MULTIDIMENSIONAL HYPERBOLIC CONSERVATION LAWS BY RELAXATION APPROXIMATIONS

MOHAMMED SEAID[1]

(September 3, 2003)

Abstract

We construct and implement a new nonoscillatory relaxation scheme for multidimensional hyperbolic systems of conservation laws. The method transforms the nonlinear hyperbolic system to a semilinear model with a relaxation source term and linear characteristics which can be solved numerically without using either Riemann solver or linear iterations. To discretize the relaxation system we consider a high-resolution reconstruction in space and a TVD Runge-Kutta time integration. Detailed formulation of the scheme is given for problems in three space dimensions and numerical experiments are implemented in both scalar and system cases to show the effectiveness of the method.

1. Introduction

In this paper we are interested in solving numerically the multidimensional hyperbolic system of conservation laws

$$\begin{aligned} \frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}(\mathbf{U})}{\partial x} + \frac{\partial \mathbf{G}(\mathbf{U})}{\partial y} + \frac{\partial \mathbf{H}(\mathbf{U})}{\partial z} = \mathbf{0}, \quad t > 0, \quad (x, y, z) \in \mathbb{R}^3, \\ \mathbf{U}(t = 0, x, y, z) = \mathbf{U}_0(x, y, z), \end{aligned} \tag{1}$$

where $\mathbf{U}(t, x, y, z) \in \mathbb{R}^N$ is a vector of conserved quantities; $\mathbf{F}(\mathbf{U}) \in \mathbb{R}^N$, $\mathbf{G}(\mathbf{U}) \in \mathbb{R}^N$ and $\mathbf{H}(\mathbf{U}) \in \mathbb{R}^N$ are nonlinear flux functions; and $\mathbf{U}_0 \in \mathbb{R}^N$ is given initial data. We assume that the Jacobian $\partial \mathbf{F}/\partial \mathbf{U}$, $\partial \mathbf{G}/\partial \mathbf{U}$ and $\partial \mathbf{H}/\partial \mathbf{U}$ are diagonalizable with real eigenvalues $\{\lambda_1, \ldots, \lambda_N\}$, $\{\mu_1, \ldots, \mu_N\}$ and $\{\xi_1, \ldots, \xi_N\}$, respectively.

[1]Fachbereich Mathematik, Technische Universität Darmstadt, 64289 Darmstadt, Germany. E-mail: seaid@mathematik.tu-darmstadt.de

The relaxation system we propose in this paper reads

$$\begin{aligned}
\frac{\partial \mathbf{U}}{\partial t}+\frac{\partial \mathbf{V}}{\partial x}+\frac{\partial \mathbf{W}}{\partial y}+\frac{\partial \mathbf{Z}}{\partial z} &= \mathbf{0},\\
\frac{\partial \mathbf{V}}{\partial t}+\mathbf{A}\frac{\partial \mathbf{U}}{\partial x} &= -\frac{1}{\varepsilon}\big(\mathbf{V}-\mathbf{F}(\mathbf{U})\big),\\
\frac{\partial \mathbf{W}}{\partial t}+\mathbf{B}\frac{\partial \mathbf{U}}{\partial y} &= -\frac{1}{\varepsilon}\big(\mathbf{W}-\mathbf{G}(\mathbf{U})\big),\\
\frac{\partial \mathbf{Z}}{\partial t}+\mathbf{C}\frac{\partial \mathbf{U}}{\partial z} &= -\frac{1}{\varepsilon}\big(\mathbf{Z}-\mathbf{H}(\mathbf{U})\big),
\end{aligned} \tag{2}$$

where $\mathbf{V}\in\mathbb{R}^N$, $\mathbf{W}\in\mathbb{R}^N$ and $\mathbf{Z}\in\mathbb{R}^N$ are relaxation variables; $\mathbf{A}=\mathrm{diag}\{A_1,\ldots,A_N\}$, $\mathbf{B}=\mathrm{diag}\{B_1,\ldots,B_N\}$ and $\mathbf{C}=\mathrm{diag}\{C_1,\ldots,C_N\}$ are positive diagonal matrices; and ε is the relaxation time. The relaxation system (2) has a typical semilinear structure with linear characteristic variables

$$\mathbf{V}\pm\sqrt{\mathbf{A}}\mathbf{U}, \qquad \mathbf{W}\pm\sqrt{\mathbf{B}}\mathbf{U} \qquad \text{and} \qquad \mathbf{Z}\pm\sqrt{\mathbf{C}}\mathbf{U}. \tag{3}$$

In the zero relaxation limit $\varepsilon\longrightarrow 0$, we recover the original system (1) provided the subcharacteristic condition [4, 8, 6],

$$\frac{\lambda_\nu^2}{A_\nu}+\frac{\mu_\nu^2}{B_\nu}+\frac{\xi_\nu^2}{C_\nu}\le 1, \qquad \forall \quad \nu=1,\ldots,N \tag{4}$$

holds in (2). Note that if we set the local equilibrium

$$\mathbf{V}=\mathbf{F}(\mathbf{U}), \quad \mathbf{W}=\mathbf{G}(\mathbf{U}) \quad \text{and} \quad \mathbf{Z}=\mathbf{H}(\mathbf{U}), \tag{5}$$

then the first equation of (2) reduces to the original conservation law (1).

2. Third-order relaxation method

To design a relaxation scheme for the system (2), it is convenient to treat the spatial discretization and the time discretization separately using the method of lines.

2.1. Semi-discrete approximation We divide the spatial domain into cells $I_{i,j,k}=[x_{i-\frac12},x_{i+\frac12}]\times[y_{j-\frac12},y_{j+\frac12}]\times[z_{k-\frac12},z_{k+\frac12}]$ with uniform sizes Δx, Δy and Δz and centred at $(x_i=i\Delta x, y_j=j\Delta y, z_k=k\Delta z)$. We use the notations $\mathbf{U}_{i\pm\frac12,j,k}(t)=\mathbf{U}(t,x_{i\pm\frac12},y_j,z_k)$, $\mathbf{U}_{i,j\pm\frac12,k}(t)=\mathbf{U}(t,x_i,y_{j\pm\frac12},z_k)$, $\mathbf{U}_{i,j,k\pm\frac12}(t)=\mathbf{U}(t,x_i,y_j,z_{k\pm\frac12})$ and

$$\mathbf{U}_{i,j,k}(t)=\frac{1}{\Delta x}\frac{1}{\Delta y}\frac{1}{\Delta z}\int_{x_{i-\frac12}}^{x_{i+\frac12}}\int_{y_{j-\frac12}}^{y_{i+\frac12}}\int_{z_{k-\frac12}}^{x_{k+\frac12}}\mathbf{U}(t,x,y,z)dxdydz,$$

to denote the point-values and the approximate cell-average of the function $\mathbf{U}$ at $(t, x_{i\pm\frac{1}{2}}, y_j, z_k)$, $(t, x_i, y_{j\pm\frac{1}{2}}, z_k)$, $(t, x_i, y_j, z_{k\pm\frac{1}{2}})$, and (t, x_i, y_j, z_k), respectively. We also use the following difference notation

$$\mathcal{D}_x \mathbf{U}_{i,j,k} = \frac{\mathbf{U}_{i+\frac{1}{2},j,k} - \mathbf{U}_{i-\frac{1}{2},j,k}}{\Delta x}, \qquad \mathcal{D}_y \mathbf{U}_{i,j,k} = \frac{\mathbf{U}_{i,j+\frac{1}{2},k} - \mathbf{U}_{i,j-\frac{1}{2},k}}{\Delta y}, \tag{6}$$
$$\mathcal{D}_z \mathbf{U}_{i,j,k} = \frac{\mathbf{U}_{i,j,k+\frac{1}{2}} - \mathbf{U}_{i,j,k-\frac{1}{2}}}{\Delta z}.$$

Then, the space discretization of (2) reads

$$\begin{aligned} \frac{d\mathbf{U}_{i,j,k}}{dt} + \mathcal{D}_x \mathbf{V}_{i,j,k} + \mathcal{D}_y \mathbf{W}_{i,j,k} + \mathcal{D}_z \mathbf{Z}_{i,j,k} &= \mathbf{0}, \\ \frac{d\mathbf{V}_{i,j,k}}{dt} + \mathbf{A}\mathcal{D}_x \mathbf{U}_{i,j,k} &= -\frac{1}{\varepsilon}\left(\mathbf{V}_{i,j,k} - \mathbf{F}(\mathbf{U})_{i,j,k}\right), \\ \frac{d\mathbf{W}_{i,j,k}}{dt} + \mathbf{B}\mathcal{D}_y \mathbf{U}_{i,j,k} &= -\frac{1}{\varepsilon}\left(\mathbf{W}_{i,j,k} - \mathbf{G}(\mathbf{U})_{i,j,k}\right), \\ \frac{d\mathbf{Z}_{i,j,k}}{dt} + \mathbf{C}\mathcal{D}_z \mathbf{U}_{i,j,k} &= -\frac{1}{\varepsilon}\left(\mathbf{Z}_{i,j,k} - \mathbf{H}(\mathbf{U})_{i,j,k}\right), \end{aligned} \tag{7}$$

The ν-th component of approximate solution is reconstructed by a piecewise polynomial over the grid points as

$$U_\nu(x, y, z, t) = \sum_{i,j,k} \mathcal{P}_{i,j,k}(x, y, z; \mathbf{U}) \chi_{i,j,k}(x, y, z), \tag{8}$$

where $\chi_{i,j,k}$'s are the characteristic functions in the cell $I_{i,j,k}$. The polynomials $\mathcal{P}_{i,j,k}$ are defined in $I_{i,j,k}$ and reconstructed direction by direction as

$$\mathcal{P}_{i,j,k}(x, y, z; \mathbf{U}) = \mathcal{P}_i(x; \mathbf{U}) + \mathcal{P}_j(y; \mathbf{U}) + \mathcal{P}_k(z; \mathbf{U}).$$

For simplicity in presentation, the subscript ν will be omitted in (8). The degree of the polynomials $\mathcal{P}_i$, $\mathcal{P}_j$ and $\mathcal{P}_k$ is determined by the required order of accuracy of the method. In this paper we consider the third order CWENO reconstruction [5]. In the following we formulate the x-direction polynomial $\mathcal{P}_i(x; \mathbf{U})$, the formulation of $\mathcal{P}_j(y; \mathbf{U})$ and $\mathcal{P}_k(z; \mathbf{U})$ can be done analogously. Hence

$$\mathcal{P}_i(x; \mathbf{U}) = \omega_L P_L(x) + \omega_R P_R(x) + \omega_C P_C(x),$$

where the weights ω_l, $l \in \{L, R, C\}$ are defined as

$$\omega_l = \frac{\alpha_l}{\sum_m \alpha_m}, \quad l, m \in \{L, R, C\}, \quad \alpha_l = \frac{c_l}{(IS_l)^2}, \quad c_L = c_R = \frac{1}{4}, \quad c_C = \frac{1}{2}.$$

Note that the normalizing factor $\sum_m \alpha_m$ is used here to guarantee $\sum_l \omega_l = 1$. The smoothness indicators IS_l and the polynomials $P_l(x)$ are given by

$$
\begin{aligned}
IS_L &= (U_{i,j,k} - U_{i-1,j,k})^2, \\
IS_R &= (U_{i+1,j,k} - U_{i,j,k})^2, \\
IS_C &= \frac{13}{3}(U_{i+1,j,k} - 2U_{i,j,k} + U_{i-1,j,k})^2 + \frac{1}{4}(U_{i+1,j,k} - U_{i-1,j,k})^2,
\end{aligned}
$$

$$
\begin{aligned}
P_L(x) &= \frac{U_{i,j,k}}{2} + \frac{U_{i,j,k} - U_{i-1,j,k}}{\Delta x}(x - x_i), \\
P_R(x) &= \frac{U_{i,j,k}}{2} + \frac{U_{i+1,j,k} - U_{i,j,k}}{\Delta x}(x - x_i), \\
P_C(x) &= \frac{U_{i,j,k}}{2} - \frac{1}{24}(U_{i+1,j,k} - 2U_{i,j,k} + U_{i-1,j,k}) + \\
&\quad \frac{U_{i+1,j,k} - U_{i-1,j,k}}{2(\Delta x)}(x - x_i) + \frac{(U_{i+1,j,k} - 2U_{i,j,k} + U_{i-1,j,k})}{(\Delta x)^2}(x - x_i)^2.
\end{aligned}
$$

We can now discretize the characteristic variables (3) as follows

$$
\begin{aligned}
(V \pm \sqrt{A_\nu}U)_{i+\frac{1}{2},j,k} &= \mathcal{P}_i(x_{i+\frac{1}{2}}; \mathbf{V} \pm \sqrt{\mathbf{A}}\mathbf{U}), \\
(W \pm \sqrt{B_\nu}U)_{i,j+\frac{1}{2},k} &= \mathcal{P}_j(y_{j+\frac{1}{2}}; \mathbf{W} \pm \sqrt{\mathbf{B}}\mathbf{U}), \\
(Z \pm \sqrt{C_\nu}U)_{i,j,k+\frac{1}{2}} &= \mathcal{P}_k(z_{k+\frac{1}{2}}; \mathbf{Z} \pm \sqrt{\mathbf{C}}\mathbf{U}).
\end{aligned} \tag{9}
$$

Notice U, V, W, Z, A_ν, B_ν and C_ν are the ν-th ($\nu = 1, \ldots, N$) components of $\mathbf{U}$, $\mathbf{V}$, $\mathbf{W}$, $\mathbf{Z}$, $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$, respectively. Solving (9) for the unknowns $U_{i+\frac{1}{2},j,k}$, $V_{i+\frac{1}{2},j,k}$, $U_{i,j+\frac{1}{2},k}$, $W_{i,j+\frac{1}{2},k}$, $U_{i,j,k+\frac{1}{2}}$ and $Z_{i,j,k+\frac{1}{2}}$ gives

$$
\begin{aligned}
U_{i+\frac{1}{2},j,k} &= \frac{1}{2\sqrt{A_\nu}}\left(\mathcal{P}_i(x_{i+\frac{1}{2}}; \mathbf{V} + \sqrt{\mathbf{A}}\mathbf{U}) - \mathcal{P}_{i+1}(x_{i+\frac{1}{2}}; \mathbf{V} - \sqrt{\mathbf{A}}\mathbf{U})\right), \\
V_{i+\frac{1}{2},j,k} &= \frac{1}{2}\left(\mathcal{P}_i(x_{i+\frac{1}{2}}; \mathbf{V} + \sqrt{\mathbf{A}}\mathbf{U}) + \mathcal{P}_{i+1}(x_{i+\frac{1}{2}}; \mathbf{V} - \sqrt{\mathbf{A}}\mathbf{U})\right); \\
U_{i,j+\frac{1}{2},k} &= \frac{1}{2\sqrt{B_\nu}}\left(\mathcal{P}_j(y_{j+\frac{1}{2}}; \mathbf{W} + \sqrt{\mathbf{B}}\mathbf{U}) - \mathcal{P}_{j+1}(y_{j+\frac{1}{2}}; \mathbf{W} - \sqrt{\mathbf{B}}\mathbf{U})\right), \\
W_{i,j+\frac{1}{2},k} &= \frac{1}{2}\left(\mathcal{P}_j(y_{j+\frac{1}{2}}; \mathbf{W} + \sqrt{\mathbf{B}}\mathbf{U}) + \mathcal{P}_{i+1}(y_{j+\frac{1}{2}}; \mathbf{W} - \sqrt{\mathbf{B}}\mathbf{U})\right); \\
U_{i,j,k+\frac{1}{2}} &= \frac{1}{2\sqrt{C_\nu}}\left(\mathcal{P}_k(z_{k+\frac{1}{2}}; \mathbf{Z} + \sqrt{\mathbf{C}}\mathbf{U}) - \mathcal{P}_{k+1}(z_{k+\frac{1}{2}}; \mathbf{Z} - \sqrt{\mathbf{C}}\mathbf{U})\right), \\
Z_{i,j,k+\frac{1}{2}} &= \frac{1}{2}\left(\mathcal{P}_k(z_{k+\frac{1}{2}}; \mathbf{Z} + \sqrt{\mathbf{C}}\mathbf{U}) + \mathcal{P}_{k+1}(z_{k+\frac{1}{2}}; \mathbf{Z} - \sqrt{\mathbf{C}}\mathbf{U})\right).
\end{aligned}
$$

Therefore, we obtain the following expressions for the numerical fluxes in the semi-discrete equations (7)

$$
\begin{aligned}
U_{i+\frac{1}{2},j,k} &= \frac{U_{i,j,k}+U_{i+1,j,k}}{2} - \frac{V_{i+1,j,k}-V_{i,j,k}}{2\sqrt{A_\nu}} + \frac{\sigma^{x,+}_{i,j,k}+\sigma^{x,-}_{i+1,j,k}}{4\sqrt{A_\nu}},\\
V_{i+\frac{1}{2},j,k} &= \frac{V_{i,j,k}+V_{i+1,j,k}}{2} - \sqrt{A_\nu}\frac{U_{i+1,j,k}-U_{i,j,k}}{2} + \frac{\sigma^{x,+}_{i,j,k}-\sigma^{x,-}_{i+1,j,k}}{4};\\
U_{i,j+\frac{1}{2},k} &= \frac{U_{i,j,k}+U_{i,j+1,k}}{2} - \frac{W_{i,j+1,k}-W_{i,j,k}}{2\sqrt{B_\nu}} + \frac{\sigma^{y,+}_{i,j,k}+\sigma^{y,-}_{i,j+1,k}}{4\sqrt{B_\nu}},\\
W_{i,j+\frac{1}{2},k} &= \frac{W_{i,j,k}+W_{i,j+1,k}}{2} - \sqrt{B_\nu}\frac{U_{i,j+1,k}-U_{i,j,k}}{2} + \frac{\sigma^{y,+}_{i,j,k}-\sigma^{y,-}_{i,j+1,k}}{4},\\
U_{i,j,k+\frac{1}{2}} &= \frac{U_{i,j,k}+U_{i,j,k+1}}{2} - \frac{W_{i,j,k+1}-W_{i,j,k}}{2\sqrt{C_\nu}} + \frac{\sigma^{z,+}_{i,j,k}+\sigma^{z,-}_{i,j,k+1}}{4\sqrt{C_\nu}},\\
Z_{i,j,k+\frac{1}{2}} &= \frac{Z_{i,j,k}+Z_{i,j,k+1}}{2} - \sqrt{C_\nu}\frac{U_{i,j,k+1}-U_{i,j,k}}{2} + \frac{\sigma^{z,+}_{i,j,k}-\sigma^{z,-}_{i,j,k+1}}{4},
\end{aligned}
$$

where $\sigma^{x,\pm}_{i,j,k}$, $\sigma^{y,\pm}_{i,j,k}$ and $\sigma^{z,\pm}_{i,j,k}$ are the slopes of $\mathbf{V}\pm\sqrt{\mathbf{A}}\mathbf{U}$, $\mathbf{W}\pm\sqrt{\mathbf{B}}\mathbf{U}$ and $\mathbf{Z}\pm\sqrt{\mathbf{C}}\mathbf{U}$ on the cell $I_{i,j,k}$, respectively. They are defined by

$$
\begin{aligned}
\sigma^{x,\pm}_{i,j,k} &= \omega^{\pm}_L\left((V\pm\sqrt{A_\nu}U)_{i,j,k} - (V\pm\sqrt{A_\nu}U)_{i-1,j,k}\right)\\
&+ \omega^{\pm}_R\left((V\pm\sqrt{A_\nu}U)_{i+1,j,k} - (V\pm\sqrt{A_\nu}U)_{i,j,k}\right)\\
&+ \frac{\omega^{\pm}_C}{2}\left((V\pm\sqrt{A_\nu}U)_{i+1,j,k} - (V\pm\sqrt{A_\nu}U)_{i-1,j,k}\right)\\
&+ \frac{\omega^{\pm}_C}{3}\left((V\pm\sqrt{A_\nu}U)_{i+1,j,k} - 2(V\pm\sqrt{A_\nu}U)_{i,j,k} + (V\pm\sqrt{A_\nu}U)_{i-1,j,k}\right)\\
&- \frac{\omega^{\pm}_C}{9}\left((V\pm\sqrt{A_\nu}U)_{i,j+1,k} - 2(V\pm\sqrt{A_\nu}U)_{i,j,k} + (V\pm\sqrt{A_\nu}U)_{i,j-1,k}\right)\\
&- \frac{\omega^{\pm}_C}{9}\left((V\pm\sqrt{A_\nu}U)_{i,j,k+1} - 2(V\pm\sqrt{A_\nu}U)_{i,j,k} + (V\pm\sqrt{A_\nu}U)_{i,j,k-1}\right),
\end{aligned}
$$

$$
\begin{aligned}
\sigma^{y,\pm}_{i,j,k} &= \omega^{\pm}_L\left((W\pm\sqrt{B_\nu}U)_{i,j,k} - (W\pm\sqrt{B_\nu}U)_{i,j-1,k}\right)\\
&+ \omega^{\pm}_R\left((W\pm\sqrt{B_\nu}U)_{i,j+1,k} - (W\pm\sqrt{B_\nu}U)_{i,j,k}\right)\\
&+ \frac{\omega^{\pm}_C}{2}\left((W\pm\sqrt{B_\nu}U)_{i,j+1,k} - (W\pm\sqrt{B_\nu}U)_{i,j-1,k}\right)\\
&+ \frac{\omega^{\pm}_C}{3}\left((W\pm\sqrt{B_\nu}U)_{i,j+1,k} - 2(W\pm\sqrt{B_\nu}U)_{i,j,k} + (W\pm\sqrt{B_\nu}U)_{i,j-1,k}\right)\\
&- \frac{\omega^{\pm}_C}{9}\left((W\pm\sqrt{B_\nu}U)_{i+1,j,k} - 2(W\pm\sqrt{B_\nu}U)_{i,k,j} + (W\pm\sqrt{B_\nu}U)_{i-1,j,k}\right)\\
&- \frac{\omega^{\pm}_C}{9}\left((W\pm\sqrt{B_\nu}U)_{i,j,k+1} - 2(W\pm\sqrt{B_\nu}U)_{i,k,j} + (W\pm\sqrt{B_\nu}U)_{i,j,k-1}\right),
\end{aligned}
$$

$$
\begin{aligned}
\sigma^{z,\pm}_{i,j,k} &= \omega^{\pm}_{L}\left((Z \pm \sqrt{C_\nu}U)_{i,j,k} - (Z \pm \sqrt{C_\nu}U)_{i,j,k-1}\right) \\
&+ \omega^{\pm}_{R}\left((Z \pm \sqrt{C_\nu}U)_{i,j,k+1} - (Z \pm \sqrt{C_\nu}U)_{i,j,k}\right) \\
&+ \frac{\omega^{\pm}_{C}}{2}\left((Z \pm \sqrt{C_\nu}U)_{i,j,k+1} - (Z \pm \sqrt{C_\nu}U)_{i,j,k-1}\right) \\
&+ \frac{\omega^{\pm}_{C}}{3}\left((Z \pm \sqrt{C_\nu}U)_{i,j,k+1} - 2(Z \pm \sqrt{C_\nu}U)_{i,j,k} + (Z \pm \sqrt{C_\nu}U)_{i,j,k-1}\right) \\
&- \frac{\omega^{\pm}_{C}}{9}\left((Z \pm \sqrt{C_\nu}U)_{i+1,j,k} - 2(Z \pm \sqrt{C_\nu}U)_{i,j,k} + (Z \pm \sqrt{C_\nu}U)_{i-1,j,k}\right) \\
&- \frac{\omega^{\pm}_{C}}{9}\left((Z \pm \sqrt{C_\nu}U)_{i,j,k+1} - 2(Z \pm \sqrt{C_\nu}U)_{i,j,k} + (Z \pm \sqrt{C_\nu}U)_{i,j,k-1}\right).
\end{aligned}
$$

The weight parameters $\omega^{\pm}_{L}$, $\omega^{\pm}_{R}$ and $\omega^{\pm}_{C}$ for $\sigma^{x,\pm}_{i,j}$ are given by

$$
\omega^{\pm}_{l} = \frac{\alpha^{\pm}_{l}}{\sum_m \alpha^{\pm}_{m}},\ l,m \in \{L,R,C\}, \quad \alpha^{\pm}_{l} = \frac{c_l}{(IS^{\pm}_{l})^2}, \quad c_L = c_R = \frac{1}{4}, \quad c_C = \frac{1}{2},
$$

$$
\begin{aligned}
IS^{\pm}_{L} &= \left((V \pm \sqrt{A_\nu}U)_{i,j,k} - (V \pm \sqrt{A_\nu}U)_{i-1,j,k}\right)^2, \\
IS^{\pm}_{R} &= \left((V \pm \sqrt{A_\nu}U)_{i+1,j,k} - (V \pm \sqrt{A_\nu}U)_{i,j,k}\right)^2, \\
IS^{\pm}_{C} &= \frac{13}{3}\left((V \pm \sqrt{A_\nu}U)_{i+1,j,k} - 2(V \pm \sqrt{A_\nu}U)_{i,j,k} + (V \pm \sqrt{A_\nu}U)_{i-1,j,k}\right)^2 \\
&\quad + \frac{1}{4}\left((V \pm \sqrt{A_\nu}U)_{i+1,j,k} - (V \pm \sqrt{A_\nu}U)_{i-1,j,k}\right)^2.
\end{aligned}
$$

The corresponding weight parameters for $\sigma^{y,\pm}_{i,j,k}$ or $\sigma^{z,\pm}_{i,j,k}$ are obtained by changing $V \pm \sqrt{A_\nu}U$ to $W \pm \sqrt{B_\nu}U$ or $Z \pm \sqrt{C_\nu}U$ in the above formulas and differentiating respect to y-direction or z-direction, respectively. We would like to point out that in this higher order scheme we approximate $\mathbf{F}(\mathbf{U})_{i,j,k}$, $\mathbf{G}(\mathbf{U})_{i,j,k}$ and $\mathbf{H}(\mathbf{U})_{i,j,k}$ in (7) using the fourth-order Simpson quadrature rule as opposed to the Midpoint Rule which was used in the first and second order reconstructions in [4].

2.2. Fully-discrete approximation The semi-discrete system (7) can be written as a system of ordinary differential equations of form

$$
\frac{d\mathbf{Y}}{dt} = \Phi(\mathbf{Y}) - \frac{1}{\varepsilon}\Psi(\mathbf{Y}), \tag{10}
$$

where the time-dependent vector functions

$$
\mathbf{Y} = \begin{pmatrix} \mathbf{U}_{i,j,k} \\ \mathbf{V}_{i,j,k} \\ \mathbf{W}_{i,j,k} \\ \mathbf{Z}_{i,j,k} \end{pmatrix}, \qquad \Psi(\mathbf{Y}) = \begin{pmatrix} \mathbf{0} \\ \mathbf{V}_{i,j,k} - \mathbf{F}(\mathbf{U})_{i,j,k} \\ \mathbf{W}_{i,j,k} - \mathbf{G}(\mathbf{U})_{i,j,k} \\ \mathbf{Z}_{i,j,k} - \mathbf{H}(\mathbf{U})_{i,j,k} \end{pmatrix},
$$

$$\Phi(\mathbf{Y}) = \begin{pmatrix} -\mathcal{D}_x \mathbf{V}_{i,j,k} - \mathcal{D}_y \mathbf{W}_{i,j,k} - \mathcal{D}_z \mathbf{Z}_{i,j,k} \\ -\mathbf{A}\mathcal{D}_x \mathbf{U}_{i,j,k} \\ -\mathbf{B}\mathcal{D}_y \mathbf{W}_{i,j,k} \\ -\mathbf{C}\mathcal{D}_z \mathbf{Z}_{i,j,k} \end{pmatrix}.$$

When $\varepsilon \longrightarrow 0$, the equations (10) become highly stiff and any explicit treatment of the right hand side in (10) requires extremely small time stepsizes. This fact might restrict any long term computation in (2). On the other hand, integrating the equations (10) by implicit scheme, either linear or nonlinear 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 splitting. The non stiff stage of the splitting for Φ is treated by an explicit Runge-Kutta scheme, while the stiff stage for Ψ is approximated by a diagonally implicit Runge-Kutta (DIRK) scheme. Compare [2, 9] for more details.

Let Δt be the time step and $\mathbf{Y}^n$ denotes the approximate solution at $t = n\Delta t$. We formulate the IMEX scheme for the system (10) as

$$\begin{aligned} \mathbf{K}_l &= \mathbf{Y}^n + \Delta t \sum_{m=1}^{l-1} \tilde{a}_{lm} \Phi(\mathbf{K}_m) - \frac{\Delta t}{\varepsilon} \sum_{m=1}^{s} a_{lm} \Psi(\mathbf{K}_m), \quad l = 1, 2, \ldots, s, \\ \mathbf{Y}^{n+1} &= \mathbf{Y}^n + \Delta t \sum_{l=1}^{s} \tilde{b}_l \Phi(\mathbf{K}_l) - \frac{\Delta t}{\varepsilon} \sum_{l=1}^{s} b_l \Psi(\mathbf{K}_l). \end{aligned} \tag{11}$$

The $s \times s$ matrices $\tilde{A} = (\tilde{a}_{lm})$, $\tilde{a}_{lm} = 0$ for $m \geq l$ and $A = (a_{lm})$ are chosen such that the resulting scheme is explicit in Φ, and implicit in Ψ. The s-vectors $\tilde{b}$ and b are the canonical coefficients which characterize the IMEX s-stage Runge-Kutta scheme [9]. They can be given by the standard double *tableau* in Butcher notation,

$\tilde{c}$	$\tilde{A}$
	$\tilde{b}^T$

c	A
	b^T

Here, $\tilde{c}$ and c are s-vectors used in non autonomous cases. The implementation of the IMEX algorithm to solve (10) can be carried out in the following steps:

1. For $l = 1, \ldots, s$,

   (a) Evaluate $\mathbf{K}_l^*$ as: $\mathbf{K}_l^* = \mathbf{Y}^n + \Delta t \sum_{m=1}^{l-2} \tilde{a}_{lm} \Phi(\mathbf{K}_m) + \Delta t \tilde{a}_{ll-1} \Phi(\mathbf{K}_{l-1})$.

(b) Solve for $\mathbf{K}_l$: $\mathbf{K}_l = \mathbf{K}_l^* - \dfrac{\Delta t}{\varepsilon}\sum_{m=1}^{l-1} a_{lm}\Psi(\mathbf{K}_m) - \dfrac{\Delta t}{\varepsilon}a_{ll}\Psi(\mathbf{K}_l)$.

2. Update $\mathbf{Y}^{n+1}$ as: $\mathbf{Y}^{n+1} = \mathbf{Y}^n + \Delta t\sum_{l=1}^{s}\tilde{b}_l\Phi(\mathbf{K}_l) - \dfrac{\Delta t}{\varepsilon}\sum_{l=1}^{s} b_l\Psi(\mathbf{K}_l)$.

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

$$\text{CFL} = \max_{1\le\nu\le N}\left(\frac{\Delta t}{h}, A_\nu\frac{\Delta t}{\Delta x}, B_\nu\frac{\Delta t}{\Delta y}, C_\nu\frac{\Delta t}{\Delta z}\right) \le 1, \tag{12}$$

where h denotes the maximum cell size, $h = \max(\Delta x, \Delta y, \Delta z)$.

In this paper we use the third order IMEX scheme proposed in [2], the associated double *tableau* can be represented as

0	0	0	0
γ	γ	0	0
$1-\gamma$	$\gamma-1$	$2-2\gamma$	0
	0	$\frac{1}{2}$	$\frac{1}{2}$

0	0	0	0
γ	0	γ	0
$1-\gamma$	0	$1-2\gamma$	γ
	0	$\frac{1}{2}$	$\frac{1}{2}$

where $\gamma = \frac{3+\sqrt{3}}{6}$. Other IMEX schemes of third and higher order are also discussed in [9]. It is clear that, at the limit ($\varepsilon \longrightarrow 0$) the time integration procedure tends to a time integration scheme of the limit equations based on the explicit scheme given by the left table.

3. Numerical examples

We present numerical results for some of hyperbolic equations in two and three space dimensions using our third order relaxation scheme. We consider both scalar and systems of nonlinear equations of conservation laws. In all the computational results presented in this section the relaxation rate ε is set to 10^{-6} and the characteristic speeds A_ν, B_ν and C_ν are chosen as

$$\begin{aligned}
A_{i+\frac{1}{2},j,k} &= \max_{U\in\left\{U^-_{i+\frac{1}{2},j,k}, U^+_{i+\frac{1}{2},j,k}\right\}}\left|\frac{\partial\mathbf{F}}{\partial U_\nu}(U)\right|,\\
B_{i,j+\frac{1}{2},k} &= \max_{U\in\left\{U^-_{i,j+\frac{1}{2},k}, U^+_{i,j+\frac{1}{2},k}\right\}}\left|\frac{\partial\mathbf{G}}{\partial U_\nu}(U)\right|,\\
C_{i,j,k+\frac{1}{2}} &= \max_{U\in\left\{U^-_{i,j,k+\frac{1}{2}}, U^+_{i,j,k+\frac{1}{2}}\right\}}\left|\frac{\partial\mathbf{H}}{\partial U_\nu}(U)\right|,
\end{aligned} \tag{13}$$

where the values at the cell boundary point are given by

$$
\begin{aligned}
U^-_{i+\frac{1}{2},j,k} &= p_i(x_{i+\frac{1}{2}};\mathbf{U}), \qquad & U^+_{i+\frac{1}{2},j,k} &= p_{i+1}(x_{i+\frac{1}{2}};\mathbf{U});\\
U^-_{i,j+\frac{1}{2},k} &= p_j(y_{j+\frac{1}{2}};\mathbf{U}), \qquad & U^+_{i,j+\frac{1}{2},k} &= p_{j+1}(y_{j+\frac{1}{2}};\mathbf{U});\\
U^-_{i,j,k+\frac{1}{2}} &= p_k(z_{k+\frac{1}{2}};\mathbf{U}), \qquad & U^+_{i,j,k+\frac{1}{2}} &= p_{k+1}(z_{k+\frac{1}{2}};\mathbf{U}).
\end{aligned}
$$

In order to avoid initial and boundary layer in (2), initial and boundary conditions are chosen to be consistent to the associated local equilibrium (5). For instance, if Dirichlet boundary condition is given, $\mathbf{U} = \mathbf{U}_b$, then boundary and initial conditions for (2) are given by

$$
\begin{aligned}
&\mathbf{V}(t,\mathbf{x}) = \mathbf{F}\left(\mathbf{U}_b\right), \ \mathbf{W}(t,\mathbf{x}) = \mathbf{G}\left(\mathbf{U}_b\right), \quad \mathbf{Z}(t,\mathbf{x}) = \mathbf{H}\left(\mathbf{U}_b\right),\\
&\mathbf{V}(0,\mathbf{x}) = \mathbf{F}\left(\mathbf{U}_0(\mathbf{x})\right), \ \mathbf{W}(0,\mathbf{x}) = \mathbf{G}\left(\mathbf{U}_0(\mathbf{x})\right), \quad \mathbf{Z}(0,\mathbf{x}) = \mathbf{H}\left(\mathbf{U}_0(\mathbf{x})\right).
\end{aligned}
\tag{14}
$$

A simplified flow chart for the relaxation scheme used to approximate solutions to system of equations (1) is presented in figure 1. First the semilinear relaxation system of hyperbolic equations (2) with initial and boundary-value independent variables is transformed into an ODE initial-value problem by approximation of the spatial derivatives with the third order reconstruction (8) on a dimension by dimension basis. Starting by an initial condition, the ODE problem is integrated in time by the higher order IMEX methods (11). As can be seen from the figure, the relaxation solution of (1) is a modular algorithm into which any higher order spatial discretization scheme and any higher order ODE solver can easily be incorporated.

Note that the algorithms presented in this paper can be highly optimized for the vector computers, because they not require nonlinear solvers 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 three-dimensional arrays.

3.1. Two-dimensional problems

3.1.1. Burgers equation We start by considering the invscid Burgers equation in two space dimensions

$$
\begin{aligned}
\frac{\partial u}{\partial t} + \left(\frac{u^2}{2}\right)_x + \left(\frac{u^2}{2}\right)_y = 0, \qquad & t > 0, \quad (x,y) \in [0,1]\times[0,1],\\
u(t=0,x,y) = \sin^2(\pi x)\sin^2(\pi y), \quad & (x,y) \in [0,1]\times[0,1],
\end{aligned}
\tag{15}
$$

augmented with periodic boundary conditions. By setting the flux functions

$$
F(u) = G(u) = \frac{u^2}{2},
$$

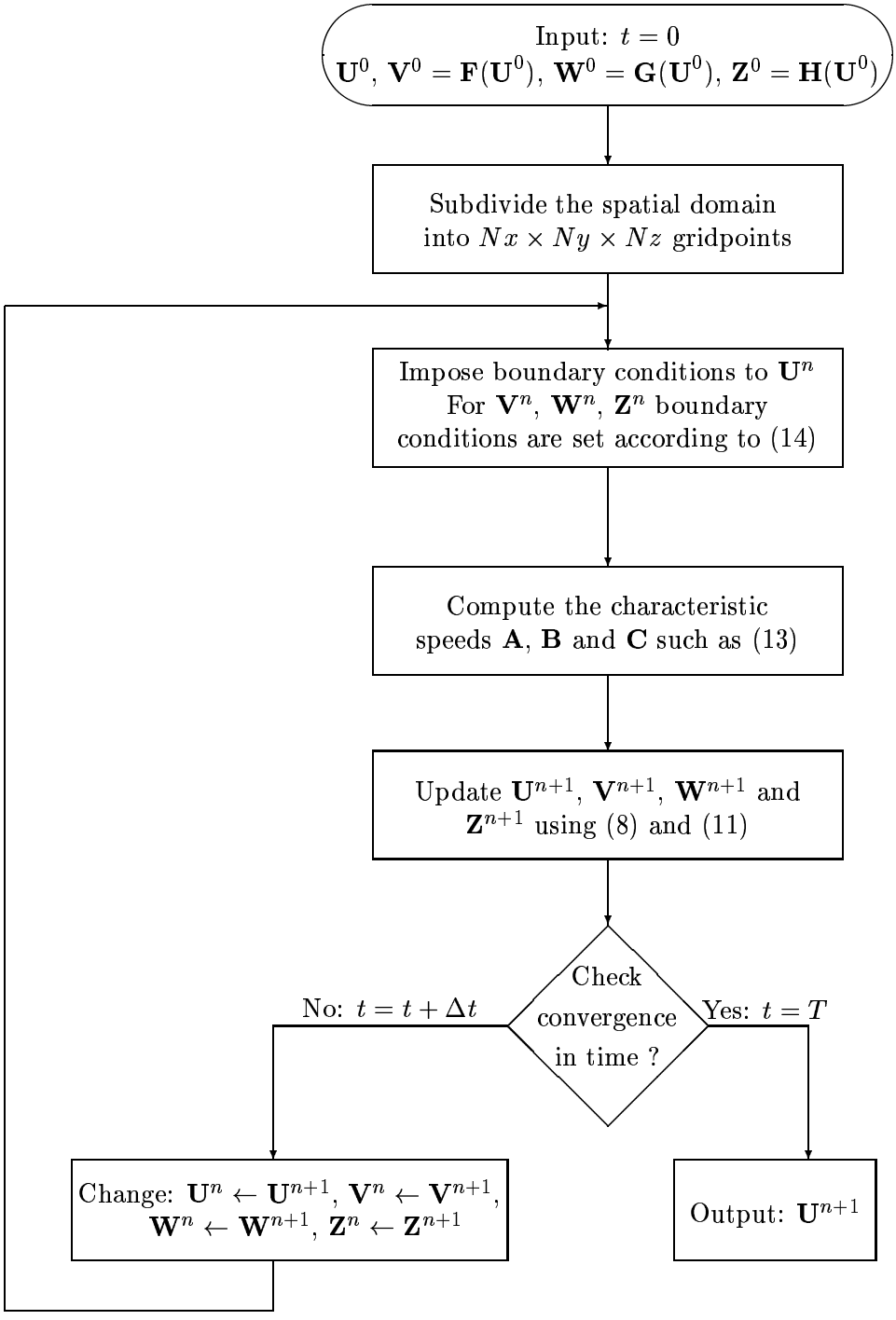

FIGURE 1. Flow chart of relaxation schemes for the three-dimensional problem (1).

the associated relaxation system to (15) can be formulated as in (2) with $\mathbf{A} = \text{diag}\{a\}$ and $\mathbf{B} = \text{diag}\{b\}$. We discretize the space domain uniformly into 50×50 gridpoints and we compute the solution using $a = 1.0$, $b = 1.0$ and CFL $= 0.87$.

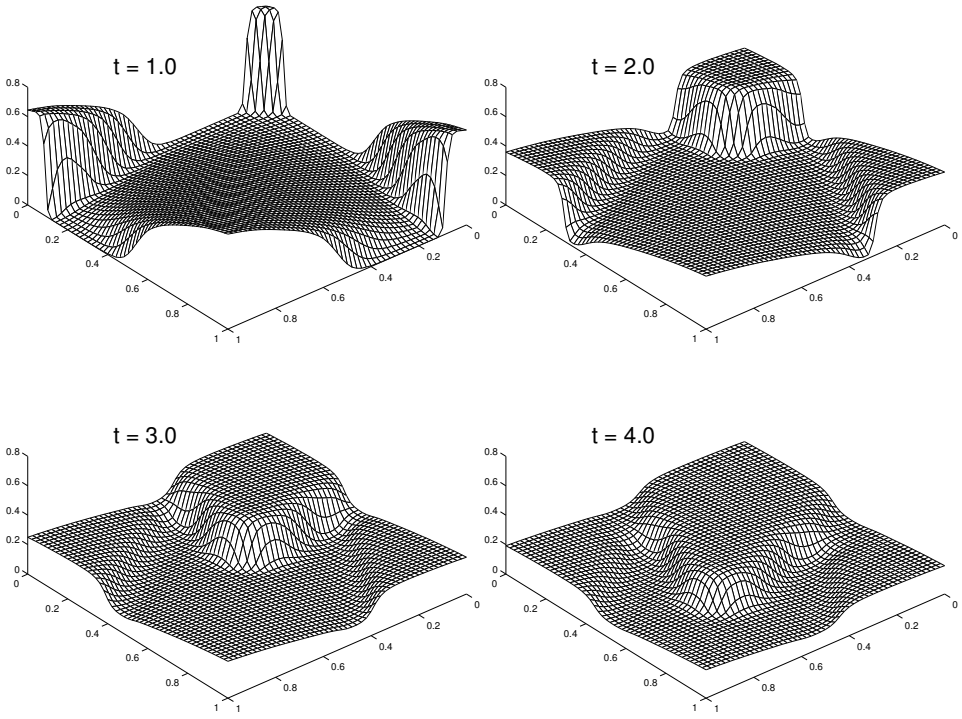

FIGURE 2. Results for the inviscid two-dimensional Burgers equation (15).

The obtained results are shown in figure 2 at four different times, $t = 1$, 2, 3, and 4. The solutions are completely free of spurious oscillations and the shocks are well resolved by the third order relaxation scheme.

3.1.2. Inviscid gas Euler equations The Euler equations for an ideal gas in two dimensions are given by the system (1) where

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ E \end{pmatrix}, \quad \mathbf{F}(\mathbf{U}) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ u(E+p) \end{pmatrix}, \quad \mathbf{G}(\mathbf{U}) = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ v(E+p) \end{pmatrix}. \quad (16)$$

In (16), ρ is the density, u is the x-velocity, v is the y-velocity, $E = \rho e = \frac{1}{2}\rho(u^2 + v^2)$ is the total energy, e is the internal energy of the gas, $p = (\gamma - 1)\rho e$ is the pressure, and γ is the ratio of specific heats. The associated relaxation system can be formulated as (2), where $\mathbf{A} = \text{diag}\{A_1, A_2, A_3, A_4\}$ and $\mathbf{B} = \text{diag}\{B_1, B_2, B_3, B_4\}$.

The eigenvalues of the Jacobian matrix $\partial \mathbf{F}(\mathbf{U})/\partial \mathbf{U}$ (or $\partial \mathbf{G}(\mathbf{U})/\partial \mathbf{U}$) 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. Thus, in all our numerical tests with equations (16) we used

$$\begin{aligned} A_1 = A_2 = A_3 = A_4 &= \max\Big(\sup|u-c|, \sup|u|, \sup|u+c|\Big), \\ B_1 = B_2 = B_3 = B_4 &= \max\Big(\sup|v-c|, \sup|v|, \sup|v+c|\Big). \end{aligned} \tag{17}$$

The following test examples are chosen:

The double Mach reflection problem: This test example consists of the canonical double Mach reflection problem [10]. The spatial 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. For comparison reasons, we use two different uniform meshes of 240×60, and 480×120 gridpoints.

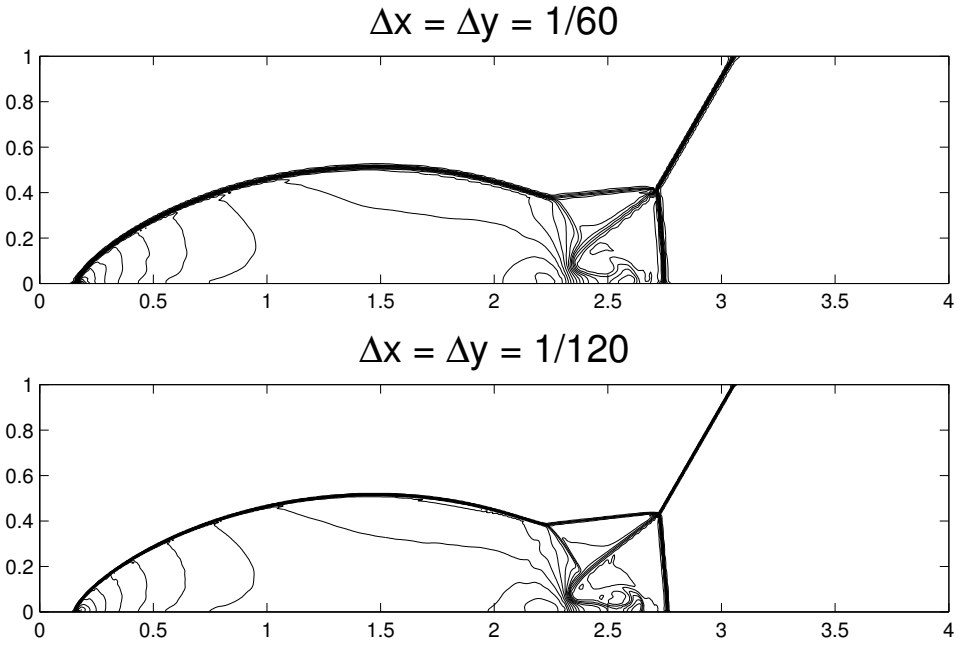

FIGURE 3. Density contours for the double Mach reflection problem on two different meshes.

Figure 3 shows 30 equi-distributed contour plots of the density at time $t = 0.2$ with $\Delta t = 0.0005$. We note that there is a very strong increase in resolution as the grids are refined due to the high order accuracy of the relaxation scheme. We can also see the complicated structures being captured by the new relaxation scheme.

The forward facing step problem: This is again a standard test problem for numerical schemes in two-dimensional Euler equations of gas dynamics (16). The setting of the problem is the following [10]:

A right going Mach 3 uniform flow enters a wind tunnel of 1 unit wide and 3 units long. The step is 0.2 units high and is located 0.6 units from the left hand end of the tunnel. The problem is initialized by a uniform, right going Mach 3 flow. Reflective boundary conditions are applied along the walls of the tunnel and inflow and outflow boundary conditions are applied at the entrance and the exit of the tunnel, respectively.

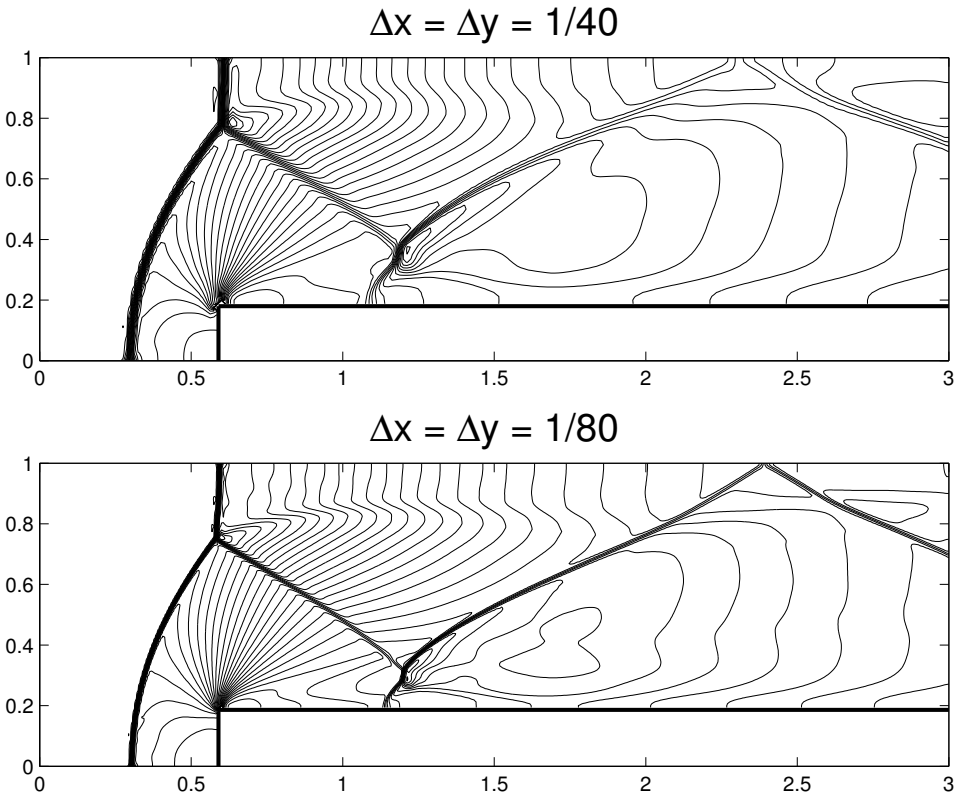

FIGURE 4. Density contours for the forward facing step problem on two different meshes.

The corner of the step is a singularity, which has to be treated carefully in numerical experiments. Unlike in [10] and many other papers, we do not modify our relaxation scheme near the corner. However, we use different

grid refinements to decrease the entropy layer at the downstream bottom wall. In figure 4 we show 30 equi-distributed contour plots of the density at time $t = 4.0$ using two different uniform meshes of 120×40, and 240×80 gridpoints. We can clearly see that the resolution in the solution improves and the artifacts caused by the corner decrease as long as the gridpoints on the mesh increase.

3.2. Three-dimensional problems

3.2.1. Advection problem We consider a three-dimensional linear advection problem introduced and carefully studied in [7]. The deformation flow in this problem is obtained by superimposing deformation in x-y plane with deformation in the x-z plane. The problem statement is

$$\frac{\partial u}{\partial t} + v_1 \frac{\partial u}{\partial x} + v_2 \frac{\partial u}{\partial y} + v_3 \frac{\partial u}{\partial z} = 0, \qquad t \in (0, T], \quad (x, y, z) \in [0, 1]^3, \tag{18}$$

where the velocities are given by

$$\begin{aligned} v_1(x, y, z) &= 2 \sin^2(\pi x) \sin(2\pi y) \sin(2\pi z) g(t), \\ v_1(x, y, z) &= 2 \sin^2(\pi x) \sin(2\pi y) \sin(2\pi z) g(t), \\ v_1(x, y, z) &= 2 \sin^2(\pi x) \sin(2\pi y) \sin(2\pi z) g(t). \end{aligned} \tag{19}$$

The function $g(t)$ is used to introduce time dependence in the flow domain and is defined as

$$g(t) = \cos\Big(\frac{\pi t}{T}\Big), \qquad t \in (0, T].$$

As reported in [7], the flow slows down and reverses direction in such a way that initial condition should be recovered at time T (*i.e.* $u(0, x, y, z) = u(T, x, y, z)$). This is a very useful test example since the analytical solution at time T is known even though the flow structure becomes complicated at this time. Here we use $T = 1.5$, and the discontinuous initial condition

$$u(0, x, y, z) = \begin{cases} 1, & \text{if} \quad x < \frac{1}{2}, \\ 0, & \text{if} \quad x \geq \frac{1}{2}. \end{cases}$$

For this initial data, the interface at $x = \frac{1}{2}$ deforms in a fully three-dimensional way and return to its initial location at time $t = T$. It is easy to verify that the velocity field (19) is divergence free, *i.e.*

$$\frac{\partial v_1}{\partial x} + \frac{\partial v_2}{\partial y} + \frac{\partial v_3}{\partial z} = 0.$$

This condition allows us to write the equation (18) in a conservative form as (1), where the flux functions are taken as

$$F(u) = v_1 u, \qquad G(u) = v_2 u, \qquad \text{and} \qquad H(u) = v_3 u.$$

Therefore, the associated relaxation system to (18) is constructed as (2) with characteristic speeds given by (13). We discretize the flow domain in $50 \times 50 \times 50$ uniform cubes and a $\Delta t = 0.5\Delta x$ is used in our computations. In figure 5 we display the obtained result at times $t = \frac{T}{8}$, $t = \frac{T}{2}$ and $t = T$, respectively. In this figure, the surface plots of the solution are shown only on the subdomain $[0,1]\times[0,1]\times[0,1] \setminus [\frac{1}{2},1]\times[\frac{1}{2},1]\times[0,1]$ for better insight, while the contour plots are taken by cross section at $z = 0.425$ as in [7].

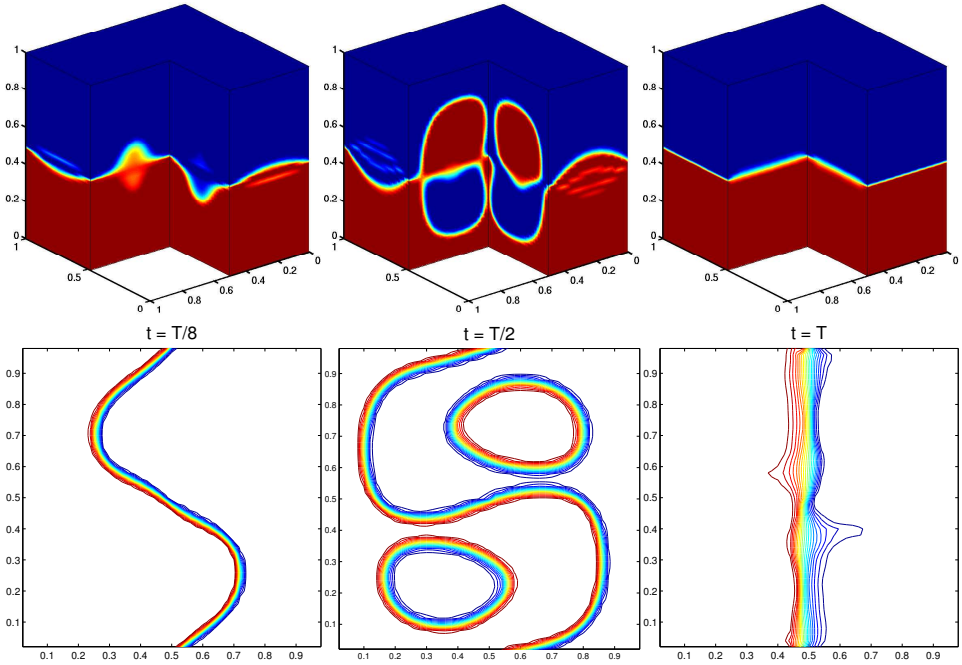

FIGURE 5. Results for the three-dimensional advection equation (18).

At $t = \frac{T}{2}$ the interface appears disconnected and at $t = T$ the initial interface is recovered with an non avoidable smearing introduced by the full three-dimensional deformation. The results shown here agree well with those presented in [7]. we would like to comment on the method used in [7] to solve the equation (18). Thus, the author in [7] proposed a high resolution scheme based on upwind techniques where direct or approximate Riemann solvers are needed. In contrast, our relaxation scheme does not require any Riemann solver and gives results which are comparable to those obtained by upwinding in [7].

3.2.2. Burgers equation the second test example is the three-dimensional Burgers equation

$$\frac{\partial u}{\partial t} + \left(\frac{u^2}{2}\right)_x + \left(\frac{u^2}{2}\right)_y + \left(\frac{u^2}{2}\right)_z = 0, \qquad t > 0, \quad (x,y,z) \in [0,1]^3, \qquad (20)$$

subject to periodic boundary conditions and to the Riemann initial data

$$u(0,x,y,z) = \begin{cases} 1.5, & \text{if} \quad x^2+y^2+z^2 \leq 0.01, \\ 1, & \text{otherwise.} \end{cases} \tag{21}$$

Note that the application of equation (20) to the initial data (21) results in a circular shock centred at the origin of the cube and moving along the main diagonal of the unit cube. The relaxation system that gives at the limit equation (20) is formulated as (2) with

$$F(u) = G(u) = H(u) = \frac{u^2}{2},$$

and characteristic speeds given by (13).

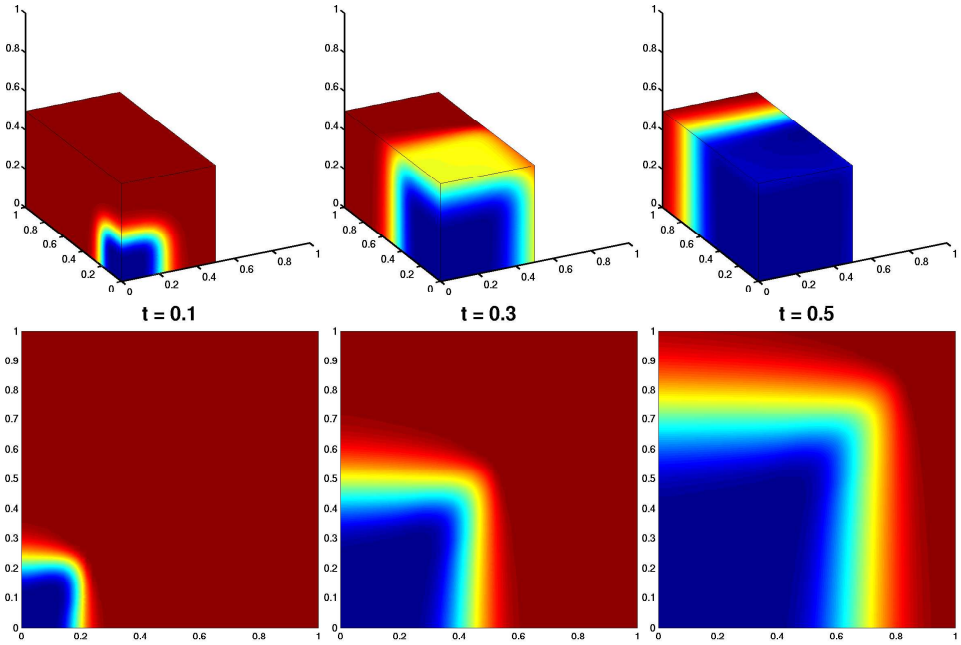

FIGURE 6. Results for the three-dimensional Burgers equation (20).

As in the previous example, we discretize the spatial domain uniformly into $50 \times 50 \times 50$ gridpoints and we set $\Delta t = 0.005$. The obtained results are shown in figure 6 at three different times $t = 0.1$, $t = 0.3$ and $t = 0.5$. Here in the three-dimensional plots only a part of the unit cube, $[0,1] \times [0, \frac{1}{2}] \times [0, \frac{1}{2}]$, is shown. Whereas, the two-dimensional plots represents a cross section at $x = \frac{1}{2}$. The third relaxation scheme captures accurately the evolution of the shock along the main diagonal of the computational domain without diffusing the fronts neither introducing oscillations near steep gradients.

3.2.3. Riemann problem in gas dynamic The three-dimensional system of inviscid Euler equations can be written in conservative form as (1) with

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ E \end{pmatrix}, \qquad \mathbf{F}(\mathbf{U}) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ u(E+p) \end{pmatrix},$$

$$\mathbf{G}(\mathbf{U}) = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho vw \\ v(E+p) \end{pmatrix}, \qquad \mathbf{H}(\mathbf{U}) = \begin{pmatrix} \rho w \\ \rho uw \\ \rho vw \\ \rho w^2 + p \\ w(E+p) \end{pmatrix}.$$

where ρ, $\mathbf{u} = (u, v, w)^T$, p, and E denote respectively the mass density, the flow velocity, the thermal pressure and the total energy. The thermal pressure and total energy and related by the equation of state

$$p = (\gamma - 1)\big(E - \frac{\rho}{2}\mathbf{u}^2\big), \tag{22}$$

The example we consider here, is inspired by the standard one-dimensional Sod tube shock problem [1]. Similar test example but in two dimensions is proposed in [3]. The computational domain is the unit cube $\Omega = [0,1] \times [0,1] \times [0,1]$. To define initial conditions for this problem we first divide the domain into eight equally subcubes as: $\Omega_1 = I_l \times I_l \times I_l$, $\Omega_2 = I_l \times I_r \times I_l$, $\Omega_3 = I_r \times I_l \times I_l$, $\Omega_4 = I_r \times I_r \times I_l$, $\Omega_5 = I_l \times I_l \times I_r$, $\Omega_6 = I_l \times I_r \times I_r$, $\Omega_7 = I_r \times I_l \times I_r$, $\Omega_8 = I_r \times I_r \times I_r$, with $I_l = [0, \frac{1}{2}]$ and $I_r = [\frac{1}{2}, 1]$. Then, velocity is set to $\mathbf{u} = \mathbf{0}$ in Ω, density and pressure are alternated between these subcubes as follows

$$(\rho, p)^T(0, x, y, z) = \begin{cases} (0.1, 0.1)^T & \text{if} \quad (x, y, z) \in \Omega_1 \cup \Omega_4 \cup \Omega_6 \cup \Omega_7, \\ (1, 1)^T & \text{if} \quad (x, y, z) \in \Omega_2 \cup \Omega_3 \cup \Omega_5 \cup \Omega_8. \end{cases}$$

Homogeneous Neumann boundary condition were used, and $\Delta t = 0.001$. The obtained results for the density variable at $t = 0.16$ on a mesh of $50 \times 50 \times 50$ gridpoints are illustrated in figure 7. Here, the three-dimensional surface plots represent the distribution of density on the whole cube Ω and on the part $\Omega \setminus [0, \frac{2}{3}] \times [0, \frac{1}{4}] \times [0, 1]$ of the cube, while the two-dimensional plot represents the projection of density on the x-y plane at $z = \frac{1}{2}$. Our relaxation scheme performs well for this test problem and high resolution of the scheme is clearly visible even on coarse mesh as the one we used.

4. Concluding remarks

Relaxation schemes of first and second order accuracy were introduced in [4]. In this paper we have reconstructed high order relaxation schemes

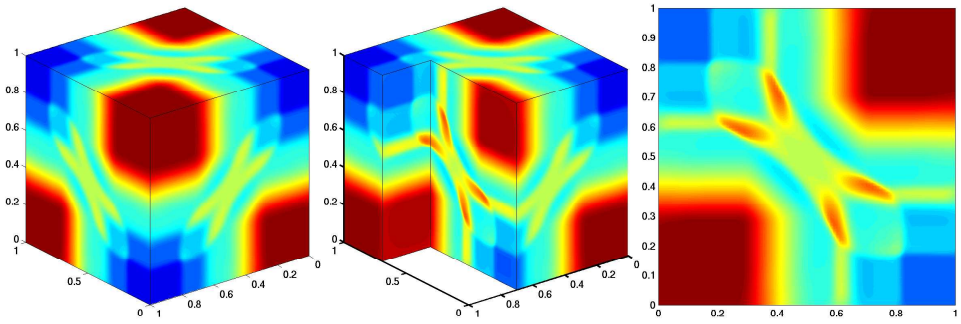

FIGURE 7. Results for the three-dimensional inviscid Euler equations.

by using WENO ideas and a class of TVD high order Runge-Kutta time integration methods. We have generalized the relaxation method for multi-dimensional hyperbolic systems of conservation laws. This procedure combines the attractive attributes of the two methods to yield a procedure for either scalar or system of hyperbolic equations. The new method retains all the attractive features of central schemes such as neither Riemann solvers nor characteristic decomposition are needed. Furthermore, the scheme does not require either nonlinear solution or special front tracking techniques.

The third-order relaxation method have been tested on Burgers equations and systems of inviscid gas Euler in two and three space dimensions. The obtained results indicate good shock resolution with high accuracy in smooth regions and without any nonphysical oscillations near the shock areas.

Acknowledgements: This work was supported by Deutsche Forschungsgemeinschaft (DFG) under grant KL 1105/9-1 .

References

[1] D. Aregba-Driollet and R. Natalini, "Discrete Kinetic Schemes for Multidimensional Systems of Conservation Laws", *SIAM. J. Numer. Anal.* **37** (2000) 1973–2004.

[2] U. Asher, S. Ruuth and R. Spiteri, "Implicit-Explicit Runge-Kutta Methods for Time-dependent Partial Differential Equations", *Appl. Numer. Math.* **25** (1997) 151–167.

[3] M.K. Banda and M. Seaïd, "A Class of the Relaxation Schemes for Two-Dimensional Euler Systems of Gas Dynamics", *Lecture Notes in Computer Science* **2329** (2002) 930–939.

[4] S. Jin and Z. Xin, "The Relaxation Schemes for Systems of Conservation Laws in Arbitrary Space Dimensions", *Comm. Pure Appl. Math.* **48** (1995) 235–276.

[5] A. Kurganov and D. Levy, "A Third-Order Semi-Discrete Central Scheme for Conservation Laws and Convection-Diffusion Equations", *SIAM J. Sci. Comp.* **22** (2000) 1461–1488.

[6] C. Lattanzio and D. Serre, "Convergence of a relaxation scheme for hyperbolic systems of conservation laws", *Numer. Math.* **2001** (88) 121–134.

[7] J. LeVeque Randall, "High-Resolution Conservative Algorithms for Advection in Incompressible Flow", *SIAM J. Numer. Anal.* **33** (1996) 627–665.

[8] R. Natalini, "Convergence to equilibrium for relaxation approximations of conservation laws", *Comm. Pure Appl. Math.* **49** (1996) 795–823.

[9] L. Pareschi and G. Russo, "Implicit-Explicit Runge-Kutta Schemes for Stiff Systems of Differential Equations", *Recent Trends in Numerical Analysis* **3** (2000) 269–289.

[10] P. Woodward and P. Colella, "The Numerical Simulation of Two-Dimensional Fluid Flow with Strong Shocks", *J. Comp. Phys.* **54** (1984) 115–173.