 2. The double-Sod tube shock problem. 30 equi-distributed density contours. (a) 100×100 grid points; (b) 200×200 grid points.

Fig. 3 shows 30 equi-distributed contour plots of the density at time $t = 0.2$ with $\Delta t = 0.0005$. Only part of Ω, $[0, 3] \times [0, 1]$, is shown. We can see the complicated structures being captured by the new relaxation scheme.

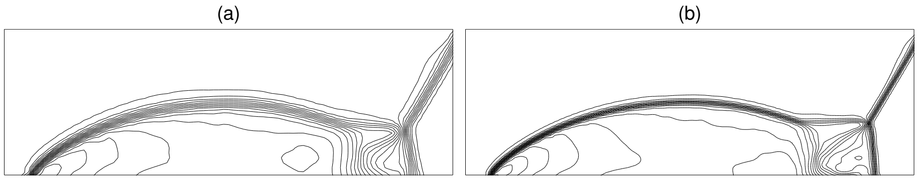

Fig. 3. The double-Mach reflection problem. 30 equi-distributed pressure contours. (a) 120×30 grid points; (b) 240×60 grid points.

5 Concluding Remarks

We have described here a third order relaxation scheme for two-dimensional Euler system of equation for inviscid flow. The system of equations is reformulated into a relaxing system. For the space discretization a generalisation of the interpolating polynomial is presented and here a third order CWENO reconstruction is used. For the time integration a third order Runge-Kutta splitting has been used. In this approach very good accuracy is achieved without using Riemann solvers nor solving nonlinear systems. The ability of the methods to handle such nonlinear systems allows for generalization to a much broader set of hyperbolic system equations.

Table 1. CPU time in minutes for the above test problems

Test problem	Grid points	Δt	N$^{\underline{o}}$ steps	CPU
The Shock Reflection Problem	120×40	0.005	1000	16
The Double-Sod Tube Shock Problem	100×100	0.001	160	7
The Double-Mach Reflection Problem	240×60	0.0005	400	12

The multidimensional algorithms presented in this paper can be highly optimized for the vector computers, because they are explicit procedures and contain no recursive elements. Some difficulties arise from the fact that for efficient vectorization the data should be stored continuously within long vectors rather than two-dimensional arrays. For completeness, we summarize in table 1 the CPU time, measured on a PC with AMD-K6 200 processor running FORTRAN code under Linux 2.2, for the test problems presented in this paper.

Future directions for this work include the following: improvement to include unstructured grids, development of refinement strategies, and three-dimensional problems.

References

1. Aregba-Driollet, D., Natalini, R.: Discrete Kinetic Schemes for Multidimensional Systems of Conservation Laws. SIAM J. Numer. Anal. **37** (2000) 1973–2004
2. Ascher, U., Ruuth, S., Spiteri, R.: Implicit-Explicit Runge-Kutta Methods for Time-Dependent Partial Differential Equations. Appl. Numer. Math. **25** (1997) 151–167
3. Banda, M.K., Klar, A., Pareschi, L., Seaïd, M.: Lattice-Boltzmann Type Relaxation Systems and Higher Order Relaxation Schemes for The Incompressible Navier-Stokes Equation. Submitted
4. Jin S., Xin, Z.: The Relaxation Schemes for Systems of Conservation Laws in Arbitrary Space Dimensions. Comm. Pure Appl. Math. **48** (1995) 235–276
5. Kurganov, A., Tadmor, E.: New High-Resolution Central Schemes for Nonlinear Conservation Laws of Convection-Diffusion Equations. J. Comp. Phys. **160** (2000) 241–282
6. Kurganov, A., Levy, D.: A Third-Order Semi-Discrete Central Scheme for Conservation Laws and Convection-Diffusion Equations. SIAM J. Sci. Comp. **22** (2000) 1461–1488
7. Lattanzio C., Serre D.: Convergence of a relaxation scheme for hyperbolic systems of conservation laws. Numer. Math. **88** (2001) 121–134
8. Liu, H.L., WarneckeG.: Convergence rates for relaxation schemes approximating conservation laws. SIAM J. Numer. Anal. **37** (2000) 1316–1337
9. Natalini, R.: Convergence to equilibrium for relaxation approximations of conservation laws. Comm. Pure Appl. Math. **49** (1996) 795–823
10. Pareschi, L., RussoG.: Implicit-Explicit Runge-Kutta Schemes for Stiff Systems of Differential Equations. Preprint
11. Woodward, P., Colella, P.: The Numerical Simulation of Two-Dimensional Fluid Flow with Strong Shocks. J. Comp. Physics. **54** (1984) 115–173