Compact Central WENO Schemes for Multidimensional Conservation Laws

Doron Levy[†] Gabriella Puppo[‡] Giovanni Russo[§]

February 11, 2000

Abstract

We present a new third-order central scheme for approximating solutions of systems of conservation laws in one and two space dimensions. In the spirit of Godunov-type schemes, our method is based on reconstructing a piecewise-polynomial interpolant from cell-averages which is then advanced exactly in time.

In the reconstruction step, we introduce a new third-order, *compact*, CWENO reconstruction, which is written as a convex combination of interpolants based on different stencils. The heart of the matter is that one of these interpolants is taken as a suitable quadratic polynomial and the weights of the convex combination are set as to obtain third-order accuracy in smooth regions. The embedded mechanism in the WENO-like schemes guarantees that in regions with discontinuities or large gradients, there is an automatic switch to a one-sided second-order reconstruction, which prevents the creation of spurious oscillations.

In the one-dimensional case, our new third order reconstruction is based on an extremely compact *three* point stencil. Analogous compactness is retained in more space dimensions. The accuracy, robustness and high-resolution properties of our scheme are demonstrated in a variety of one and two dimensional problems.

Key words. Hyperbolic systems, central difference schemes, high-order accuracy, non-oscillatory schemes, WENO reconstruction, CWENO reconstruction.

AMS(MOS) subject classification. Primary 65M10; secondary 65M05.

1 Introduction

We are concerned with multidimensional systems of hyperbolic conservation laws of the form

$$u_t + \nabla_x \cdot f(u) = 0, \qquad x \in \mathbb{R}^d, \quad u = (u_1, \dots, u_n). \tag{1.1}$$

[†]Department of Mathematics, University of California, Berkeley, CA 94720, and Lawrence Berkeley National Lab; dlevy@math.berkeley.edu

[‡]Dipartimento di Matematica, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy; puppo@calvino.polito.it

[§]Dipartimento di Matematica, Università dell'Aquila, Via Vetoio, loc. Coppito - 67100 L'Aquila, Italy; russo@univaq.it

Methods for approximating solutions to equation (1.1) have attracted a lot of attention in recent years (see [5], [11], [25] and the references therein).

In this work we focus on Godunov-type schemes, where one first reconstructs a piecewise-polynomial interpolant which is then advanced exactly in time according to (1.1) and finally projected on its cell-averages. Generally, one can divide Godunov-type schemes into two sub-classes - *upwind methods* and *central methods*.

In *upwind schemes*, one first reconstructs a polynomial in every cell, which is then used to compute a new cell average in the same location in the next time step. This procedure requires solving Riemann problems at the discontinuous interfaces. For high-order methods, instead of analytically solving the resulting Riemann problems, one typically implements approximate Riemann solvers or some form of flux splitting. For systems of conservation laws, or in the demanding context of more space dimensions, this procedure turns out to be more intricate as such Riemann solvers do not exist. The essentially non-oscillatory (ENO) methods by Harten are the prototype of high-order methods (see [6], [23] and the references therein). A recent review of ENO and WENO methods can be found in [22].

Central schemes, on the other hand, are based on averaging over the Riemann fans, a procedure which is typically done by staggering between two grids. They require no Riemann solvers, no projection along characteristic directions and no flux splitting. Therefore, all that one has to do in order to solve a problem is to supply the flux function. They are more simple when compared with upwind schemes.

The major difference between different central methods is in the reconstruction step, where one computes a piecewise-polynomial interpolant from the previously computed cell-averages.

The prototype of central schemes is the Lax-Friedrichs [4] scheme which is based on a piecewise-constant interpolant. Even though it is very robust, it is only first-order accurate and, moreover, it suffers from excessive numerical dissipation. A second-order central method was proposed by Nessyahu and Tadmor in [20]. This method is based on a MUSCL-like [10] piecewise linear interpolant and nonlinear limiters which prevent spurious oscillations (see [21] for a different approach). A variety of extensions to these methods were suggested. The one-dimensional third-order method of Liu and Tadmor [19] is based on the third-order reconstruction by Liu and Osher in [17]. For the two-dimensional method see [1] and [8].

A first step for importing the high-order reconstructions that were derived in the upwind framework was taken in [2]. There, the ENO method was transformed into the central setup, and a new *mostly centered stencil* was shown to produce the least oscillatory results. The next step was taken in the 1D case in [13], where a new *central weighted non-oscillatory* (CWENO) reconstruction was introduced. This CWENO method is based on the upwind WENO methods by [18] and [7], in which an interpolant is written as a convex combination of several reconstructions which are based on different stencils. These methods include an internal switch which is designed such as to provide the maximum possible accuracy in smooth regions, while automatically switching to a more robust one-sided stencil in the presence of discontinuities and large gradients. The 1D CWENO method was extended to two space dimensions in [15] and [16]. For a numerical study of the behavior of the total variation for the CWENO scheme, see [14].

In this paper we present a new, compact CWENO reconstruction. This new reconstruction is based on defining a suitable quadratic function which is added to linear interpolants in such a way as to obtain third-order accuracy in smooth regions (in one and two space dimensions). In regions with discontinuities or large gradients, the weights are automatically changed so that they switch to a one-sided second-order linear reconstruction. This reconstruction turns out to be extremely compact; in the one dimensional case, e.g., the reconstruction is based on a three-point stencil.

The structure of this paper is as follows: We start in §2 with a brief overview of central schemes for conservation laws in one and two space dimensions.

We then proceed to present our new, compact, third-order CWENO reconstruction in §3. The idea of introducing a quadratic reconstruction is first presented in the one dimensional framework and then extended to two space dimensions.

We end in §4 where we demonstrate our new method in several test cases. First, the accuracy tests (both in one-dimensional and two-dimensional cases) show the third-order accuracy of the method. We then solve the one-dimensional system of the Euler equations of gas dynamics for a few test problems and we illustrate the behavior of the scheme in scalar two-dimensional cases. In particular, we would like to stress that in our numerical results we observe a very robust and non-oscillatory behavior of the weights, which can be related to the overall robustness and accuracy properties of our new method.

Acknowledgment: The work of D.L. was supported in part by the Applied Mathematical Sciences subprogram of the Office of Energy Research, U.S. Department of Energy, under contract DE–AC03–76–SF00098. The work of G.P. was partly supported by the MURST-Cofin'98 program. Part of this work was done while G.P. and G.R. were visiting the Lawrence Berkeley Lab.

2 Central Schemes for Conservation Laws

In this section we give a brief overview of central schemes for approximating solutions to hyperbolic conservation laws in one and two space dimensions. For further details we refer the reader to [25], [8], [13] and the references therein.

Starting in the one-dimensional case, we seek numerical solutions of the Cauchy problem

$$\begin{cases} u_t + f(u)_x = 0, \\ u(x, t{=}0) = u_0(x). \end{cases} \tag{2.1}$$

For simplicity, we introduce a uniformly spaced grid in the (x,t) space, where the mesh spacings are denoted by $h := \Delta x$ and $k := \Delta t$, respectively. We denote by $\bar{u}_j^n$, the numerical approximation of the cell average in the cell $I_j := [x_{j-1/2}, x_{j+1/2}]$ at time $t^n = nk$, where $x_j = jh$. The finite-difference method will approximate the cell-averages at time t^{n+1} based on their values at time t^n.

We start by reconstructing at time t^n a piecewise-polynomial conservative interpolant from the known cell-averages, $\bar{u}_j^n$, i.e.,

$$P_u(x,t^n) := \sum_j R_j(x)\chi_j, \tag{2.2}$$

where χ_j is the characteristic function of the interval I_j and $R_j(x)$ is a polynomial defined in I_j. It is here, in the reconstruction step, where the accuracy and non-oscillatory requirements enter. Different central methods will be typically based on different reconstructions.

The reconstruction, $P_u(x,t^n)$, is then evolved exactly in time (integrating (2.1)), and then projected on staggered cells, in order to compute the cell average at $I_{j+1/2}$. With this procedure we obtain

$$\bar{u}_{j+1/2}^{n+1} = \bar{u}_{j+1/2}^n + \frac{1}{h}\int_{t^n}^{t^{n+1}} [f(P_u(x_j,\tau)) - f(P_u(x_{j+1},\tau))]\, d\tau. \tag{2.3}$$

Based on the reconstruction, (2.2), the first term on the RHS of (2.3) can be explicitly computed,

$$\bar{u}_{j+1/2}^n = \frac{1}{h}\int_{x_j}^{x_{j+1}} P_u(x,t^n)\, dx.$$

In order to compute the second integral on the RHS of (2.3), namely the integral in time over the fluxes, one should observe that due to the staggering, up to a suitable CFL condition, these integrals involve only smooth functions, and hence can be approximated by a sufficiently smooth quadrature. A second-order method can be obtained, e.g., using the mid-point rule in time (see [20]). Simpson's rule for the quadrature in time will provide fourth order accuracy (which will naturally be sufficient also for a third-order method).

The quadratures for the time integrals of the fluxes, require the *prediction* of point-values of the function in several points in the interval $[t^n, t^{n+1}]$. One possible approach is to use a Taylor expansion based on the equation, (2.1). Such an approach was used, e.g., in [20] and [19]. In order to avoid the technical complications involved in the Taylor series (in particular with high-order methods, and when dealing with systems of equations), one can alternatively use a Runge-Kutta (RK) method directly on (2.1), to predict the required values in later times. By using the *Natural Continuous Extension* (NCE) of Runge-Kutta schemes [26], the value of the flux at the nodes of the quadrature formula can be computed with a single RK step. Such a method is simpler compared with the Taylor expansion method, but it does require another reconstruction: the reconstruction of the point values of the derivatives of the fluxes at time t^n which are then used as the input to the RK solver (see [2] and [13] for more details).

The simplicity of central schemes manifests itself when turning to deal with systems of equations. Basically, the algorithm that was described in the scalar case, repeats itself component-wise. Based on the type of the reconstruction, when solving systems of equations, there can even be simplifications over a purely componentwise extension of the scalar scheme. These can be found, e.g., in §3.3 below.

The extension to two space dimensions is straightforward; We consider

$$u_t + f(u)_x + g(u)_y = 0, \tag{2.4}$$

subject to the initial condition, $u(x, y, t=0) = u_0(x, y)$. Here, Δx and Δy will denote the spatial mesh spacings, while Δt denotes the spacing in time. The two-dimensional cells are now $I_{i,j} = [x_{i-1/2}, x_{i+1/2}] \times [y_{j-1/2}, y_{j+1/2}]$.

Following the general methodology, we wish to construct the cell-averages, $\bar{u}_{j,k}^{n+1}$, based on the cell averages at time t^n. First, we construct a two-dimensional interpolant which reads

$$P_u(x, y, t^n) = \sum_{i,j} R_{i,j}(x, y)\chi_{i,j}, \tag{2.5}$$

with $\chi_{i,j}$ being the characteristic function of the cell $I_{i,j}$ and $R_{i,j}(x, y)$ a polynomial of a suitable degree. An exact evolution in time of the interpolant (2.5) which is projected on its cell-averages now reads (compare with (2.3))

$$\begin{aligned} \bar{u}_{i+1/2,j+1/2}^{n+1} &= \bar{u}_{i+1/2,j+1/2}^{n} + \\ &+\frac{1}{\Delta x \Delta y}\int_{t^n}^{t^{n+1}}\int_{y=y_j}^{y_{j+1}} [f(P_u(x_i, y, \tau)) - f(P_u(x_{i+1}, y, \tau)]\, dyd\tau + \\ &+\frac{1}{\Delta x \Delta y}\int_{t^n}^{t^{n+1}}\int_{x=x_i}^{x_{i+1}} [f(P_u(x, y_j, \tau)) - f(P_u(x, y_{j+1}, \tau)]\, dxd\tau. \end{aligned} \tag{2.6}$$

The staggered cell-average at time t^n can be directly computed by

$$\bar{u}_{i+1/2,j+1/2}^{n} = \frac{1}{\Delta x \Delta y}\int_{x_i}^{x_{i+1}}\int_{y_j}^{y_{j+1}} P_u(x, y, t^n)\, dydx.$$

Analogously to the one-dimensional case, the flux integrals on the RHS of (2.6) can be approximated using a quadrature rule in time. This can be coupled with a quadrature rule for the line integrals in space. Here it is necessary to avoid quadrature points at which the fluxes may not be smooth. The details can be found, e.g., in [15, 16] (see also [8]).

3 A Compact third-order CWENO Reconstruction

3.1 The One-Dimensional Framework

In this section we derive our new CWENO reconstruction in one space dimension. The two dimensional extension will follow in §3.2.

We first note that in the absence of large gradients, we obtain third order accuracy if we choose for the reconstruction the *optimal* polynomial

$$P_j(x) = P_{\text{OPT},j}(x),$$

where $P_{\text{OPT},j}(x)$ is the parabola that interpolates the data $\bar{u}^n_{j-1}, \bar{u}^n_j, \bar{u}^n_{j+1}$ in the sense of cell averages to enforce conservation:

$$\int_{x_{j+l-1/2}}^{x_{j+l+1/2}} P_{\text{OPT},j}(x)\, dx = \bar{u}^n_{j+l}, \qquad l = -1, 0, 1.$$

These conditions determine $P_{\text{OPT},j}(x)$ completely, namely:

$$P_{\text{OPT},j}(x) = u^n_j + u'_j(x - x_j) + \frac{1}{2} u''_j (x - x_j)^2, \tag{3.1}$$

with

$$\begin{aligned} u^n_j &= \bar{u}^n_j - \frac{1}{24}(\bar{u}^n_{j+1} - 2\bar{u}^n_j + \bar{u}^n_{j-1}), \\ u'_j &= \frac{\bar{u}^n_{j+1} - \bar{u}^n_{j-1}}{2\Delta x}, \qquad u''_j = \frac{\bar{u}^n_{j-1} - 2\bar{u}^n_j + \bar{u}^n_{j+1}}{\Delta x^2}. \end{aligned}$$

However, when discontinuities or large gradients occur, this reconstruction would be oscillatory. Therefore following the WENO methodology ([18], [7], [13]), we construct an essentially non-oscillatory interpolant as a convex combination of polynomials which are based on different stencils. Specifically, in the cell I_j we write

$$P_j(x) = \sum_i w^j_i P^j_i(x), \quad \sum_i w^j_i = 1, \quad w_i \geq 0, \quad i \in \{\text{L}, \text{C}, \text{R}\}, \tag{3.2}$$

where P_{L} and P_{R} are linear functions based on a left stencil and a right stencil, respectively, and P_{C} is a quadratic polynomial. In order to simplify the notations, we will omit the upper index j, remembering that the weights and the three polynomials change from cell to cell.

Conservation requires that $P_{\text{R}}(x)$ will interpolate the cell averages

$$\int_{x_{j-1/2}}^{x_{j+1/2}} P_{\text{R}}(x) dx = \Delta x\, \bar{u}^n_j, \qquad \int_{x_{j+1/2}}^{x_{j+3/2}} P_{\text{R}}(x) dx = \Delta x\, \bar{u}^n_{j+1},$$

which in turn, results with

$$P_{\text{R}}(x) = \bar{u}^n_j + \frac{\bar{u}^n_{j+1} - \bar{u}^n_j}{\Delta x}(x - x_j). \tag{3.3}$$

Similarly, for the left interpolant we have

$$P_{\text{L}}(x) = \bar{u}^n_j + \frac{\bar{u}^n_j - \bar{u}^n_{j-1}}{\Delta x}(x - x_j). \tag{3.4}$$

All that is left is to reconstruct a centered polynomial, P_{C}, such that the convex combination, (3.2), will be third-order accurate in smooth regions. It must, therefore, satisfy

$$P_{\text{OPT}}(x) = C_{\text{L}} P_{\text{L}}(x) + C_{\text{R}} P_{\text{R}}(x) + C_{\text{C}} P_{\text{C}}(x), \quad \sum_i C_i = 1, \quad i \in \{\text{L}, \text{C}, \text{R}\}, \tag{3.5}$$

where $C_{\text{L}}, C_{\text{C}}$ and C_{R} are constants. Due to the staggering between every two consecutive steps of the central method, our reconstruction should provide half-cell averages which are third-order accurate. A straightforward calculation shows that *any* symmetric choice of constants C_i in (3.5) provides the desired accuracy. In particular, for the specific choice of $C_{\rm L}=C_{\rm R}=1/4$, equations (3.3)–(3.5) yield

$$\begin{aligned} P_{\rm C}(x) &= 2P_{\rm OPT}(x)-\frac{1}{2}(P_{\rm R}(x)+P_{\rm L}(x)) = \bar{u}_j^n-\frac{1}{12}(\bar{u}_{j+1}^n-2\bar{u}_j^n+\bar{u}_{j-1}^n)+\\ &\quad+\frac{\bar{u}_{j+1}^n-\bar{u}_{j-1}^n}{2\Delta x}(x-x_j)+\frac{\bar{u}_{j+1}^n-2\bar{u}_j^n+\bar{u}_{j-1}^n}{\Delta x^2}(x-x_j)^2. \end{aligned} \tag{3.6}$$

In order to complete the reconstruction of $P_j(x)$ in (3.2), it is left to compute the weights w_i. Following [7], [13], we write

$$w_i=\frac{\alpha_i}{\sum_k \alpha_k}, \qquad \alpha_i=\frac{C_i}{(\varepsilon+IS_i)^p}, \qquad i,k\in\{\rm L,C,R\}. \tag{3.7}$$

The constants C_i's in (3.7) are chosen to be the same as in (3.5), i.e., $C_{\rm L}=C_{\rm R}=1/4$, while $C_{\rm C}=1/2$.

The *smoothness indicators*, IS_i, are responsible for detecting large gradients or discontinuities and to automatically switch to the stencil that generates the least oscillatory reconstruction in such cases. Once again, we follow [7], [13] and define in each cell I_j the three smoothness indicators, IS_i, as

$$IS_i=\sum_{l=1}^{2}\int_{x_{j-1/2}}^{x_{j+1/2}} h^{2l-1}(P_i^{(l)}(x))^2dx. \qquad i\in\{\rm L,C,R\}. \tag{3.8}$$

A direct computation, based on (3.3), (3.4) and (3.6), yields

$$\begin{aligned} IS_{\rm L} &= (\bar{u}_j^n-\bar{u}_{j-1}^n)^2, \qquad IS_{\rm R}=(\bar{u}_{j+1}^n-\bar{u}_j^n)^2,\\ IS_{\rm C} &= \frac{13}{3}(\bar{u}_{j+1}^n-2\bar{u}_j^n+\bar{u}_{j-1}^n)^2+\frac{1}{4}(\bar{u}_{j+1}^n-\bar{u}_{j-1}^n)^2. \end{aligned} \tag{3.9}$$

The constant ε is taken to prevent the denominator from vanishing. Furthermore, its value has to satisfy two requirements, namely

i) $\varepsilon+h^2u_x^2 \gg |IS-h^2u_x^2|$ in smooth regions

ii) $\varepsilon \ll IS$ near discontinuities

The first condition guarantees that, on smooth regions, the weights are basically equal to the constants that provide high accuracy. The second condition guarantees that in the presence of a discontinuity, the weights of the parabola and of one of the one-sided linear reconstructions will be practically zero, and we will be left with the other one-sided linear reconstruction. Hence, our third-order method automatically switches to a second-order method in the presence of large gradients, which is exactly what makes it so robust as will be evident in our numerical computations presented below.

The constant p weights the departure from smoothness. We used the value $p=2$. For a discussion about its choice see [7] and [13].

A second, non-oscillatory reconstruction is required for the flux derivative. It is natural to adapt the reconstruction that we used for the half-cell averages also to the reconstruction of the point-values of the flux derivative. Here however the interpolation requirements will be in the sense of point values instead of cell averages. Once again, a direct computation shows that any symmetric choice of constants will provide the desired accuracy and it will only be natural to use the same constants, C_i's, that were used in (3.5). This is simpler than the case of [13] where we had to use different sets of constants for the two different reconstructions (the reconstruction of the half cell averages, and the reconstruction of the point-values of the flux derivative at the center of the cells).

Remarks:

1. We would like to emphasize that our new method is based on adding the parabola P_{C} into the convex combination which is the heart of our non-oscillatory reconstruction. Our numerical simulations showed that the freedom we have in selecting the constants C_i has no influence on the properties of the method. It is easy to prove that we obtain third-order accuracy regardless of the smoothness of the weights, as long as they are symmetric. Of course this is mainly of theoretical interest, since it is very unlikely that only a smooth departure from symmetry is maintained if the weights are not smooth.

2. By extending our new ideas, one can modify our previous CWENO method presented in [13] in order to obtain a fifth-order, central, non-oscillatory scheme. This can be done by simply adding a fourth-order polynomial for the computation of the point-values.

3. Our new reconstruction is equivalent to limiting with the minimum slope instead of slope zero in the presence of a discontinuity. Hence, it is based on a second-order reconstruction close to shocks, unlike the scheme of [19] which can be only first order accurate in such regions.

4. In the one dimensional case, the additional parabola is needed only for the accurate recovery of the point-values. In the 2D framework, however, the equivalent additional parabola will be required also for the accurate reconstruction of the fractional cell-averages.

3.2 A Two-Dimensional Extension

We extend the ideas of §3.1 to the two-dimensional framework. This extension is straightforward and is based on reconstructing an interpolant as a convex combination of four one-sided, piecewise-linear interpolants, and a centered, quadratic interpolant, in order to get the desired third-order accuracy in smooth regions.

Following these ideas, the reconstruction in the cell I_{ij}, can be written as

$$P_{i,j}(x,y) = \sum_k w_k^{i,j} P_k^{i,j}(x,y), \qquad k \in \{\text{NE}, \text{NW}, \text{SE}, \text{SW}, \text{C}\}, \tag{3.10}$$

with $\sum_k w_k^{i,j} = 1$, and $w_k^{i,j} \geq 0$. Here, P_{NE}, P_{NW}, P_{SE} and P_{SW} are the one-sided linear reconstructions, and P_{C} is a centered quadratic reconstruction (see Figure 3.1). As in the one-dimensional case, we will simplify our notations by omitting the superscripts, remembering that both the weights and the polynomials change from cell to cell.

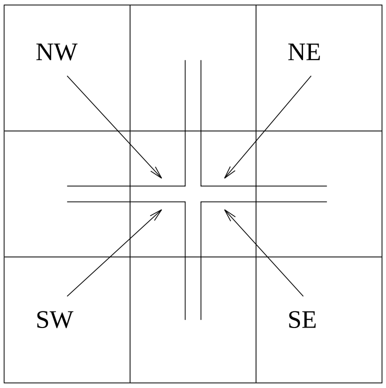

Figure 3.1: The Two-Dimensional Stencil

Clearly, in analogy with the one-dimensional case, (3.3–3.4), the four linear reconstructions are given by

$$
\begin{aligned}
P_{\text{NE}}(x,y) &= \bar{u}^n_{i,j} + \frac{\bar{u}^n_{i+1,j}-\bar{u}^n_{i,j}}{\Delta x}(x-x_i) + \frac{\bar{u}^n_{i,j+1}-\bar{u}^n_{i,j}}{\Delta y}(y-y_j),\\
P_{\text{NW}}(x,y) &= \bar{u}^n_{i,j} + \frac{\bar{u}^n_{i,j}-\bar{u}^n_{i-1,j}}{\Delta x}(x-x_i) + \frac{\bar{u}^n_{i,j+1}-\bar{u}^n_{i,j}}{\Delta y}(y-y_j),\\
P_{\text{SW}}(x,y) &= \bar{u}^n_{i,j} + \frac{\bar{u}^n_{i,j}-\bar{u}^n_{i-1,j}}{\Delta x}(x-x_i) + \frac{\bar{u}^n_{i,j}-\bar{u}^n_{i,j-1}}{\Delta y}(y-y_j),\\
P_{\text{SE}}(x,y) &= \bar{u}^n_{i,j} + \frac{\bar{u}^n_{i+1,j}-\bar{u}^n_{i,j}}{\Delta x}(x-x_i) + \frac{\bar{u}^n_{i,j}-\bar{u}^n_{i,j-1}}{\Delta y}(y-y_j).
\end{aligned}
\tag{3.11}
$$

The centered polynomial, $P_{\text{C}}(x,y)$, is taken such as to satisfy

$$
P_{\text{OPT}}(x,y) = \sum_k C_k P_k(x,y), \qquad \sum_k C_k = 1, \qquad k \in \{\text{NE}, \text{NW}, \text{SE}, \text{SW}, \text{C}\}. \tag{3.12}
$$

Here, P_{OPT} is the quadratic polynomial based on a nine-point stencil, centered around $I_{i,j}$, which is given by (see [12])

$$
\begin{aligned}
P_{\text{OPT}}(x,y) &= u^n_{i,j} + u'_{i,j}(x-x_i) + u^{\backprime}_{i,j}(y-y_j) + u^{\prime\backprime}_{i,j}(x-x_i)(y-y_j) + \\
&\quad + \frac{1}{2}u''_{i,j}(x-x_i)^2 + \frac{1}{2}u^{\backprime\backprime}_{i,j}(y-y_j)^2,
\end{aligned}
\tag{3.13}
$$

where

$$
\begin{aligned}
u^n_{i,j} &= \bar{u}_{i,j} - \frac{1}{24}\left((\Delta x)^2 u''_{i,j} + (\Delta y)^2 u^{\backprime\backprime}_{i,j}\right),\\
u'_{i,j} &= \frac{\bar{u}^n_{i+1,j}-\bar{u}^n_{i-1,j}}{2\Delta x}, \qquad u^{\backprime}_{i,j} = \frac{\bar{u}^n_{i,j+1}-\bar{u}^n_{i,j-1}}{2\Delta y},\\
u''_{i,j} &= \frac{\bar{u}^n_{i+1,j}-2\bar{u}^n_{i,j}+\bar{u}^n_{i-1,j}}{\Delta x^2}, \qquad u^{\backprime\backprime}_{i,j} = \frac{\bar{u}^n_{i,j+1}-2\bar{u}^n_{i,j}+\bar{u}^n_{i,j-1}}{\Delta y^2},\\
u^{\prime\backprime}_{i,j} &= \frac{\bar{u}^n_{i+1,j+1}+\bar{u}^n_{i-1,j-1}-\bar{u}^n_{i+1,j-1}-\bar{u}^n_{i-1,j+1}}{4\Delta x\Delta y}.
\end{aligned}
\tag{3.14}
$$

Unlike the one-dimensional case, not every symmetric selection of the constants C_k's will provide a third-order reconstruction for the quarter cell-averages. Here, a straightforward computation shows that in order to satisfy the accuracy requirements, we must take $C_{\text{NE}} = C_{\text{NW}} = C_{\text{SW}} = C_{\text{SE}} = 1/8$. Hence, $C_{\text{C}} = 1/2$, and (3.12) implies

$$\begin{aligned} P_{\text{C}}(x,y) &= 2P_{\text{OPT}}(x,y) - \frac{1}{4}\Big[P_{\text{NE}}(x,y) + P_{\text{NW}}(x,y) + P_{\text{SW}}(x,y) + P_{\text{SE}}(x,y)\Big] = \\ &= u^n_{i,j} + u'_{i,j}(x-x_i) + u^{\backprime}_{i,j}(y-y_j) + 2u^{\prime\backprime}_{i,j}(x-x_i)(y-y_j) + \\ &\quad + u''_{i,j}(x-x_i)^2 + u^{\backprime\backprime}_{i,j}(y-y_j)^2, \end{aligned} \tag{3.15}$$

where for this formula:

$$u^n_{i,j} = \bar{u}^n_{i,j} - \frac{1}{12}\Big[(\Delta x)^2 u''_{i,j} + (\Delta y)^2 u^{\backprime\backprime}_{i,j}\Big].$$

All that remains is to determine the weights $w^{i,j}_k$ in (3.10). Once again, we write

$$w^{i,j}_k = \frac{\alpha^{i,j}_k}{\sum_l \alpha^{i,j}_l}, \qquad \alpha^{i,j}_k = \frac{C^{i,j}_k}{(\varepsilon + IS^{i,j}_k)^p}, \qquad k,l \in \{\text{NE}, \text{NW}, \text{SE}, \text{SW}, \text{C}\}.$$

The constants C_k are the same constants that were used to reconstruct the centered parabola in (3.12). The constants ε and p play the same role as in the one-dimensional case. At this point, to simplify the notations we assume that the mesh spacings are equal in the x and y directions, i.e., $\Delta x = \Delta y = h$. We can then follow [15], and define the smoothness indicators, $IS^{i,j}_k$, as

$$IS^{i,j}_k = \sum_{|\alpha|=1,2} \int_{x_{i-h/2}}^{x_{i+h/2}} \int_{y_{j-h/2}}^{y_{j+h/2}} h^{2(|\alpha|-1)} (D^\alpha P_k)^2, \qquad k \in \{\text{NE}, \text{NW}, \text{SE}, \text{SW}, \text{C}\}. \tag{3.16}$$

If $\Delta x \neq \Delta y$, then only a trivial enhancement to (3.16) is required. For the four one-sided linear reconstructions, which can be all written as

$$P_k = \hat{u}_k + \hat{u}'_k(x-x_i) + \hat{u}^{\backprime}_k(y-y_j), \qquad k \in \{\text{NE}, \text{NW}, \text{SE}, \text{SW}\}.$$

with suitable reconstructed point-values and first derivatives, a direct computation of (3.16) results with

$$IS_k = h^2[(\hat{u}')^2 + (\hat{u}^{\backprime})^2]. \tag{3.17}$$

The centered smoothness indicator, IS_{C}, which corresponds to the centered quadratic reconstruction, $P_c(x,y)$, (3.15), is given by

$$IS_{\text{C}} = h^2\left[(u')^2 + (u^{\backprime})^2\right] + \frac{h^4}{3}\left[13(u'')^2 + 14(u^{\prime\backprime})^2 + 13(u^{\backprime\backprime})^2\right],$$

(the discrete derivatives are given by (3.14)).

3.3 A Note on Systems

Almost nothing changes when turning to deal with systems. The reconstruction that was described in the previous sections, directly transforms to systems and is performed component-wise. The only relatively subtle issue is the computation of the smoothness indicators.

In our previous work [13], we have suggested several approaches for the computation of the smoothness indicators. Three different options were suggested: the first is to allow every component to have a strictly individual behavior, namely to allow a different stencil with different smoothness indicators to each component. In the second approach, we designed *global smoothness indicators* such as to force every component to adjust even when the discontinuity is in a different component. The last approach was to use external information about the system. For example, in the Euler equations of gas dynamics, one expects both shocks and contact discontinuities to occur in the density. Hence, all stencils can be tuned according to this component.

Our results in [13] showed that the best approach is the one which was based on the *global smoothness indicators*. It requires no additional information on the system, it produced less oscillatory results compared with the individual smoothness indicators for each component, and it was the simplest to implement.

In the one-dimensional case, e.g., the global smoothness indicators are given by (compare with (3.8))

$$IS_k = \frac{1}{d}\sum_{r=1}^{d} \frac{1}{\|\bar{u}_r\|_2}\left(\sum_{l=1}^{2}\int_{x_{j-1/2}}^{x_{j+1/2}} h^{2l-1}\left(P_{k,r}^{(l)}\right)^2\,dx\right), \quad k \in \{\textsc{l},\textsc{c},\textsc{r}\} \tag{3.18}$$

Here the k-th polynomial in the r-th component is denoted by $P_{k,r}$, and d is the number of equations. The scaling factor $\|\bar{u}_r\|_2$ is defined as the L^2 norm of the cell averages of the r-th component of u,

$$\|\bar{u}_r\|_2 = \left(\sum_{\text{all}\, j} |\bar{u}_{j,r}|^2 h\right)^{1/2}.$$

The numerical examples performed for systems, appearing in §4, were carried out with the global smoothness indicators.

4 Examples

We present numerical tests in one and two space dimensions, in which we demonstrate the accuracy, robustness and high-resolution properties of our new method.

Following our previous works ([2], [13], [15] and [16]), in all our numerical examples we integrate in time using a two-step Runge-Kutta method with natural continuous extension, which was presented in [26], with a fixed time step.

The time step is determined by imposing that the Courant number is a given fraction of the maximum Courant number determined by linear stability analysis. The Courant number is defined by

$$C = \frac{\lambda}{\max_j \rho_j}$$

where ρ_j denotes the spectral radius of the matrix $f'(u_j)$ computed on the initial condition, and $\lambda = \Delta t/\Delta x$ is the mesh ratio.

The linear stability analysis carried out in [3] yields a Courant number $C = 3/7$ for the one dimensional case. We remark that the stencil used in [3] was different than the one that we use, and therefore linear stability should be repeated for the compact scheme in order to obtain a sharp estimate on the maximum Courant number.

Example 1: Accuracy Tests

Our first example checks the accuracy of our new method in several one and two-dimensional test cases. In all of the one-dimensional tables, the norms of the errors are given by

$$\begin{array}{ll} L^1 - \text{error}: & ||\text{Error}||_1 = \sum_{j=1}^{N} |u(x_j, t^n) - u_j^n| h, \\ L^\infty - \text{error}: & ||\text{Error}||_\infty = \max_{1\le j\le N} |u(x_j, t^n) - u_j^n|. \end{array}$$

Analogous expressions hold for the two-dimensional norms.

1. **Linear advection:** This test estimates the convergence rate at large times. We solve $u_t + u_x = 0$, subject to the initial data $u(x, t\!=\!0) = \sin(\pi x)$ and to periodic boundary conditions on $[-1, 1]$. The integration time was taken as $T = 10$.

   In Table 4.1, we show the results obtained for this test problem with $\epsilon = 10^{-2}$. We clearly see a third order convergence rate in both L^1 and L^∞ norms. We also show in Table 4.2 the results obtained for the same example with $\epsilon = 10^{-4}$. Note that in order to observe high order accuracy, a higher value of ϵ is required compared to the one used by the purely parabolic reconstruction (see [7] and [13]).

   Compared with Table 4.1, the errors are larger, and the convergence rate is not as regular as before. This is mainly due to the fact that for a very small value of ε, condition i) is not satisfied, until the grid spacing Δx becomes very small. Note that, at variance with CWENO based on piecewise parabolic reconstruction, here a small deviation of the weights from their optimal constant value results in a degradation of the order of accuracy of the reconstruction.

2. **Linear advection with oscillatory data:** This test is used to detect deteriorations of accuracy due to oscillations in the parameters that control the selection of the stencil (for details see [2] and the references therein). Once again, the equation is $u_t + u_x = 0$, subject to the oscillatory initial data, $u(x, t\!=\!0) = \sin^4(\pi x)$ and to periodic boundary conditions on $[-1, 1]$. Here, the integration time is taken as $T = 1$ and $\epsilon = 10^{-4}$.

The results of this test are displayed in Table 4.3, and confirm the third-order accuracy of the method with no deteriorations in its accuracy.

3. **Burgers equation:** We solve the Burgers equation $u_t + (0.5u^2)_x = 0$, subject to the initial data $u(x, t=0) = 1 + 0.5\sin(\pi x)$ and to periodic boundary conditions on $[-1, 1]$. The integration time is $T = .33$, and $\epsilon = 10^{-4}$. Here, a shock develops at $T = 2/\pi$. Note that here the maximum speed of propagation is $f'(u) = 3/2$. Thus we use $\lambda = .66 * 3/7 \simeq 2/3\lambda_{\max}$

   Table 4.4 shows the results we obtained which verify the third-order accuracy of the method also for nonlinear problems.

   The result of the computation for time $t = 1.5$ is shown in Figure 4.1.

4. **2D Linear advection:** Finally, we implemented our method for the two-dimensional linear advection problem, $u_t + u_x + u_y = 0$. The initial condition is taken as $u(x, t=0) = \sin^2(\pi x)\sin^2(\pi y)$, and we impose periodic boundary conditions on $[0, 1]^2$. The errors and the estimated convergence rate are computed at time $T = 1$.

   In Table 4.5 we present the results obtained when the weights are taken as constants (3.12). Table 4.6 shows the results obtained with the fully non-linear scheme, where the weights of the reconstruction include also the oscillatory indicators. With constant weights our method is third-order as expected, while with non-constant weights, there seems to be a better convergence rate. A careful study of the tables shows, however, that this better convergence rate is mainly due to larger errors on the coarser grids.

N	L^1 error	L^1 order	L^∞ error	L^∞ order
20	0.1423	-	0.1484	-
40	0.1308E-01	3.44	0.1708E-01	3.12
80	0.7054E-03	4.21	0.1071E-02	4.00
160	0.7517E-04	3.23	0.7823E-04	3.78
320	0.9391E-05	3.00	0.7977E-05	3.29
640	0.1174E-05	3.00	0.9406E-06	3.08
1280	0.1467E-06	3.00	0.1158E-06	3.02

Table 4.1: Linear advection, $T = 10$, $u_0(x) = \sin(\pi x)$, $\epsilon = 10^{-2}$, Courant number $C = 0.9C_{\max}$, $C_{\max} = 3/7$

Example 2: 1D Systems - Euler Equations of Gas Dynamics

Next, we solve the Euler equations of gas dynamics,

$$\frac{\partial}{\partial t}\begin{pmatrix} \rho \\ m \\ E \end{pmatrix} + \frac{\partial}{\partial x}\begin{pmatrix} m \\ \rho u^2 + p \\ u(E+p) \end{pmatrix} = 0, \qquad p = (\gamma - 1)\cdot\left(E - \frac{\rho}{2}u^2\right),$$

N	L^1 error	L^1 order	L^∞ error	L^∞ order
20	0.2275	-	0.2499	-
40	0.7798E-01	1.54	0.8705E-01	1.52
80	0.8786E-02	3.15	0.1756E-01	2.31
160	0.6024E-03	3.87	0.1603E-02	3.45
320	0.3031E-04	4.31	0.6726E-04	4.58
640	0.1635E-05	4.21	0.2803E-05	4.58
1280	0.1467E-06	3.48	0.1740E-06	4.01

Table 4.2: Linear advection, $T = 10$, $u_0(x) = \sin(\pi x)$, $\epsilon = 10^{-4}$, Courant number $C = 0.9C_{\max}$, $C_{\max} = 3/7$

N	L^1 error	L^1 order	L^∞ error	L^∞ order
20	0.1368	-	0.2026	-
40	0.4685E-01	1.55	0.8759E-01	1.21
80	0.1247E-01	1.91	0.3320E-01	1.40
160	0.1326E-02	3.23	0.6239E-02	2.41
320	0.7851E-04	4.08	0.3951E-03	3.98
640	0.5383E-05	3.87	0.1585E-04	4.64
1280	0.5092E-06	3.40	0.8398E-06	4.24

Table 4.3: Linear advection, $T = 1$, $u_0(x) = \sin^4(\pi x)$, $\epsilon = 10^{-4}$, Courant number $C = 0.9C_{\max}$, $C_{\max} = 3/7$

N	L^1 error	L^1 order	L^∞ error	L^∞ order
20	0.1874E-01	-	0.3065E-01	-
40	0.3513E-02	2.42	0.1042E-01	1.56
80	0.2675E-03	3.72	0.9609E-03	3.44
160	0.1567E-04	4.09	0.4008E-04	4.58
320	0.1175E-05	3.74	0.3172E-05	3.66
640	0.1309E-06	3.17	0.4482E-06	2.82
1280	0.1660E-07	2.98	0.7042E-07	2.67

Table 4.4: Burgers equation, $T = .33$, $u_0(x) = 1 + 0.5\sin(\pi x)$, $\epsilon = 10^{-4}$, $\lambda = 0.66 * 3/7$, corresponding to a Courant number $C = 0.99C_{\max}$

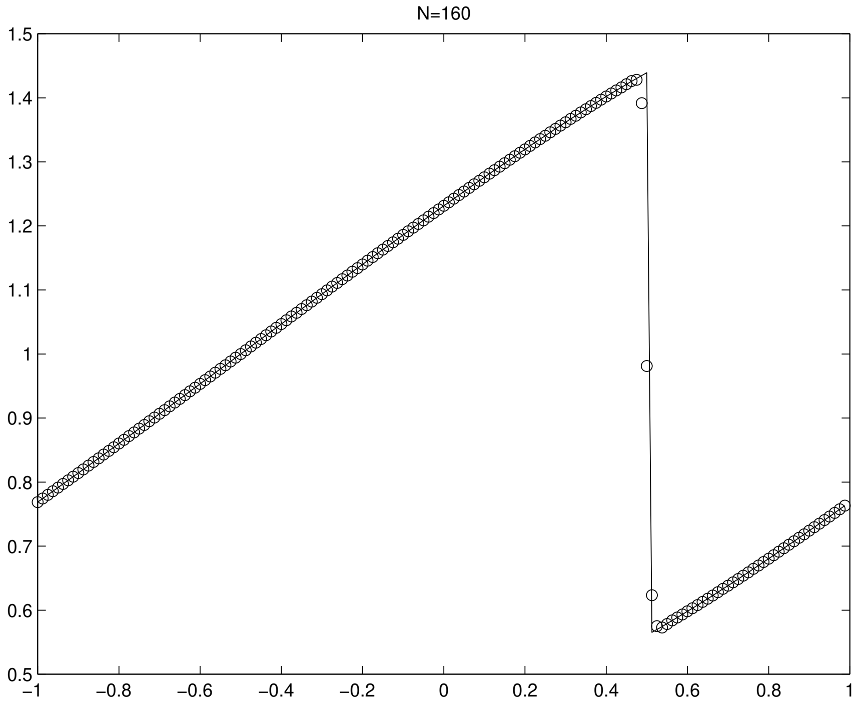

Figure 4.1: Burgers equation. Solution at time $t = 1.5$. $\lambda = 0.66 * 3/7$, $N = 160$, $\epsilon = 10^{-4}$

N	L^1 error	L^1 order	L^∞ error	L^∞ order
10	3.696E-02	-	1.252E-01	-
20	4.964E-03	2.90	1.767E-02	2.83
40	6.304E-04	2.98	2.264E-03	2.96
80	7.902E-05	3.00	2.842E-04	2.99
160	9.880E-06	3.00	3.555E-05	3.00

Table 4.5: 2D Linear Advection, Constant weights; $T = 1$, $\lambda = 0.425$, $u_0(x) = \sin^2(\pi x)\sin^2(\pi y)$,

N	L^1 error	L^1 order	L^∞ error	L^∞ order
10	1.154E-01	-	4.024E-01	-
20	3.273E-02	1.82	1.675E-01	1.26
40	7.933E-03	2.04	5.413E-02	1.63
80	6.107E-04	3.70	9.097E-03	2.57
160	2.662E-05	4.52	4.744E-04	4.26

Table 4.6: 2D Linear Advection, Non-Constant weights; $T = 1$, $\lambda = 0.425$, $u_0(x) = \sin^2(\pi x)\sin^2(\pi y)$, $\epsilon = 10^{-4}$

where the variables ρ, u, $m = \rho u$, p and E denote the density, velocity, momentum, pressure and total energy, respectively. We use two sets of initial data:

1. Shock tube problem with Sod's initial data, [24],

$$\begin{cases} (\rho_l, m_l, E_l) = (1, 0, 2.5), & x < 0.5, \\ (\rho_r, m_r, E_r) = (0.125, 0, 0.25), & x > 0.5. \end{cases}$$

2. Shock tube problem with Lax initial data, [9],

$$\begin{cases} (\rho_l, m_l, E_l) = (0.445, 0.311, 8.928), & x < 0.5, \\ (\rho_r, m_r, E_r) = (0.5, 0, 1.4275), & x > 0.5. \end{cases}$$

We integrate the equations up to time $T = 0.16$, using $\epsilon = 10^{-4}$. In Figure 4.2 we show the density components for Sod's initial data, and in Figure 4.3 we show the equivalent plot for Lax initial data. In both figures we also present the weight of the central stencil at the final time. All the results are given for 200 and 400 grid points. Following [20], we pick $\lambda = .1$. Note that the maximum characteristic speed for Sod's problem is roughly 2.5, while for Lax problem the maximum propagation speed is $\simeq 5$. Since our scheme has a Courant number no larger than .5, we see that $\lambda = .1$ is actually the maximum value compatible with stability.

It is interesting to compare the behavior of the central weight of the new method to the behavior of the central weight of the original CWENO method [13]. Here, the weights are much smoother compared with the weights in [13]. The accuracy and stability properties of the method can be related to the smoothness of the nonlinear weights involved (see [2]).

Example 3: 2D Problems

1. **Linear rotation:** Following [15], we consider a linear rotation of a square patch on $[0,1]^2$, with initial condition $u_0(x,y) = 1$ for $\{|x-1/2| \leq 1/2\} \times \{|y-1/2| \leq 1/2\}$ and zero elsewhere. In Figure 4.4 we display the solution after a rotation of $\pi/4$ and of $\pi/2$. As before, the value of ϵ is taken as 10^{-4}. There are no spurious oscillations. We also show the corresponding central weight. As expected, this weight is zero in regions where the solution has steep gradients, and that is exactly the property that prevents spurious oscillations from developing. Even though we are dealing with linear waves, the resulting resolution is relatively good. Due to the compactness of the stencil, when the slopes are not sharp, they are not identified as discontinuities. This can be observed in the plots of the central weight, which returns to its constant value (1/2) on the slope.

2. **2D - Burgers equation:** We end by solving the two-dimensional Burgers equation, $u_t+(u^2/2)_x+(u^2/2)_y = 0$, subject to the initial data $u(x,t=0) = \sin^2(\pi x)\sin^2(\pi y)$ and periodic boundary conditions on $[0,1]^2$. Here also $\epsilon = 10^{-4}$. In Figure 4.5 we present the solution obtained at time $T = 1.5$ with different mesh spacings. One can easily notice that the shocks are well resolved and there are no spurious oscillations.

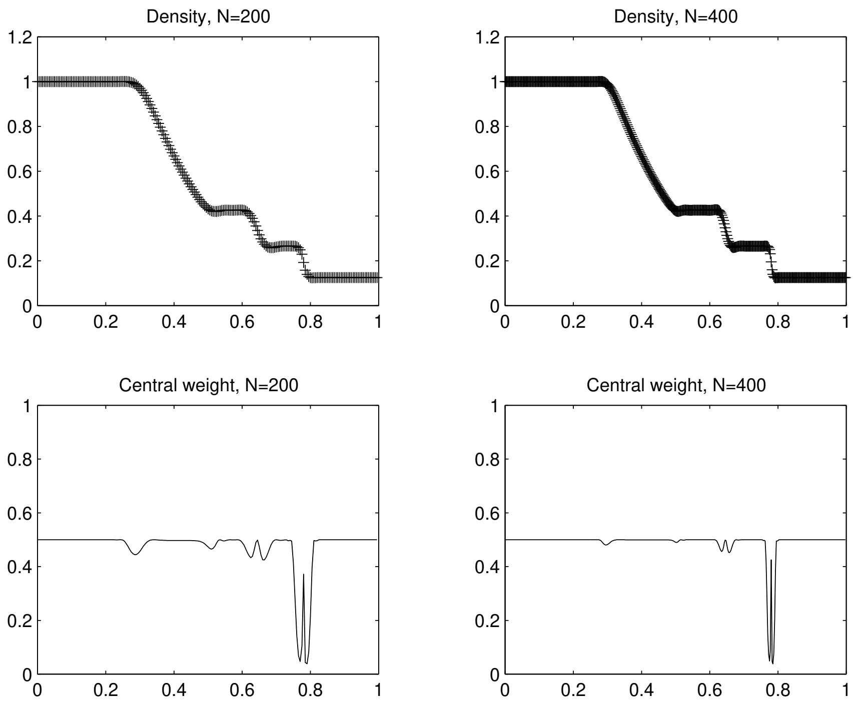

Figure 4.2: Euler equations of gas dynamics - Sod initial data, $\lambda = 0.1$, $T = 0.16$

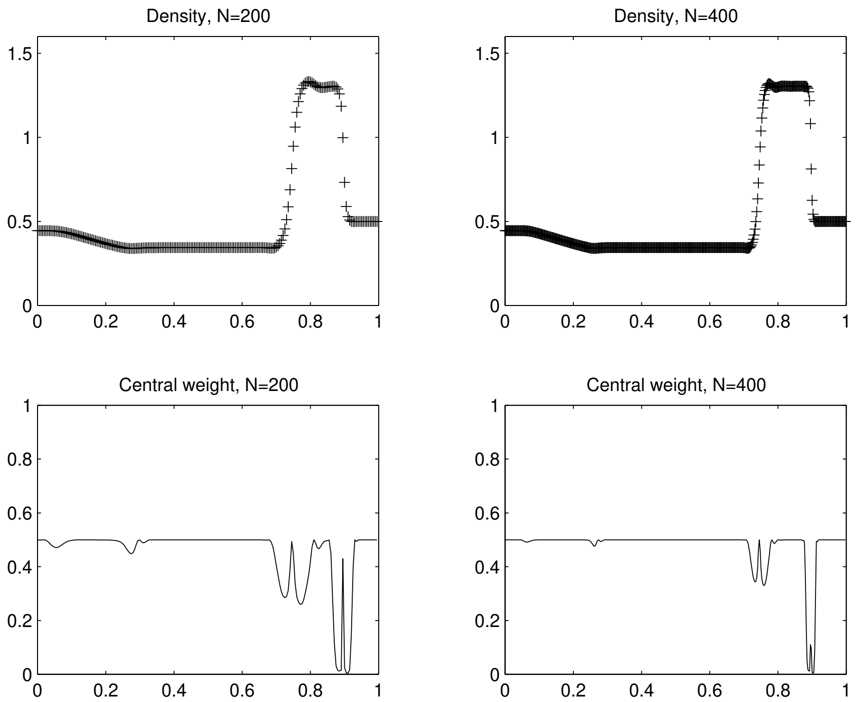

Figure 4.3: Euler equations of gas dynamics - Lax initial data, $\lambda = 0.1$, $T = 0.16$

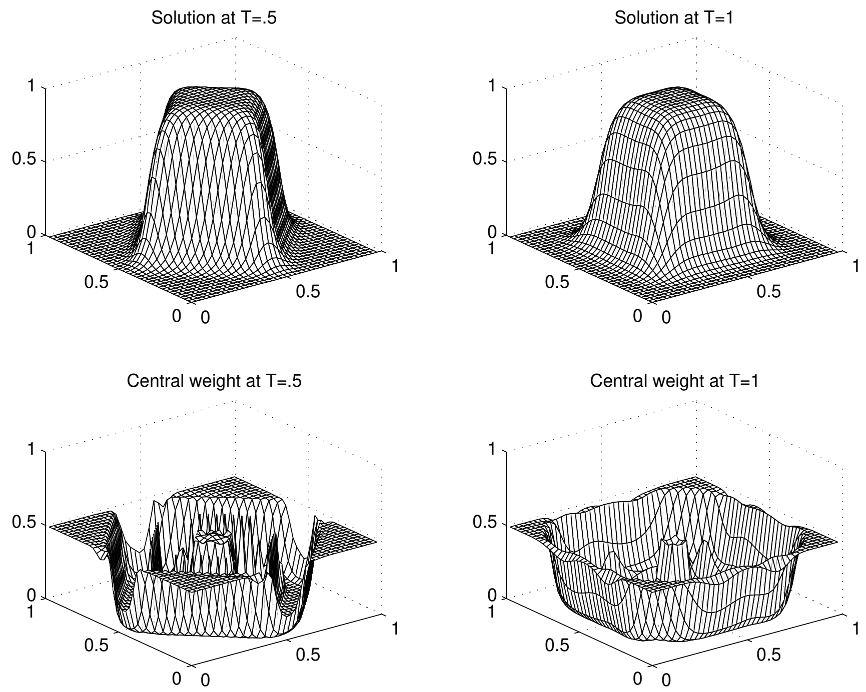

Figure 4.4: Linear Rotation, $\lambda = 0.425$, $N = 40$

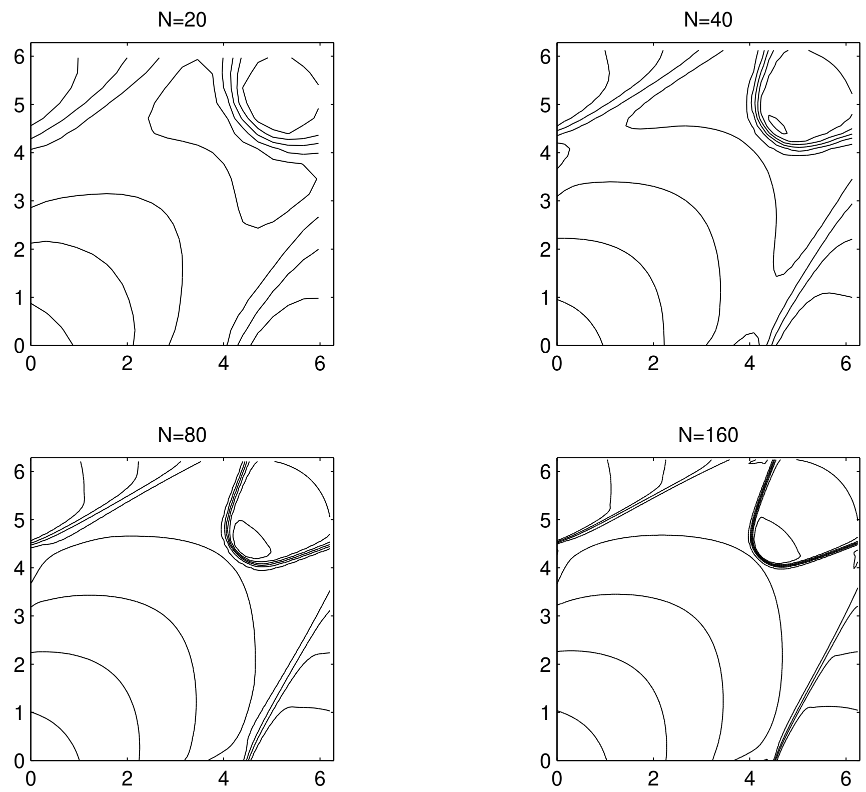

Figure 4.5: Two-Dimensional Burgers equation, $\lambda = 0.425$, $T = 1.5$

References

[1] Arminjon P., Stanescu D., Viallon M.-C., *A Two-Dimensional Finite Volume Extension of the Lax-Friedrichs and Nessyahu-Tadmor Schemes for Compressible Flow*, Proc. 6th. Int. Symp. on CFD, Lake Tahoe, 1995, M. Hafez and K. Oshima, editors, Vol. IV, pp.7-14.

[2] Bianco F., Puppo G., Russo G., *High Order Central Schemes for Hyperbolic Systems of Conservation Laws*, SIAM J. Sci. Comp., **21**, (1999), pp.294-322.

[3] Bianco F., Puppo G., Russo G., *High Order Central Schemes for Hyperbolic Systems of Conservation Laws*, International Series of Num. Math.,**129**, Birkhäuser Verlag (1999), pp.55-64.

[4] Friedrichs K. O., Lax P. D., *Systems of Conservation Equations with a Convex Extension*, Proc. Nat. Acad. Sci., **68**, (1971), pp.1686-1688.

[5] Godlewski E., Raviart P.-A., *Numerical Approximation of Hyperbolic Systems of Conservation Laws*, Springer, New York, 1996.

[6] Harten A., Engquist B., Osher S., Chakravarthy S., *Uniformly High Order Accurate Essentially Non-oscillatory Schemes III*, JCP, **71**, (1987), pp.231-303.

[7] Jiang G.-S., Shu C.-W., *Efficient Implementation of Weighted ENO Schemes*, JCP, **126**, (1996), pp.202-228.

[8] Jiang G.-S., Tadmor E., *Nonoscillatory Central Schemes for Multidimensional Hyperbolic Conservation Laws*, SIAM J. Sci. Comp., **19**, (1998), pp.1892-1917.

[9] Lax P. D., *Weak Solutions of Non-Linear Hyperbolic Equations and Their Numerical Computation*, CPAM, **7**, (1954), pp.159-193.

[10] van Leer B., *Towards the Ultimate Conservative Difference Scheme, V. A Second-Order Sequel to Godunov's Method*, JCP, **32**, (1979), pp.101-136.

[11] LeVeque R. J., *Numerical Methods for Conservation Laws*, Lectures in Mathematics, Birkhauser Verlag, Basel, 1992.

[12] Levy D., *A Third-order 2D Central Schemes for Conservation Laws*, INRIA School on Hyperbolic Systems, Vol. I (1998), pp.489-504.

[13] Levy D., Puppo G., Russo G., *Central WENO Schemes for Hyperbolic Systems of Conservation Laws*, Math. Model. and Numer. Anal., **33**, no. 3 (1999), pp. 547-571.

[14] Levy D., Puppo G., Russo G., *On the Behavior of the Total Variation in CWENO Methods for Conservation Laws*, Appl. Nume. Math., in press.

[15] Levy D., Puppo G., Russo G., *A Third Order Central WENO Scheme for 2D Conservation Laws*, Appl. Nume. Math., in press.

[16] Levy D., Puppo G., Russo G., *Central Weno Schemes for Multi-Dimensional Hyperbolic Systems of Conservation Laws*, in preparation.

[17] Liu X.-D., Osher S., *Nonoscillatory High Order Accurate Self-Similar Maximum Principle Satisfying Shock Capturing Schemes I*, SINUM, **33**, no. 2 (1996), pp.760-779.

[18] Liu X.-D., Osher S., Chan T., *Weighted Essentially Non-oscillatory Schemes*, JCP, **115**, (1994), pp.200-212.

[19] Liu X.-D., Tadmor E., *Third Order Nonoscillatory Central Scheme for Hyperbolic Conservation Laws*, Numer. Math., **79**, (1998), pp.397-425.

[20] Nessyahu H., Tadmor E., *Non-oscillatory Central Differencing for Hyperbolic Conservation Laws*, JCP, **87**, no. 2 (1990), pp.408-463.

[21] Sanders R., Weiser A., *A High Resolution Staggered Mesh Approach for Nonlinear Hyperbolic Systems of Conservation Laws*, JCP, **1010**, (1992), pp.314-329.

[22] Shu C.-W., *Essentially Non-Oscillatory and Weighted Essentially Non-Oscillatory Schemes for Hyperbolic Conservation Laws* in Advanced Numerical Approximation of Nonlinear Hyperbolic Equations, Lecture Notes in Mathematics (editor: A. Quarteroni), Springer, Berlin, 1998.

[23] Shu C.-W., Osher S., *Efficient Implementation of Essentially Non-Oscillatory Shock-Capturing Schemes, II*, JCP, **83**, (1989), pp.32-78.

[24] Sod G., *A Survey of Several Finite Difference Methods for Systems of Nonlinear Hyperbolic Conservation Laws*, JCP, **22**, (1978), pp.1-31.

[25] Tadmor E., *Approximate Solutions of Nonlinear Conservation Laws*, in Advanced Numerical Approximation of Nonlinear Hyperbolic Equations, Lecture Notes in Mathematics (editor: A. Quarteroni), Springer, Berlin, 1998.

[26] Zennaro M., *Natural Continuous Extensions of Runge-Kutta Methods*, Math. Comp., **46**, (1986), pp.119-133.