2 Central Schemes and Systems of Balance Laws

Introduction

The development of shock capturing schemes for the numerical approximation of the solution of conservation laws has been a very active field of research in the last two decades. There are several motivation for this effort. First, the challenge is a mathematical one. Solutions of conservation laws may develop jump discontinuities in finite time. To understand how to obtain numerical approximations that converge to the (discontinuous) solution has been a non trivial task. The mathematical theory of quasilinear hyperbolic systems of conservation laws has been used as a guideline in the development of modern shock capturing schemes. The concept of entropy condition and total variation diminishing are common to mathematical analysts and numerical analysts who deal with the problem. Another motivation for the development of shock capturing schemes is provided by the large number of applications. Complex flows in gas dynamics require the use of efficient and accurate schemes, which are able to deal with complex geometries. Unstructured mesh or adaptive mesh refinement become necessary so solve realistic problems. The schemes that are used for practical problems are usually different from the schemes for which theoretical results can be proven. For example, very little is known about the convergence property of high order schemes.

As a research field, I find it is one of the most fascinating in numerical analysis. In fact, it requires the knowledge of many areas of basic numerical analysis, from approximation theory to the theory of methods for ordinary differential equations (many sophisticated concepts developed in the context of numerical methods for ODE's are now being imported and used for the development of numerical methods of PDE's), it is based on the mathematical theory of weak solutions for conservation laws, and it is strongly motivated by the several applications in many physical systems, from gas dynamics, MHD, reacting flows, semiconductor modeling, and so on.

A good introductory book which deals with wave propagation and shock capturing schemes (mainly upwind, in one space dimension) is the book by LeVeque [37]. A mathematically oriented reference book on the numerical solutions of conservation laws is the book by Godlewski and Raviart [19]. Several schemes, mainly based on Riemann solver, with a lot of numerical examples are considered in the book by Toro [66]. A good review of modern numerical techniques for the treatment of hyperbolic systems of conservation laws is given in the lecture notes of a CIME course held in 1998 [14].

Most schemes for the numerical solution of conservation laws are based on the Godunov scheme, and on the numerical solution of the Riemann problem [34]. For numerical purpose it is often more convenient to resort to approximate solvers, such as the one proposed by Roe for gas dynamics [54].

Although such solution or its numerical approximation is known in many cases of physical relevance, there are other cases, such as gas in Extended Thermodynamics, or hydrodynamical models of semiconductors, in which the eigenvalues and eigenvectors of the matrix of the system are not known analytically, and for which the solution to the Riemann problem or its numerical approximation is hard to compute. In such cases it is desirable to use schemes that do not require the knowledge of the solution of the Riemann problem.

We call such schemes "Central schemes". The prototype central scheme is Lax-Friedrichs scheme. It is well known that Lax-Friedrichs scheme is more dissipative than first order upwind scheme, however it is simpler to use, since it does not require the knowledge of the sign of the flux derivative or the eigenvector decomposition of the system matrix (see for example [37] for a comparison between Lax-Friedrichs and upwind schemes).

Second order central schemes have been introduced in [50] and [57]. After that, central schemes have developed in several directions. We mention here the improvement of second order central scheme and the development of semidiscrete central scheme in one space dimension [32], the development of high order central schemes in one dimension [45, 11, 12, 38, 31], central schemes in several space dimensions on rectangular grids [6, 25, 28, 39, 40, 41, 43, 56, 57], and on unstructured grids [7, 8, 20], the development of central schemes to hyperbolic systems with source term [10, 46, 51], and the application of central schemes to geometrical optics [16], Hamilton-Jacobi equation [33, 44], incompressible flows, [30, 42, 29], hydrodynamical models of semiconductors [1, 4, 5, 55, 67]. The above list is far from being complete. It is meant to give an idea of the spreading of the recent spreading of central schemes in the numerical community.

The plan of the chapter is the following. In the next section we give a brief review on hyperbolic systems and their numerical approximation. Then we describe the popular Godunov method, based on the solution of the Riemann problem. Finally we present the second order Nessyahu-Tadmor scheme, with a few numerical examples. The next section is devoted to the development of high order central schemes in one space dimension. They are based on accurate non-oscillatory reconstruction of a function from cell-averages (Weighted Essentially Non-Oscillatory reconstruction), and on accurate evaluation of the integral of the flux, through the use of Runge-Kutta methods with Natural Continuous Extension. Section 2.3 is devoted to second and high order central schemes in two space dimensions. Some application to gas dynamics is presented. In the last section of the chapter we describe the treatment of systems with source terms, with particular emphasis on stiff sources. Finally, we describe possible developments of central schemes for systems with source.

2.1 Second order central schemes

2.1.1 Hyperbolic systems

Let us consider a system of equations of the form

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0 \tag{2.1}$$

where $u(x,t) \in \mathbb{R}^d$ is the unknown vector field, and $f : \mathbb{R}^d \to \mathbb{R}^d$ is assumed to be a smooth function. The system is hyperbolic in the sense that for any $u \in \mathbb{R}^d$, the Jacobian matrix $A = \nabla_u f(u)$ has real eigenvalues and its eigenvectors span $\mathbb{R}^d$.

Such system is linear if the Jacobian matrix does not depend on u , otherwise it is called quasilinear.

Linear hyperbolic systems are much easier to study. For these systems, the initial value problem is well posed, and the solution maintains the regularity of the initial data for any time.

Such systems can be diagonalized, and therefore they can be reduced to d linear scalar equations.

The situation is much different with quasilinear systems. For them the initial value problem is well posed locally in time. In general, the solution loses the regularity of the initial data after finite time. Even in the case of the single scalar equation, i.e. $d = 1$, the strong solution ceases to exist, and it is necessary to consider weak solutions. These have the general appearance of piecewise smooth functions, which contain jump discontinuities [35].

If we denote by x_Σ the position of the discontinuity and by V_Σ its velocity, then the jump conditions across Σ read

$$-V_\Sigma [\![u]\!] + [\![f]\!] = 0 \tag{2.2}$$

where for any quantity $h(x)$, $[\![h]\!] = h(x_\Sigma^+) - h(x_\Sigma^-)$ denotes the jump across the discontinuity interface Σ .

As an example, figure (2.1) shows the solution of Burgers equation, for which $d = 1$ and $f = u^2/2$, with the initial condition

$$u(x,0) = 1 + \frac{1}{2}\sin(\pi x) \tag{2.3}$$

$x \in [-1,1]$, and periodic boundary conditions. A discontinuity forms at time $t = 2/\pi \approx 0.6366$. The figure shows the initial condition, and the solution at times $t = 0.5$ and $t = 0.8$. In the latter case, the parametric solution constructed by the characteristics is multi-valued. Single valued solution is restored by fitting a shock discontinuity at a position that maintains conservation [69].

Piecewise smooth solutions that satisfy the jump conditions are not unique. Entropy condition is used to guarantee uniqueness of the solution, at least in the scalar case.

Entropy condition states that for any convex function $\eta(u)$ there exist an entropy flux $\psi(u)$ such that the pair $[\eta,\psi]$ satisfies

$$\frac{\partial \eta}{\partial t} + \frac{\partial \psi(u)}{\partial x} \le 0 \tag{2.4}$$

for any weak solution of the equation, and the equal sign holds for smooth solutions.

In the scalar case the entropy condition ensures that the weak solution is the unique viscosity solution, i.e. it is obtained as the limit

$$\lim_{\epsilon \to 0} u^\epsilon(x,t),$$

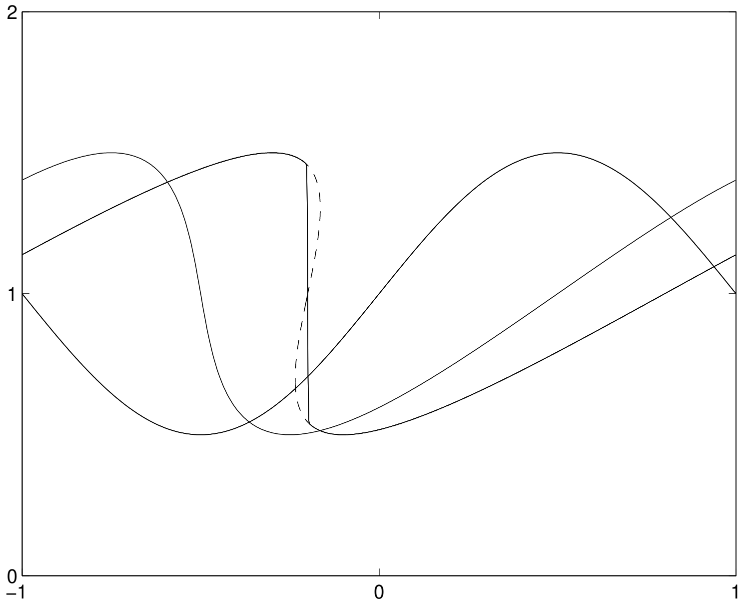

Figure 2.1 Burgers equation at different times ($t = 0{,}0.5{,}0.8$)

where u^ϵ satisfies the equation

$$\frac{\partial u^\epsilon}{\partial t} + \frac{\partial f(u^\epsilon)}{\partial x} = \epsilon \frac{\partial^2 u^\epsilon}{\partial x^2}.$$

For the relation between entropy condition and viscosity solutions in the case of systems see, for example, [35].

The mathematical theory of hyperbolic systems of conservation laws is used as a guideline for the construction of schemes for the numerical approximation of conservation laws. Consider, for example, the conservation property. Integrating Eq.(2.1) over an interval $[a,b]$ one has

$$\frac{d}{dt}\int_a^b u(x,t)\,dx = f(u(a,t)) - f(u(b,t))$$

if $u(a,t) = u(b,t)$ (for example if the boundary conditions are periodic) then the quantity $\int_a^b u(x,t)$ is conserved in time. Such conservation property is directly related to the jump condition (2.2).

It is important that a similar conservation property is maintained at a discrete level. In this way the schemes will provide the correct propagation speed for the discontinuities.

2.1.2 Conservative schemes

Most commonly used schemes for the numerical approximation of conservation laws are finite element, finite difference, and finite volume schemes. We shall mainly consider here finite volume schemes.

These are obtained by discretizing space into cells $I_j = [x_{j-1/2}, x_{j+1/2}]$, and time in discrete levels t_n. For simplicity we assume that the cells are all of the same size $\Delta x = h$, so that the center of cell j is $x_j = x_0 + jh$. This assumption is not necessary for finite volume schemes.

Integrating the conservation law on a cell in space-time $I_j \times [t_n, t_{n+1}]$ (see figure (2.2)) one has

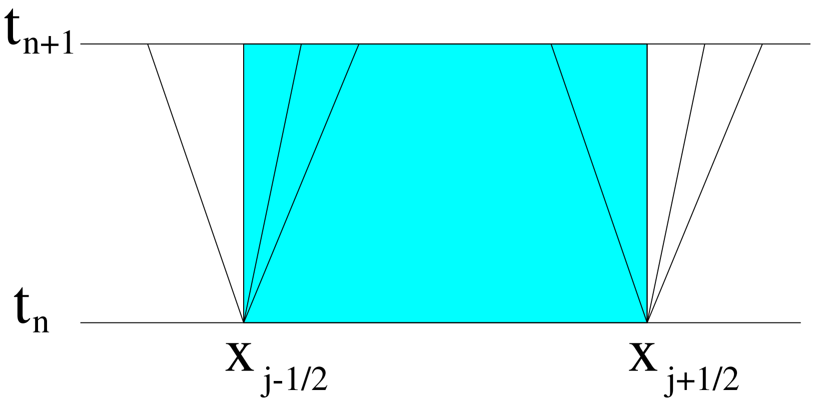

Figure 2.2 Integration over a cell and Godunov methods

$$\int_{x_{j-1/2}}^{x_{j+1/2}} u(x, t_{n+1}) = \int_{x_{j-1/2}}^{x_{j+1/2}} u(x, t_n) - \int_{t_n}^{t_{n+1}} (f(u(x_{j+1/2}, t)) - f(u(x_{j-1/2}, t)))\, dt. \quad (2.5)$$

This (exact) relation suggests the use of numerical scheme of the form

$$\bar{u}_j^{n+1} = \bar{u}_j^n - \frac{\Delta t}{h}(F_{j+1/2} - F_{j-1/2}), \quad (2.6)$$

where $\bar{u}_j^n$ denotes an approximation of the cell average of the solution on cell j at time t_n, and $F_{j+1/2}$, which approximates the integral of the flux on the boundary of the cell, is the so called *numerical flux*, and depends on the cell average of the cells surrounding point $x_{j+1/2}$. In the simplest case it is

$$F_{j+1/2} = F(\bar{u}_j, \bar{u}_{j+1})$$

with $F(u, u) = f(u)$ for consistency. Such schemes, called conservative schemes, have the properties that they satisfy a conservation property at a discrete level. This is essential in providing the correct propagation speed for discontinuities, which depend, through Eq. (2.2), uniquely on the conservation properties of the system.

Furthermore, Lax-Wendroff theorem ensures that if $u(x, t)$ is the limit of a sequence of discrete solutions $\bar{u}_j^n$ of a consistent conservative scheme, obtained as the discretization parameter h vanishes, then $u(x, t)$ is a weak solution of the original equation.

Lax-Wendroff theorem assumes that the sequence of numerical solutions converges strongly to a function $u(x,t)$. Convergence of numerical schemes is studied through the TVD (Total Variation Diminishing) property. A discrete entropy condition is used to guarantee that the numerical solution converges to the unique admissible solution of Eq. (2.1). For a discussion on these issues see the book by LeVeque [37] of Godlewski and Raviart [19].

2.1.3 Godunov scheme

The prototype of upwind schemes for conservation laws is the Godunov scheme. It is based on two fundamental ideas. The first is that the solution is reconstructed from cell averages at each time step as a piecewise polynomial in x. In its basic form, the solution is reconstructed as a piecewise constant function

$$u(x,t_n) \approx R(x;\bar{u}^n) = \sum_j \bar{u}_j^n \chi_j(x) \tag{2.7}$$

where $\chi_j(x)$ is the indicator function of interval $I_j = [x_{j-1/2},x_{j+1/2}]$. The second is that for a piecewise constant function, the solution of the system, for short time, can be computed as the solution of a sequence of Riemann problems.

A Riemann problem is a Cauchy problem for a system of conservation laws, where the initial condition is given by two constant vectors separated by an interface

$$u(x,0) = \begin{cases} u_- & x < 0 \\ u_+ & x > 0 \end{cases} \tag{2.8}$$

For the scalar equation the solution to the Riemann problem is known analytically. For a system of conservation laws, the solution consists of a similarity solution that depends on the variable x/t. In several cases of interests, such as gas dynamics with polytropic gas, the solution to the Riemann problem is known analytically [35].

Sometimes, for efficiency reason, it is more convenient to use approximate solutions to the Riemann problem [54].

Once the solution to the Riemann problem is known, it can be used for the construction of the Godunov scheme.

Let us denote by $u^*(u_-,u_+)$ the solution of the Riemann problem at $x=0$. Then the *exact* solution of Eq.(2.1) can be computed from Eq.(2.5):

$$\bar{u}_j^{n+1} = \bar{u}_j^n - \frac{\Delta t}{\Delta x}(f(u^*(\bar{u}_{j-1},\bar{u}_j)) - f(u^*(\bar{u}_j,\bar{u}_{j+1}))) \tag{2.9}$$

This relation is exact if the Riemann fan does not reach the boundary of the cell (see figure (2.2)), i.e. if the following CFL condition is satisfied

$$\Delta t < \frac{\Delta x}{\rho(A)} \tag{2.10}$$

where $\rho(A) = \max_{1\leq i\leq d} |\lambda_i(A)|$ is the spectral radius of the Jacobian matrix $A = \nabla_u f$, λ_i denoting the i-th eigenvalue.

Once the cell averages are computed at the new time t_{n+1}, then the solution at this time is again approximated by a piecewise constant solution of the form (2.7).

Godunov scheme is first order accurate, it is Total Variation Diminishing, and it satisfies a discrete entropy inequality. When applied to a linear system, Godunov method is equivalent to first order upwind scheme (see [37]).

Higher order version of Godunov scheme can be constructed. They are based on high order non oscillatory reconstruction and on the solution to the generalized Riemann problem.

High order non-oscillatory reconstruction is a crucial step, and we shall discuss it in detail when we deal with high order central schemes.

For the moment, we assume we are able to compute such reconstruction, of the form

$$u(x,t_n) \approx R(x;\bar{u}^n) = \sum_j R_j(x)\chi_j(x) \tag{2.11}$$

Then high order Godunov-type schemes are obtained by solving the system

$$\frac{\mathrm{d}\bar{u}_j}{\mathrm{d}t} = -\frac{f(u^*(u^-_{j+1/2},u^+_{j+1/2})) - f(u^*(u^-_{j-1/2},u^+_{j-1/2}))}{h} \tag{2.12}$$

where $u^-_{j+1/2} = R_j(x_{j+1/2})$, $u^+_{j+1/2} = R_{j+1}(x_{j+1/2})$. Because the values $u^-_{j+1/2}$ and $u^+_{j+1/2}$ depend on the reconstruction, which depends on the cell averages, it turns out that system (2.13) is a system of ordinary differential equations for the evolution of cell averages.

Such system may be solved by a suitable ODE solver, for example a Runge-Kutta method, which maintains the accuracy of the spatial discretization (see [58] and references therein).

These methods are based on the (exact or approximate) solution to the Riemann problem. Such solution is not always available or inexpensive. As an alternative to these high order extension of the Godunov method, simpler schemes can be constructed, which make use of less expensive numerical flux function.

The general structure of such schemes is given by

$$\frac{\mathrm{d}\bar{u}_j}{\mathrm{d}t} = -\frac{F(u^-_{j+1/2},u^+_{j+1/2}) - F(u^-_{j-1/2},u^+_{j-1/2})}{h} \tag{2.13}$$

where $u^-_{j+1/2}$ and $u^+_{j+1/2}$ are defined as above, and the the numerical flux function $F(u^-,u^+)$ defines the scheme. The simplest choice of the numerical flux function is the so called Local Lax-Friedrichs flux:

$$F(u^-,u^+) = \frac{1}{2}(f(u^-) + f(u^+) - \alpha(u^+ - u^-)) \tag{2.14}$$

where $\alpha = \max(\rho(A(u^-)),\rho(A(u^+)))$, and $\rho(A)$ denotes the spectral radius of matrix A. The advantage of the local Lax-Friedrichs flux is that it does not require the knowledge of the solution to the Riemann problem, nor the exact knowledge of the eigenvalues and eigenvectors of the Jacobian matrix. Only an estimate of the largest eigenvalue is needed.

The disadvantage of this flux with respect to the Riemann solver is that it introduces a larger numerical dissipation.

Other flux functions are available. Common requirements that they have to satisfy are the following: they have to be

i) locally Lipschitz continuous in both argument

ii) nondecreasing in the first argument and non-increasing in the second argument (symbolically $F(\uparrow, \downarrow)$).

iii) consistent with the flux function, i.e. $F(u,u) = f(u)$.

Popular flux functions (for scalar equation), besides the local Lax Friedrichs described above, are the Godunov flux

$$F(a,b) = \begin{cases} \min_{a \le u \le b} f(u) & \text{if } a \le b \\ \max_{b \le u \le a} f(u) & \text{if } a > b \end{cases}$$

and the Engquist-Osher flux [15]

$$F(a,b) = \int_0^a \max(f'(u),0)\,du + \int_0^b \min(f'(u),0)\,du + f(0)$$

Godunov flux is the least dissipative, and the Lax-Friedrichs the most of the three.

Notice, however, that the numerical dissipation is proportional to the jump $u^+ - u^-$, which is extremely small for high order schemes and smooth solution. For a scheme of order p, in fact, it is $u^+ - u^- = O(h^p)$. Therefore the numerical dissipation becomes large only near discontinuities, i.e. where it is most needed. As a result, the difference between exact Riemann flux and local Lax-Friedrichs flux is very pronounced for low order schemes, but it is not so dramatic for very high order schemes [58].

We shall come back to discuss about high order schemes later.

Now let us consider again a piecewise polynomial reconstruction and let us integrate the equation (2.1) on a staggered grid, as shown in Fig.(2.3).

Integrating on the staggered grid one obtains

$$\int_{x_j}^{x_{j+1}} u(x,t_{n+1}) = \int_{x_j}^{x_{j+1}} u(x,t_n) - \int_{t_n}^{t_{n+1}} \left(f(u(x_{j+1},t)) - f(u(x_j,t))\right) dt \tag{2.15}$$

Once again, this formula is *exact*. In order to convert it into a numerical scheme one has to approximate the staggered cell average at time t_n, and the time integral of the flux on the border of the cells.

Let us assume that the function $u(x,t_n)$ is reconstructed form cell averages as a piecewise polynomial function. Then the function is smooth at the center of the cell and its discontinuities are at the edge of the cell. If we integrate the equation on a staggered cell, then there will be a fan of characteristics propagating from the center of the staggered cell, while the function on the edge (dashed vertical lines in the figure) will remain smooth, provided the characteristic fan does not intersects the edge of the cell, i.e. provided a suitable CLF condition of the form

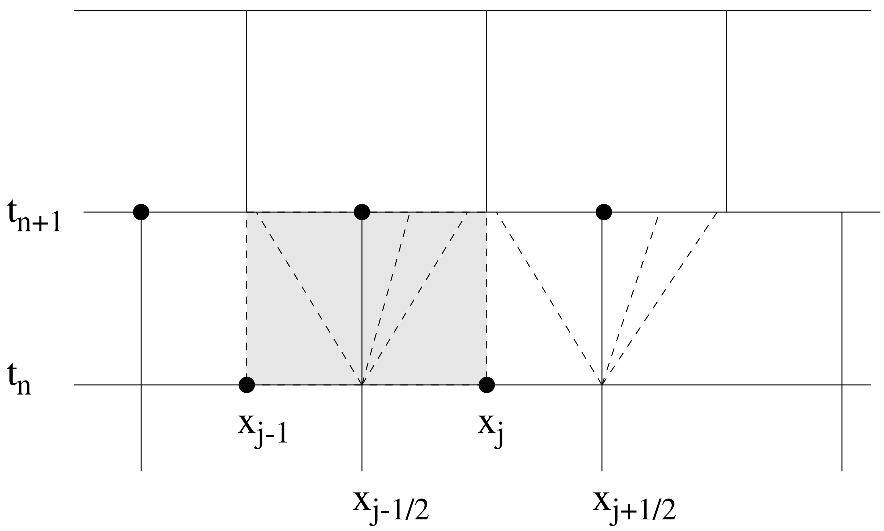

Figure 2.3 Integration on a staggered grid

$$\Delta t < \frac{\Delta x}{2\rho(A)} \tag{2.16}$$

is satisfied.

The simplest central scheme is obtained by piecewise constant reconstruction of the function, and by using a first order quadrature rule in the evaluation of the integrals. The resulting scheme is

$$\bar{u}^n_{j+1/2} = \frac{1}{2}(\bar{u}^n_j + \bar{u}^n_{j+1}) - \lambda(f(\bar{u}^n_{j+1}) - f(\bar{u}^n_j)) \tag{2.17}$$

where $\lambda = \Delta t/\Delta x$ denotes the mesh ratio. Such scheme is just Lax-Friedrichs scheme on a staggered grid. The advantage of the Lax-Friedrichs scheme over upwind scheme is that it is much simpler to apply, since no knowledge is required of the characteristic structure of the system. However, the disadvantage is that it is much more dissipative that the upwind scheme.

As an example, let us consider the numerical solution of the Riemann problem for the simple linear equation

$$u_t + u_x = 0, \quad u(x,0) = \begin{cases} 1 & x < 0 \\ 0 & x > 0 \end{cases} \tag{2.18}$$

The result for upwind and Lax Friedrichs schemes are shown in Fig.(2.4), obtained with 100 grid points on the interval $[-0.2,1.8]$, $t_{\max} = 1.4$ and $\Delta t/\Delta x = 0.8$ for the upwind scheme and $\Delta t/\Delta x = 0.4$ for the staggered Lax-Friedrichs. Both schemes are exact for the linear equation when the maximum mesh ratio (1 for upwind and 1/2 for LxF) is used. Notice that the first order upwind scheme performs slightly better than the staggered Lax-Friedrichs, in spite of the fact that it uses half time steps (due do the more severe stability condition of the latter).

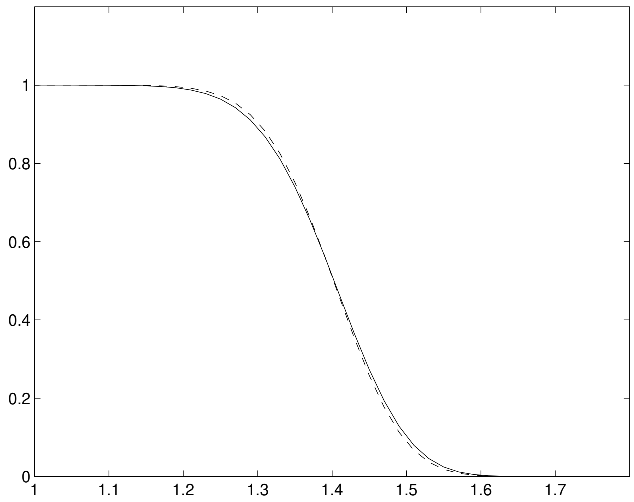

Figure 2.4 Comparison between staggered Lax-Friedrichs (continuous line) and first order upwind (dashed line). $\Delta t/\Delta x$ is 0.8 for upwind scheme and 0.4 for Lax-Friedrichs scheme

2.1.4 The Nessyahu-Tadmor scheme

A second order scheme is obtained by using a piecewise linear approximation for the reconstruction of the function, and a second order quadrature rule (for example the midpoint rule) for the computation of the time integral of the flux on the edges of the cell.

Such scheme has been proposed by Nessyahu and Tadmor [50], and independently by Sanders and Weiser [57].

The staggered cell average is given by

$$\begin{aligned}\bar{u}^n_{j+1/2} &= \frac{1}{h}\int_{x_j}^{x_{j+1}} R(x;\bar{u}^n)\,dx = \frac{1}{h}\left(\int_{x_j+1/2}^{x_{j+1}} L_j(x)\,dx + \int_{x_j}^{x_{j+1/2}} L_{j+1}(x)',dx\right)\\ &= \frac{1}{2}(\bar{u}^n_j + \bar{u}^n_{j+1}) - \frac{1}{8}(u'_{j+1} - u'_j),\end{aligned}$$

where $L_j(x) = \bar{u}^n_j + u'_j(x - x_j)/h$ is the linear reconstruction in cell I_j . . Here u'_j/h denotes a first order approximation of the derivative of the function in the cell. The value of the field u at the node of the midpoint rule, $u_j^{n+1/2}$, can be computed by first order Taylor expansion, which is equivalent to forward Euler scheme,

$$u_j^{n+1/2} = \bar{u}^n_j - (\lambda/2)f'_j.$$

Here f'_j/h denotes a first order approximation of the space derivative of the flux.

In order to prevent spurious oscillations in the numerical solution, it is essential that these derivatives are computed by using a suitable *slope limiter*. Several choices are possible for the slope limiter. The simplest one is the MinMod limiter. It is defined according to

$$\mathrm{MM}(a,b) = \begin{cases} \min(a,b) & \text{if} \quad a<0 \quad \text{AND} \quad b<0 \\ \max(a,b) & \text{if} \quad a>0 \quad \text{AND} \quad b>0 \\ 0 & \text{if} \quad ab<0 \end{cases} .$$

Such simple limiter, however, degrades the accuracy of the scheme near extrema. A better limiter is the so called UNO (Uniform Non Oscillatory) limiter, proposed by Harten et al. [23].

Such limiter can be written as

$$u'_j = \mathrm{MM}(d_{j-1/2} + \frac{1}{2}\mathrm{MM}(D_{j-1},D_j), d_{j+1/2} - \frac{1}{2}\mathrm{MM}(D_j,D_{j+1})), \tag{2.19}$$

where

$$d_{j+1/2} = u_{j+1} - u_j, \quad D_j = u_{j+1} - 2u_j + u_{j-1}.$$

Other limiters are possible (see [50]).

The quantity f'_j can be computed either by applying a slope limiter to $f(\bar{u}_j^n)$ or by using the relation

$$f'_j = A(\bar{u}_j^n)u'_j.$$

After one time step, one finds the solution $\{\bar{u}_{j+1/2}^{n+1}\}$ on the staggered cells. Then one repeats a similar step, and after the second step, the solution at time t_{n+2}, $\{\bar{u}_j^{n+1}\}$ is determined on the original grid.

Theoretical properties of the scheme, such as TVD property and the so called "cell entropy inequality" are discusses in [50].

As an example we show here some numerical tests performed by the NT scheme. Figure

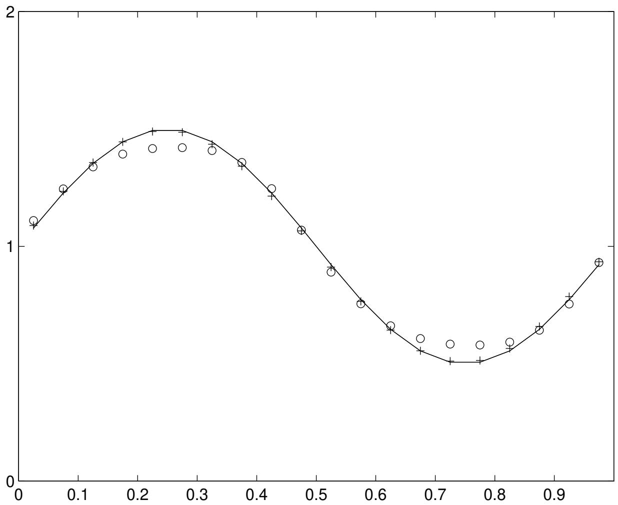

Figure 2.5 NT scheme applied to the linear equation. Mesh ratio $\Delta t/\Delta x = 0.45$. $t_{\max} = 5$. $N = 20$. MM limiter (circles) and UNO limiter (+). The line represents the exact solution.

(2.5) represents the computation on the linear equation $u_t + u_x = 0$, performed by using the two limiters described above. The initial condition is $u(x,0) = 1 + 0.5\sin(2\pi x)$, and the boundary conditions are periodic in $[0,1]$. The solution is computed after 5 periods. The clipping effect due to the MinMod limiter is evident, while the NT scheme with UNO limiter maintains good accuracy after a long time.

The computation shown in Figure (2.1) was performed by NT scheme with 1000 grid points.

More interesting test concerns the solution of the Euler equation of gas dynamics.

Such equations are of the form (2.1), with

$$u = \begin{pmatrix} \rho \\ \rho v \\ E \end{pmatrix}, \quad f = \begin{pmatrix} \rho v \\ \rho v^2 + p \\ v(E+p) \end{pmatrix}, \tag{2.20}$$

where ρ , v , p , and E denote, respectively, the density, velocity, pressure and energy density per unit volume of the gas. For a polytropic gas, energy and pressure are related by

$$p = (\gamma - 1)(E - \frac{1}{2}\rho v^2),$$

where $\gamma = c_p/c_v$ is the polytropic constant, c_p and c_v denote the specific heat at constant pressure and volume, respectively. For a monoatomic gas it is $\gamma = 5/3$, and for a biatomic gas, like air, it is $\gamma = 7/5$. Typical test problems for gas dynamics are the Sod and Lax tests. These are defined by the initial condition

$$\begin{cases} (\rho_l,(\rho v)_l,E_l) = (1,0,2.5), & x < 0.5, \\ (\rho_r,(\rho v)_r,E_r) = (0.125,0,0.25), & x > 0.5. \end{cases} \qquad \text{Sod problem} \tag{2.21}$$

$$\begin{cases} (\rho_l,(\rho v)_l,E_l) = (0.445,0.311,8.928), & x < 0.5, \\ (\rho_r,(\rho v)_r,E_r) = (0.5,0,1.4275), & x > 0.5. \end{cases} \qquad \text{Lax problem} \tag{2.22}$$

The polytropic constant is $\gamma = 1.4$ and the final time is $t_{\max} = 0.16$.

The result of NT scheme applied to the Sod problem is shown in Figure (2.6).

2.2 High order central schemes

In most applications, second order schemes have been successfully used. There is, however, a great interest in constructing and using high order (or it would be more appropriate to say high resolution) schemes.

It may seem strange to look for high order schemes for the numerical solution of quasi-linear hyperbolic systems, since the solution itself is not regular, and in general presents jump discontinuities. However, we shall show that there are several reasons for using high order schemes.

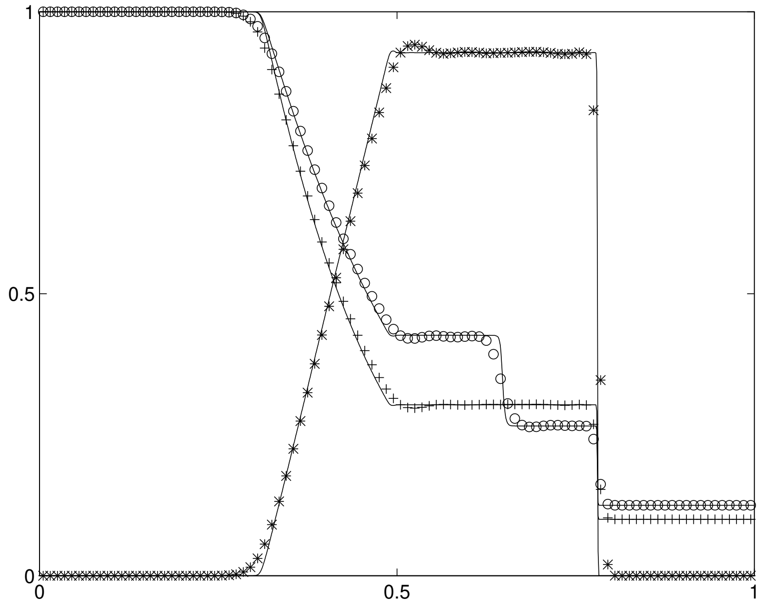

Figure 2.6 NT scheme applied to the Sod problem of gas dynamics. $N = 100$. $t_{\max} = 0.16$. UNO limiter. Courant number 0.475. Density (o), velocity (*), and pressure (+). The line is the reference solution computed by 1000 grid points.

Although the mathematical theory of quasilinear hyperbolic systems is developed in the space of bounded variation functions, in most practical cases the solutions are more regular than that. In fact in many practical cases they are piecewise smooth functions, very well represented by piecewise polynomials. This is the reason why numerically computed convergence rate are better than the sharpest theoretical estimate: the convergence rate is computed on piecewise smooth solutions, while the theoretical estimates are computed on BV functions. Pointwise error estimates for the convergence of viscosity solution to the scalar conservation law are considered in [63, 64].

High order schemes give good accuracy of the solution in the smooth regions. They also provide sharp resolution of the discontinuities. These two benefits justify the use of high resolution schemes.

A warning, however, is necessary. Most schemes for the treatment of discontinuous solutions are non linear, and with variable order. They adjust themselves, so that near nonlinear discontinuities the right amount of numerical dissipation is introduced in order to prevent the formation and amplification of spurious oscillations.

The loss of accuracy near the discontinuities may affect the accuracy even in the smooth regions, since small perturbations formed near the shock may propagate in the inner flow. Such effect is usually small, and it is presently subject of investigation [17].

There is a more subtle reason for desiring high resolution central schemes: economy. We shall see that it is possible to design high resolution central schemes that are very easy to use, and that can be easily adapted to a large variety of systems, without any knowledge of the solution of the Riemann problem. These are the Central WENO schemes, that we describe in the next sections, and their semidiscrete analogue, i.e. the WENO schemes with local Lax-Friedrichs flux function. Although such schemes are more dissipative than state-of-the-art Riemann-based flux functions, the difference in the resolution is sensible at the level of low (first and second) order schemes, while the difference is almost negligible for high resolution schemes. For this reason, it is sometimes more convenient to use a third order central scheme than a second order upwind scheme.

In the construction of high order central schemes, let us start from relation (2.15). We assume that the solution at time t_n is reconstructed as a piecewise polynomial function as in (2.11). Then, by approximating the integral of the flux by a quadrature formula, one obtains the general scheme

$$\bar{u}^{n+1}_{j+1/2} = \frac{1}{\Delta x}\int_{x_j}^{x_{j+1}} R(x;\bar{u}^n)\,dx \tag{2.23}$$

$$+\lambda\sum_{i=0}^{s}\gamma_l[f(\hat{u}(x_j,t_n+\beta_l\Delta t)) - f(\hat{u}(x_{j+1},t_n+\beta_l\Delta t))] \tag{2.24}$$

The parameters γ_l and β_l are the weights and the nodes of the particular quadrature formula, and $\hat{u}$ are the intermediate values of the field at the nodes, predicted by Runge-Kutta method (see below). For a fourth-order method one can use, for example, Simpson's rule.

The staggered cell-averages at time t_n, $\bar{u}^n_{j+1/2}$, are given by

$$\bar{u}^n_{j+1/2} = \frac{1}{h}\int_{x_j}^{x_{j+1}} R(x;\bar{u}^n)dx = \frac{1}{h}\left[\int_{x_j}^{x_{j+1/2}} R_j(x)dx + \int_{x_{j+1/2}}^{x_{j+1}} R_{j+1}(x)dx\right]. \tag{2.25}$$

2.2.1 Time evolution: Runge-Kutta methods with Natural Continuous Extension

Consider the Cauchy problem

$$\begin{cases} y' = F(t,y(t)) \\ y(t_0) = y_0. \end{cases}$$

We adapt to our context the notation used in [70] according to which the solution at the $(n+1)$-th time step obtained with a ν stage Runge-Kutta scheme can be written as

$$y^{n+1} = y^n + \Delta t\sum_{i=1}^{\nu} b_i K^{(i)}, \tag{2.26}$$

where the $K^{(i)}$'s are the Runge-Kutta fluxes

$$K^{(i)} = F\left(t^n + \Delta t c_i, y^n + \Delta t\sum_{j=1}^{i-1} a_{ij}K^{(j)}\right),$$

and the c_i are given by $c_i = \sum_j a_{ij}$. The method is completely determined by the vector b and the matrix a, which is lower triangular with zero diagonal for explicit schemes.

In our case we are solving a sequence of Cauchy problems in order to obtain the predicted mid-values required for the quadrature of the fluxes. At the j-th grid point we have

$$\begin{cases} y_j'(\tau) &= F(\tau, y_j(\tau)) = -f_x\left(y(x_j, t^n + \tau)\right) \\ y_j(\tau = 0) &= R(x_j; \bar{u}^n). \end{cases} \tag{2.27}$$

Thus the computation of the i-th Runge-Kutta flux $K^{(i)}$ requires the evaluation of the x-derivative of f at the intermediate time $t = t^n + c_i\Delta t$. The predicted point-values of u_j at time $t = t^n + c_i\Delta t$ are used to compute the point-values, $f(u_j)$. These predicted values of f are then used for reconstructing an interpolant from which the point-values of the derivative $(f_x)_j$ are computed. To maintain high accuracy and control over oscillations in the evaluation of f_x required in (2.27), the reconstruction of the interpolant of both f and u is essentially the same and is described in §2.2.2.

Following [11], we use here *Natural Continuous Extensions* (NCE) of Runge-Kutta schemes which provide a uniform accuracy of the solution in the time interval $[t^n, t^{n+1}]$ (consult [70]). Each ν-stage Runge-Kutta method of order p has an NCE u of degree $\bar{d} \leq p$ in the sense that there exist ν polynomials $b_i(\theta)$, $i = 1, \cdots, \nu$ of degree at most $\bar{d}$, such that

1. $u(t^n + \theta\Delta t) := y^n + \Delta t \sum_{i=1}^{\nu} b_i(\theta) K^{(i)} \qquad 0 \leq \theta \leq 1;$
2. $u(t^n) = y^n \qquad$ and $\qquad u(t^n + \Delta t) = y^{n+1};$
3. $\max_{t^n \leq t \leq t^n + \Delta t} \left| w^{(l)}(t) - u^{(l)}(t) \right| = O((\Delta t)^{d+1-l}), \qquad 0 \leq l \leq \bar{d},$

where y^n is the numerical solution computed with the RK scheme at time level t^n, and $w(t)$ is the exact solution of the equation with $w(t^n) = y^n$. Note that the polynomials $b_i(\theta)$ depend only on the Runge-Kutta method chosen and not on the particular ODE being solved. In the next sections we shall show how to use Runge-Kutta methods with NCE in order to guarantee high order accuracy in time.

We end this section listing the NCE's which are of interest for our schemes.

1) RK1 (for *second-order* scheme)
$d = \nu = p = 1$
$b_1(\theta) = \theta;$

2) RK2 (for *third-order* scheme)
$d = \nu = p = 2$
$b_1(\theta) = (b_1 - 1)\theta^2 + \theta$
$b_2(\theta) = b_2\theta^2;$

3) RK3 (Not used since no NCE of degree $\bar{d} = 3$ exists in this case) (2.28)
$d = 2, \nu = p = 3$
$b_i(\theta) = 3(2c_i - 1)b_i\theta^2 + 2(2 - 3c_i)b_i\theta \qquad i = 1,2,3$

4) RK4 (for *fourth-order* scheme)
$d = 3, \nu = p = 4$
$b_1(\theta) = 2(1 - 4b_1)\theta^3 + 3(3b_1 - 1)\theta^2 + \theta$
$b_i(\theta) = 4(3c_i - 2)b_i\theta^3 + 3(3 - 4c_i)b_i\theta^2 \qquad i = 2,3,4$

No NCE of degree $\bar{d} = 4$ exists in this case.

In our case, we used the following set of coefficients. For our third order method:

$$b = \begin{pmatrix} 1/2 \\ 1/2 \end{pmatrix} \qquad a = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \tag{2.29}$$

For our fourth order method:

$$b = \begin{pmatrix} 1/6 \\ 1/3 \\ 1/3 \\ 1/6 \end{pmatrix} \qquad a = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1/2 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{2.30}$$

We proceed in the next section by presenting the Central-WENO (CWENO) reconstruction which supplies the required elements, R_j, for the overall piecewise-polynomial interpolant (2.11).

2.2.2 The Central-Weno (CWENO) Reconstruction

In this section, we present our new Central-WENO (CWENO) piecewise-parabolic reconstruction which will be then utilized to construct a fourth-order method.

Let u be the exact solution at time t^n, and $\bar{u}_j^n$ the numerical approximation of its cell average on the cell I_j. Starting from the data $\{\bar{u}_j^n\}$ we apply the reconstruction scheme, obtaining the function $R(x;\bar{u}^n)$. By construction, $R(x;\bar{u}^n) = \sum_j R_j(x)\chi_j(x)$, where in this case $R_j(x)$ is a polynomial of degree 2. We require that our reconstruction satisfies the following properties:

1. Accuracy

i) Cell averages We shall require that Eq. (2.25) provides a high order accurate approximation of the staggered cell averages at time t^n, i.e.

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

where s denotes the spatial order of the method. This requirement is satisfied by imposing that the polynomial reconstruction from cell-averages must satisfy:

$$\begin{aligned} \frac{1}{2h}\int_{x_{j-1/2}}^{x_j} R(x;\bar{u}^n)dx &= \frac{1}{2h}\int_{x_{j-1/2}}^{x_j} u(x,t^n)dx + O(h^s), \\ \frac{1}{2h}\int_{x_j}^{x_{j+1/2}} R(x;\bar{u}^n)dx &= \frac{1}{2h}\int_{x_j}^{x_{j+1/2}} u(x,t^n)dx + O(h^s). \end{aligned} \tag{2.31}$$

ii) Pointwise values We look for a reconstruction, $\tilde{R}(x;\bar{u}^n)$, that satisfies the following requirements

$$\tilde{R}(x_j;\bar{u}^n) = u(x_j,t^n) + O(h^s). \tag{2.32}$$

iii) Space derivatives of the flux A separate reconstruction $\hat{R}(x; f^n)$ of f will be used for the derivatives of the fluxes, $\partial_x f(u(x,t))$, for which we require

$$\hat{R}'(x_j; f^n) = \partial_x f(u(x_j,t^n)) + O(h^{s-1}). \tag{2.33}$$

2. Conservation
   For the function reconstruction R we require

$$\frac{1}{h}\int_{I_j} R_j(x)dx = \bar{u}_j. \tag{2.34}$$

3. Non-oscillatory Reconstruction
   Avoid spurious oscillations in the sense of ENO/WENO reconstruction ([23], [47]).

As we shall see, in general different reconstructions are obtained for R and $\tilde{R}$, since the accuracy requirements are different.

In each cell, I_j, we reconstruct three polynomials of degree 2, $P_{j-1}(x)$,$P_j(x)$,$P_{j+1}(x)$. Each of these polynomials is constructed by posing the following interpolation requirements:

$$\begin{cases} \frac{1}{h}\int_{I_{k-1}} P_k(x)dx = \bar{u}_{k-1}, \\ \frac{1}{h}\int_{I_k} P_k(x)dx = \bar{u}_k, \\ \frac{1}{h}\int_{I_{k+1}} P_k(x)dx = \bar{u}_{k+1}. \end{cases} \qquad k = j-1,j,j+1 \tag{2.35}$$

The reconstruction is created by considering a convex combination of the above polynomials, $P_k(x)$,

$$R_j(x) = w_j^{-1}P_{j-1}(x) + w_j^0 P_j(x) + w_j^1 P_{j+1}(x), \tag{2.36}$$

where the weights w_j^k,$k = -1,0,1$, satisfy $w_j^k \geq 0$, and $\sum_{k=-1}^{1} w_j^k = 1$.

A basic fact of interpolation theory states that each polynomial P_k, $k = j-1, j, j+1$, is uniformly third order accurate in the cell j [58].

$$P_k(x) - u(x,t_n) = O(h^3), \quad k = j-1, j, j+1, \quad x \in I_j$$

If we consider a convex combination of these three polynomials, we have two more degrees of freedom, which may be used to impose more order conditions. For example we can impose that the pointwise value of the polynomial is fifth order accurate at a given point, or we can impose that the half-cell average of it is fifth order accurate. Classical WENO schemes are based on imposing fifth order accuracy at the edge of the cell [26].

The stencil used in the reconstruction of the second degree polynomial $R_j(x)$ contains five points. Note that this convex combination retains the interpolation requirement (2.34) for $R_j(x)$ in I_j, but otherwise does not fulfill any other interpolation requirements in the neighboring cells.

Since $deg(P_k(x)) = 2$, $k = j-1, j, j+1$, one can rewrite

$$P_k(x) = \tilde{u}_k + \tilde{u}'_k(x - x_k) + \frac{1}{2}\tilde{u}''_k(x - x_k)^2, \quad k = j-1,j,j+1. \tag{2.37}$$

The reconstructed point-values, $\tilde{u}_k$, and the reconstructed discrete first and second derivatives, $\tilde{u}'_k, \tilde{u}''_k$, are uniquely determined by the interpolation requirements (2.35), as

$$\begin{aligned}
\tilde{u}_k &= \bar{u}_k - \frac{\bar{u}_{k-1} - 2\bar{u}_k + \bar{u}_{k+1}}{24}, \\
\tilde{u}'_k &= \frac{\bar{u}_{k+1} - \bar{u}_{k-1}}{2h}, \\
\tilde{u}''_k &= \frac{\bar{u}_{k+1} - 2\bar{u}_k + \bar{u}_{k-1}}{h^2}, \qquad k = j-1, j, j+1.
\end{aligned} \tag{2.38}$$

All that is left in order to end the reconstruction is to determine the weights, $w_j^k, k = -1, 0, 1$. Two ingredients are taken into account in the construction: the accuracy requirements and the non-oscillatory requirements.

The two requirements will be satisfied by the procedure described below. Following the notations of [26], in order to guarantee convexity, $\sum_{k=-1}^{1} w_j^k = 1$, the weights, w_j^k, are written as

$$w_j^k = \frac{\alpha_j^k}{\alpha_j^{-1} + \alpha_j^0 + \alpha_j^1}, \qquad k = -1, 0, 1, \tag{2.39}$$

where

$$\alpha_j^k = \frac{C_k}{(\epsilon + \mathrm{IS}_j^k)^p}, \qquad k = -1, 0, 1. \tag{2.40}$$

The constants, C_k, ϵ, p, and the smoothness indicator, IS_j^k, will be determined below.

Since we can not satisfy all the accuracy requirements simultaneously, we split the computation into two parts and by that we are led to two different sets of constants C_k. The first set corresponds to the accuracy requirement in the reconstruction of the cell-averages (2.31), the second set of constants corresponds to the accuracy requirements in the reconstruction of the pointwise values, and the third set corresponds to the accuracy requirements in the computation of the flux derivatives (2.33). Due to cancellation, any symmetric choice of coefficients will result in a fourth-order approximation of the point-values in the center of the cells ($s = 4$ in (2.32)). In order to satisfy (2.32) for $s = 5$ one has to use non-positive constants, namely a *non-convex* combination of the stencils. The treatment of negative weights has been recently investigated by Shu [59]. We shall describe later the use of this technique.

Clearly, the use of different sets of constants imposes absolutely no problems on the implementation of the algorithm.

A straightforward computation results with the desired constants which are displayed in Table 2.1.

	C_{-1}	C_0	C_1	accuracy
cell-averages	$3/16$	$5/8$	$3/16$	h^5
derivatives	$1/6$	$2/3$	$1/6$	h^4
point-values	$-9/80$	$49/40$	$-9/80$	h^6
	any symmetric combination			h^4

Table 2.1 The constants of the Central-WENO reconstruction

By setting

$$\alpha_j^k = C_k, \qquad k=-1,0,1 \tag{2.41}$$

instead of (2.40), one obtains a linear high order scheme, which would give good accuracy in the smooth regions, and would produce oscillations near discontinuities. The nonlinear weights are used to prevent oscillations. The basic idea is that one should weight the stencils so that only stencils that do not contain the discontinuity will be used in practice in the reconstruction. This is obtained by formula (2.40) and by suitable smoothness indicators, which are a quantitative measure of the local roughness of the solution.

Several different ways to determine the smoothness indicator were suggested in the literature (see, e.g., [47] and [26]). Here we use the measure taken from [26], which amounts to a measure on the L^2 -norms of the derivatives:

$$\mathrm{IS}_j^k = \sum_{l=1}^{2} \int_{x_{j-1/2}}^{x_{j+1/2}} h^{2l-1} (P_k^{(l)})^2 dx, \qquad k=-1,0,1, \tag{2.42}$$

where $P_k^{(l)}$ denotes the l -th derivative of $P_k(x)$. An explicit integration of (2.42) yields

$$\begin{aligned}
\mathrm{IS}_j^{-1} &= \frac{13}{12}(\bar u_{j-2} - 2\bar u_{j-1} + \bar u_j)^2 + \frac{1}{4}(\bar u_{j-2} - 4\bar u_{j-1} + 3\bar u_j)^2, \\
\mathrm{IS}_j^{0} &= \frac{13}{12}(\bar u_{j-1} - 2\bar u_{j} + \bar u_{j+1})^2 + \frac{1}{4}(\bar u_{j-1} - \bar u_{j+1})^2, \\
\mathrm{IS}_j^{1} &= \frac{13}{12}(\bar u_{j} - 2\bar u_{j+1} + \bar u_{j+2})^2 + \frac{1}{4}(3\bar u_{j} - 4\bar u_{j+1} + \bar u_{j+2})^2.
\end{aligned} \tag{2.43}$$

In smooth regions, a Taylor expansion of (2.43) gives

$$\mathrm{IS}_j^k = (\tilde u' h)^2 + \frac{13}{12}(\tilde u'' h^2)^2 + O(h^6), \qquad k=-1,0,1. \tag{2.44}$$

Hence, $\mathrm{IS}_j^k = O(h^2)$, and in critical points it is $O(h^4)$. In non-smooth regions, $\mathrm{IS}_j^k = O(1)$, and by that the normalized weight of the corresponding stencil will be negligible. Therefore, our reconstruction follows the WENO methodology by automatically avoiding the information coming from non-smooth regions which are the cause for spurious oscillations.

The remaining parameters to be determined in (2.40) are ϵ and p . The constant ϵ was inserted in the denominator in order to prevent it from vanishing. In [26] an $\epsilon = 10^{-6}$ and $p=2$ were selected. We shall use the same values.

In short, our reconstruction from cell averages routine accepts in input the values $\{\bar u_j\}$ of cell averages at time t^n , and produces in output the point values u_j,u_j',u_j'' which completely determine the reconstruction polynomial $R_j(x) = u_j + u_j'(x-x_j)/h + (1/2)u_j''(x-x_j)/h^2$.

A few modifications are needed to compute the reconstruction from point values for the flux $f_j = f(u_j)$ which is needed in the Runge-Kutta step.

Here the candidate polynomials P_k satisfy the interpolation requirements (compare with (2.35)):

$$\begin{cases} P_k(x_{k-1}) = f_{k-1}, \\ P_k(x_k) = f_k, \\ P_k(x_{k+1}) = f_{k+1}. \end{cases} \qquad k = j-1, j, j+1 \tag{2.45}$$

Thus the reconstructed point values $\tilde{f}_k$ and the reconstructed first and second derivatives $\tilde{f}'_k$ and $\tilde{f}''_k$ are given by:

$$\tilde{f}_k = f_k, \quad \tilde{f}'_k = \frac{f_{k+1} - f_{k-1}}{2h}, \quad \tilde{f}''_k = \frac{f_{k+1} - 2f_k + f_{k-1}}{h^2}, \quad k = j-1, j, j+1.$$

For the evaluation of the intermediate values, all that will be needed is a pointwise reconstruction of the space derivative f'_j which is given by:

$$f'_j = w_j^{-1}(\tilde{f}'_{j-1} + h\tilde{f}''_{j-1}) + w_j^0 \tilde{f}'_j + w_j^1(\tilde{f}'_{j+1} - h\tilde{f}''_{j+1}). \tag{2.46}$$

The computation of the weights in (2.46) is the same as above (2.39)–(2.44). Here, we use the second set of constants C_k 's of Table 2.1, and the computation of the smoothness indicators IS_k^j in (2.43) involves the point values f_j , instead of the cell averages.

Three different families of smoothness indicators have to be computed: one for the cell average, one for the pointwise values and one for the flux derivative. The latter, in principle, has to be computed at each stage of the Runge-Kutta method. Numerical experiment show that it is enough to compute it only once per time step, at the beginning of each step.

We shall only construct a fourth order scheme, therefore we shall use only positive weights. For the pointwise reconstruction we use the same set of constants used for the calculation of the cell average. In this way only two families of smoothness indicators have to be computed at each time step.

2.2.3 Numerical results

In this section we test the fourth-order scheme. We start from a single scalar equation where we numerically compute the order of accuracy of our schemes. We also demonstrate on a model problem how the smoothness indicators trigger the selection of the correct stencil when discontinuities are present.

The construction of the smoothness indicators is more delicate for systems of equations than in the scalar case. A naive component by component extension of the scalar scheme does not yield the best results. Better results are obtained by using a global smoothness indicator which is the same for all components of vector u .

We then apply our scheme to some classical test problems of gas dynamics. Our results show that the use of the same smoothness indicator for all the components produces better results and is computationally cheaper compared with the componentwise indicator.

Scalar equation

As an application of the central WENO schemes we study the performance of our schemes by applying them to the Burgers equation.

$u_t + (\frac{1}{2}u^2)_x = 0,$
$u(x,t=0) = 1 + \frac{1}{2}\sin(\pi x),$
periodic boundary conditions on [-1,1],
integration times: $T = 0.33$ and $T = 1.5$.

Burgers equation, $u_0 = 1 + 1/2\sin(\pi x)$

N	L^1 error	L^1 order	L^∞ error	L^∞ order
20	0.2926E-02		0.9462E-02	
40	0.2459E-03	3.5728	0.1139E-02	3.0549
80	0.1419E-04	4.1150	0.8631E-04	3.7216
160	0.6821E-06	4.3787	0.4461E-05	4.2742
320	0.3227E-07	4.4017	0.2296E-06	4.2800
640	0.1766E-08	4.1916	0.1269E-07	4.1779

Table 2.2 Convergence test for the fourth order CWENO scheme. $T = 0.33$, $\lambda = 0.66 * 2$, $\epsilon = 10^{-6}$, $p = 2$.

Here $T = 0.33$ is used for convergence tests, and $T = 1.5$ for the shock capturing test (the shock develops at $T_s = 2/\pi$).

Convergence tests are useful when testing a nonlinear scheme. First they are used to check the correctness of the code. Second, because the schemes are nonlinear, one should check that the nonlinear weights do not destroy the accuracy.

If the expected accuracy is not reached, then it is useful to perform the check on a smooth solution with a linear scheme, where the weights equal the constant C_k. In this way one verifies whether the lack of accuracy is due to the nonlinear weights or to a programming error.

The computational parameters used in the following tests are $\epsilon = 10^{-6}$, $p = 2$. The mesh ratio was chosen as while $\lambda = 0.66\lambda_{\max}$ for the nonlinear (Burgers) equation. The parameter $\lambda_{\max}$ was computed in order to satisfy stability conditions and it depends on the scheme. We used $\lambda_{\max} = 2/7$.

First we perform an accuracy test, computing the order of accuracy at T=0.33, well before the shock formation time $T = 2/\pi$. The results appear in Table 2.2.3. We observe the correct order of accuracy.

The shock-capturing properties of the CWENO scheme are illustrated in Figure 2.7 for the Burgers equation. The pictures on the left refer to the solution before shock formation (T=0.5) and the pictures on the right refer to the solution after shock formation (T=1.5). The bottom part of the picture shows the weights w_j^k computed in the reconstruction from cell averages. In particular, the central weight corresponds to w_j^0, while the left weight corresponds to w_j^{-1}. Before the shock formation, the weights remain close to their equilibrium values (given in Table 1). An abrupt change can be seen after the shock formation. Here the stencils that would yield oscillations are assigned almost a zero weight. Thus the solution is oscillation-free even after the shock forms. The shock transition occurs within two cells.

Systems of equations

We apply our schemes to the system of Euler equations of gas dynamics for a polytropic gas with constant $\gamma = 1.4$. The variables ρ, v, E, and p, denote the density, velocity, total energy per unit volume and the pressure, respectively. We consider the test problems (2.21) and (2.22).

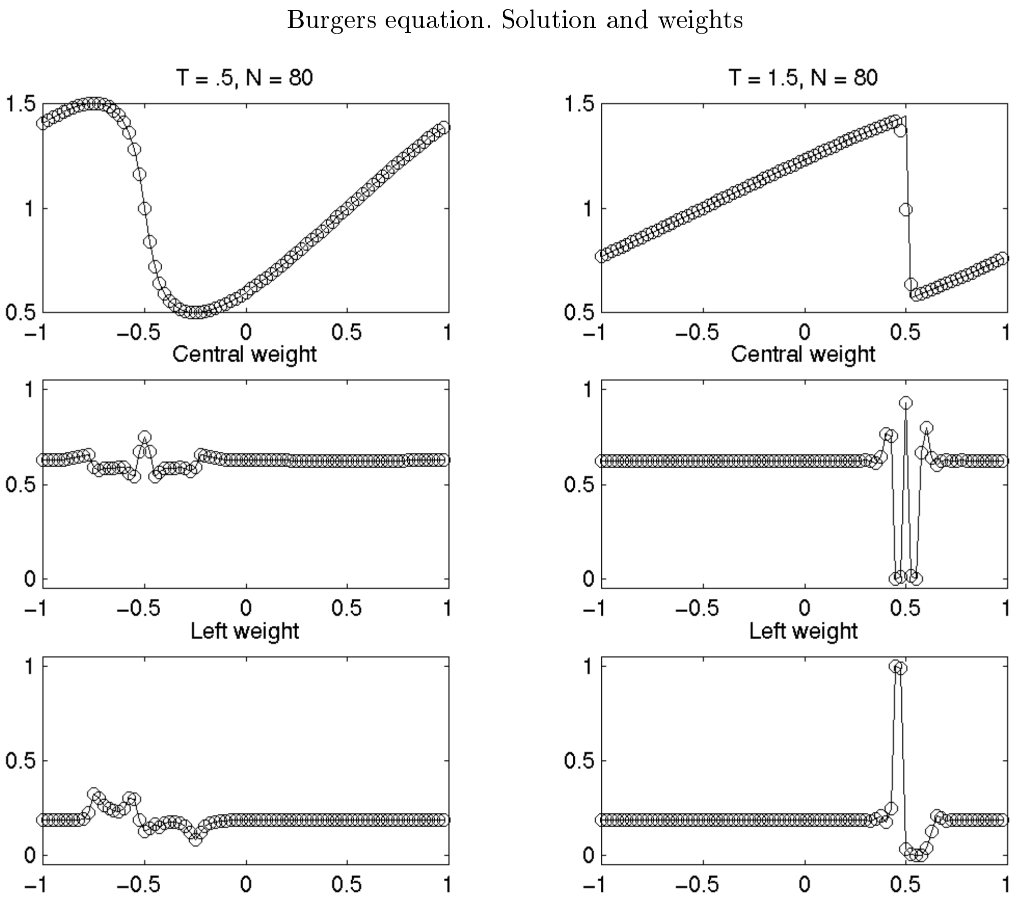

Figure 2.7 Weights computed from cell averages, $m = 4$

In both cases the computational domain is $[0,1]$; we integrate the equations up to $T = 0.16$, i.e. before the perturbations reach the boundary of the computational region. The mesh ratio here is $\lambda = 0.1$, which is the optimal one for Lax initial data.

The number of cells, N, was taken as $N = 100$ in order to compare with the upwind literature, and also as $N = 400$ to show the behavior of the weights: when the solution is well resolved, the weights coincide almost everywhere with the linear weights.

We apply one smoothness indicator for all the components:

$$\mathrm{IS}_j^k = \frac{1}{d}\sum_{r=1}^{d} \frac{1}{||\bar{u}_r||_2} \left(\sum_{l=1}^{2} \int_{x_{j-1/2}}^{x_{j+1/2}} h^{2l-1} \left(P_{j+k,r}^{(l)} \right)^2 \, dx \right), \quad k = -1, 0, 1. \tag{2.47}$$

Here d is the number of equations, and $P_{k,r}$ denotes the k-th polynomial for the r-th component. The quantity $||\bar{u}_r||_2$ is a scaling factor, and it is defined as the L^2 norm of the cell averages of the r-th component of u, namely:

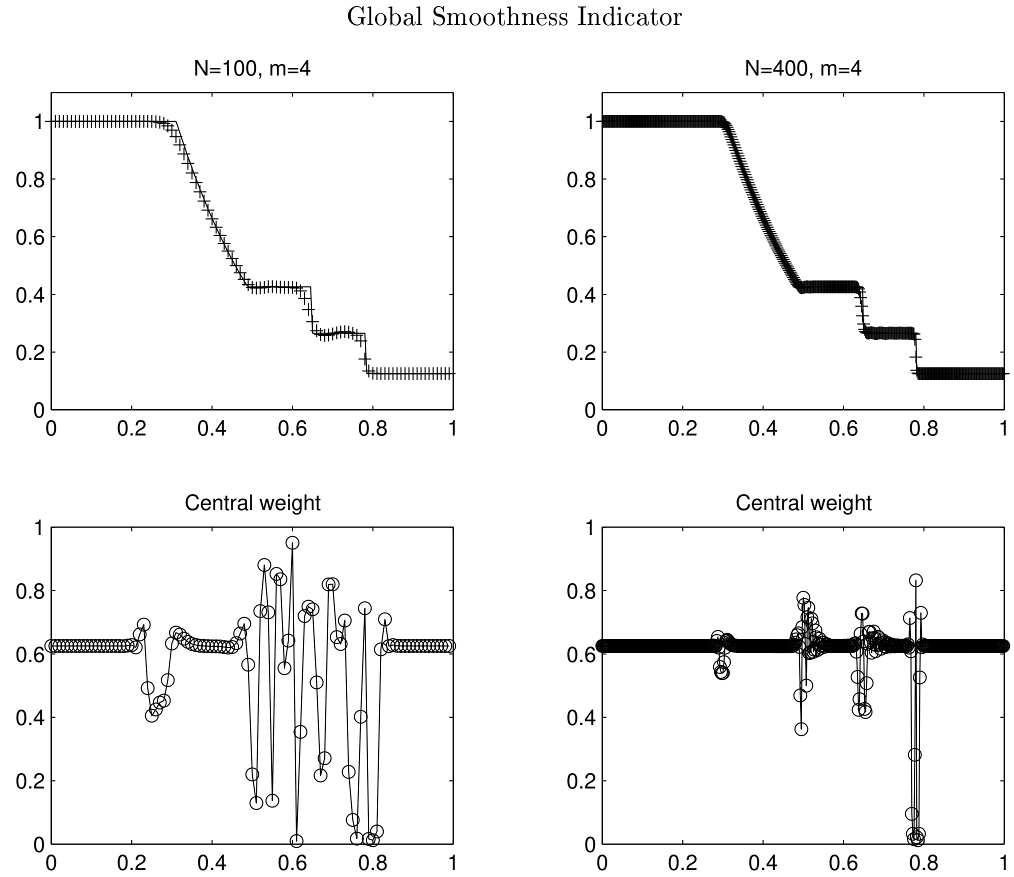

Figure 2.8 Density (top) and Central weights (bottom) for Sod's problem: $\lambda = 0.1$, $T = 0.16$. The weight shown is computed once per time step during the reconstruction from cell-averages

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

This choice is not crucial if all components are of the same order. However it would be more appropriate to use $||\bar{u}_r||_2^2$ as a scaling factor, so that the smoothness indicators will be scale-invariant with respect to a change in the physical units of the component of the field vector u .

The integral in (2.47) can be exactly integrated (see (2.43)).

This strategy will be called "Global smoothness indicators": its results on Sod's problem are shown in Figure 2.8. A comparison between componentwise and global smoothness indicators is shown in the case of the Lax test (see Figure 2.9).

To summarize, the "Global smoothness indicators" strategy is much less expensive and gives better results than the "Componentwise smoothness indicators" strategy.

Componentwise and Global Smoothness Indicators

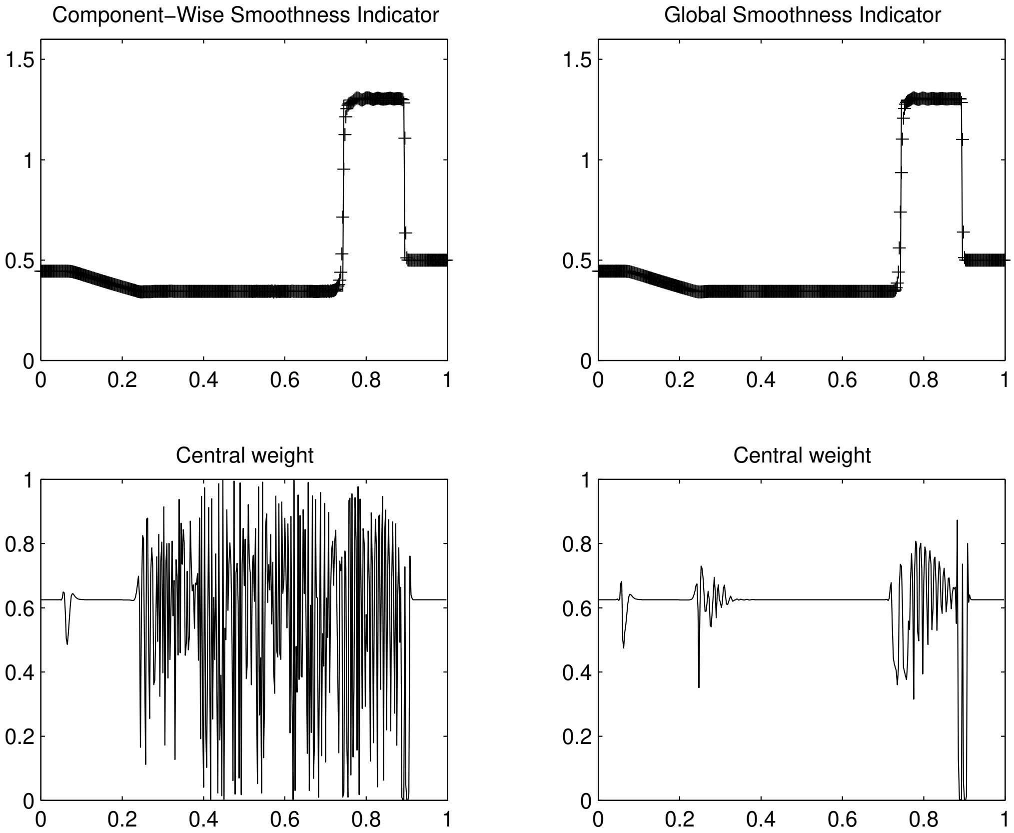

Figure 2.9 Density (top) and Central weights (bottom) for Lax' problem: $\lambda = 0.1$, $T = 0.16$, $m = 4$, $N = 400$. The weight shown is computed once per time step during the reconstruction from cell-averages for the density

2.2.4 Improvements

The Centered WENO schemes described before are not optimal for two reasons.

First, because of the restriction on the uniform accuracy of NCE of Runge Kutta methods, a fourth order RK scheme has been used to compute the predictor values for the fourth order CWENO scheme. This requires the computation of the stage values, and then the evaluation of the flux at the nodes of the quadrature formula.

Second, only fourth order CWENO is obtained with three parabola, because when looking for a fifth order scheme, negative weights appear. Here we mention how it is possible to optimize CWENO schemes, by using fourth order RK scheme for constructing a fourth order CWENO, without the additional evaluation of the flux at the nodes of the quadrature formula, and how to use a fifth order Runge-Kutta method for a fifth order scheme [52].

Let us write Eq.(2.1) in the form

$$\frac{\partial u}{\partial t} = F(u), \tag{2.48}$$

where $F = -\partial f/\partial x$. By applying a ν-stage Runge-Kutta to Eq. (2.48), one obtains an expression for the approximation of the pointwise value of $u(x,t+\Delta t)$

$$u(x,t+\Delta t) = u(x,t) + \sum_{i=1}^{\nu} b_i K^{(i)}(x), \tag{2.49}$$

$$K^{(i)}(x) = F(u(x,t) + \Delta t \sum_{l=1}^{i-1} a_{il} K^{(l)}(x)), \quad i = 1, \ldots, \nu \tag{2.50}$$

By integrating Eqs. (2.49) and (2.50) between x_j and x_{j+1} one has

$$\bar{u}^{n+1}_{j+1/2} = \bar{u}^{n}_{j+1/2} + \Delta t \sum_{i=1}^{\nu} b_i \bar{K}^{(i)}_{j+1/2} \tag{2.51}$$

$$\bar{K}^{(i)}_{j+1/2} = -\frac{1}{h}[f(u_{j+1}(t+c_i\Delta t)) - f(u_j(t+c_i\Delta t))], \quad i = 1, \ldots, \nu \tag{2.52}$$

where $u_j(t+c_i\Delta t)$ denotes the stage value of the field u at grid point x_j. The scheme can be therefore summarized as follows

- Compute the stage values (predictor)

$$u^{(i)}_j = u^n_j + \Delta t \sum_{l=1}^{i-1} a_{il} F(u^{(l)}_j) \tag{2.53}$$

- compute the new cell average (corrector)

$$\bar{u}^{n+1}_{j+1/2} = \bar{u}^{n}_{j+1/2} - \lambda \sum_{l=1}^{\nu} b_l [f(u^{(l)}_{j+1}) - f(u^{(l)}_j)] \tag{2.54}$$

CWENO reconstruction will be used to compute

- cell average, $h\bar{u}^n_{j+1/2} = \int_{x_j}^{x_{j+1/2}} R_j(x)\,dx + \int_{x_{j+1/2}}^{x_{j+1}} R_{j+1}(x)\,dx$
- pointwise values of the field u^n_j at point x_j, $u^n_j = \hat{R}_j(x_j)$
- flux derivative, $F(u^{(l)}_j) \approx -\tilde{R}'(x_j)$

Notice that, as usual, three different CWENO reconstructions will be used, according to the objective.

By using this technique, fourth order RK predictor can be used to obtain a fourth order CWENO scheme. No additional flux evaluation is needed, since the fluxes used in the corrector are computed at the stage values, and therefore $\nu - 1$ of them have already been computed during the predictor stage.

In order to obtain a fifth order scheme, then the constants $\mathbb{C} = (-9/80, 49/40, -9/80)$ have to be used for the reconstruction of the pointwise values (see table (2.1)).

Note that negative weights appear in the reconstruction of pointwise values from cell averages. Straightforward use of WENO reconstruction with negative weights would produce spurious oscillations (and possibly blow-up of the numerical solution). This problem can be overcome using a technique recently developed by Shu [59]. First, use two sets of positive constants, C_i^+, C_i^-, such that $C_i = C_i^+ - C_i^-$, $i = -1,0,1$. This is done in such a way that the constants are well bounded away from zero. Shu proposes

$$C_i^+ = \frac{1}{2}(\theta|C_i| + C_i), \quad C_i^- = \frac{1}{2}(\theta|C_i| - C_i),$$

and uses $\theta = 3$. Then compute

$$\alpha_\pm^i = \frac{C_i^\pm}{(\mathrm{IS}^i + \epsilon)^2}, \quad i = -1,0,1$$

Finally, compute the two sets of weights

$$w_+^i = \frac{\alpha_+^i}{\sum_{l=-1}^{1} \alpha_+^i} \sum_{l=-1}^{1} C_i^+, \quad w_-^i = \frac{\alpha_-^i}{\sum_{l=-1}^{1} \alpha_-^i} \sum_{l=-1}^{1} C_i^-,$$

The final reconstruction is obtained by

$$R_j(x) = \sum_{i=-1}^{1} (w_{j,+}^i - w_{j,-}^i) P_j^i$$

2.2.5 Semidiscrete central schemes

One of the drawbacks of NT and many other central schemes is staggering. There is a non-staggered version of the scheme, which is developed in [24].

The central schemes we have considered so far are based on a simultaneous discretization in space and time. There is a procedure to derive semidiscrete central schemes, which are discretized only in space. The discretization in time can then be performed by a suitable ODE solver (method of lines). A second order semidiscrete central scheme has been derived by Kurganov and Tadmor [32]. The derivation is based on a different treatment of the the region which lies in the characteristic fan from the region of the mesh which lies in the smooth region. The authors derive a fully discrete second order scheme, which is less dissipative that Nessyahu-Tadmor scheme, and then, by letting the time step vanish, they deduce a semidiscrete scheme. When applied to conservation laws, the scheme is basically equivalent to a second order monotone finite volume scheme with local Lax-Friedrichs flux function. The authors apply the scheme to several problems containing a source term, such as convection diffusion equations. However, they are not interested in the treatment of very stiff sources. A third order extension of the scheme is carried out in [31], and applied to several balance laws. A we shall remark later, semidiscrete central schemes can be used as building blocks for the construction of high order schemes for systems with stiff source.

2.3 Multidimensional central schemes

In this section we describe how to derive second and higher order central schemes, and we show some applications to gas dynamics.

Central schemes can be extended to problems in several dimensions. Second order central schemes on rectangular grids have been considered by Jiang and Tadmor [25], Sanders and Weiser [57], and by Arminjon et al. [6]. They have been extended to unstructured grids by Arminjon et al. [7]. High order central schemes in two dimensions have been considered in the papers [39], [40], [41]

Consider the two-dimensional system of conservation laws

$$u_t + f(u)_x + g(u)_y = 0, \tag{2.55}$$

subject to the initial values

$$u(x,y,t=0) = u_0(x,y),$$

and to boundary conditions, which we do not specify at this point. The flux functions f and g are smooth vector valued functions, $f,g : \mathbb{R}^d \to \mathbb{R}^d$. The system (2.55) is assumed to be hyperbolic in the sense that for any unit vector $(n_x,n_y) \in \mathbb{R}^2$, the matrix $n_x \nabla_u f + n_y \nabla_u g$ has real eigenvalues and its eigenvectors form a basis of $\mathbb{R}^d$. In order to integrate numerically (2.55), we introduce a rectangular grid which for simplicity will be assumed to be uniform with mesh sizes $h = \Delta x = \Delta y$ in both directions. We will denote by $I_{i,j}$ the cell centered around the grid point $(x_i,y_j) = (i\Delta x, j\Delta y)$, i.e., $I_{i,j} = [x_i - h/2, x_i + h/2] \times [y_j - h/2, y_j + h/2]$. Let Δt be the time step and denote by $u^n_{i,j}$ the approximated point-value of the solution at the (i,j) -th grid point at time $t^n = n\Delta t$. Finally, let $\bar{u}^n_{i,j}$ denote the cell average of a function u evaluated at the point (x_i,y_j) ,

$$\bar{u}^n_{i,j} = \frac{1}{h^2}\int_{I_{i,j}} u(x,y,t^n)\, dx\, dy.$$

Given the cell-averages $\{\bar{u}^n_{i,j}\}$ at time t^n , Godunov-type methods provide the cell-averages at the next time-step, t^{n+1} , in the following way: first, a piecewise-polynomial reconstruction is computed from the data $\{\bar{u}^n_{i,j}\}$ resulting with

$$u^n(x,y) = \sum_{i,j} R_{i,j}(x,y)\chi_{i,j}(x,y). \tag{2.56}$$

Here, $R_{i,j}(x,y)$ is a suitable polynomial (which has to satisfy conservation, accuracy and non-oscillatory requirements), while $\chi_{i,j}(x,y)$ is the characteristic function of the cell $I_{i,j}$. Thus, in general, the function $u^n(x,y)$ will be discontinuous along the boundaries of each cell $I_{i,j}$.

In order to proceed, the reconstruction, $u^n(x,y)$, is evolved according to some approximation of (2.55) for a time step Δt . We will use the fact that the solution remains smooth at the vertical edges of the staggered control volume, $I_{i+1/2,j+1/2} \times [t^n,t^{n+1}]$, provided that the time-step Δt satisfies the CFL condition

$$\Delta t < \frac{h}{2}\frac{1}{\max(|\sigma_x|,|\sigma_y|)}.$$

Here, $I_{i+1/2,j+1/2} = [x_i,x_{i+1}] \times [y_j,y_{j+1}]$ (see figure 2.10; the edges at which the solution remains smooth are denoted by dotted vertical lines), and σ_x and σ_y are the largest (in modulus) eigenvalues of the Jacobian of f and g, respectively.

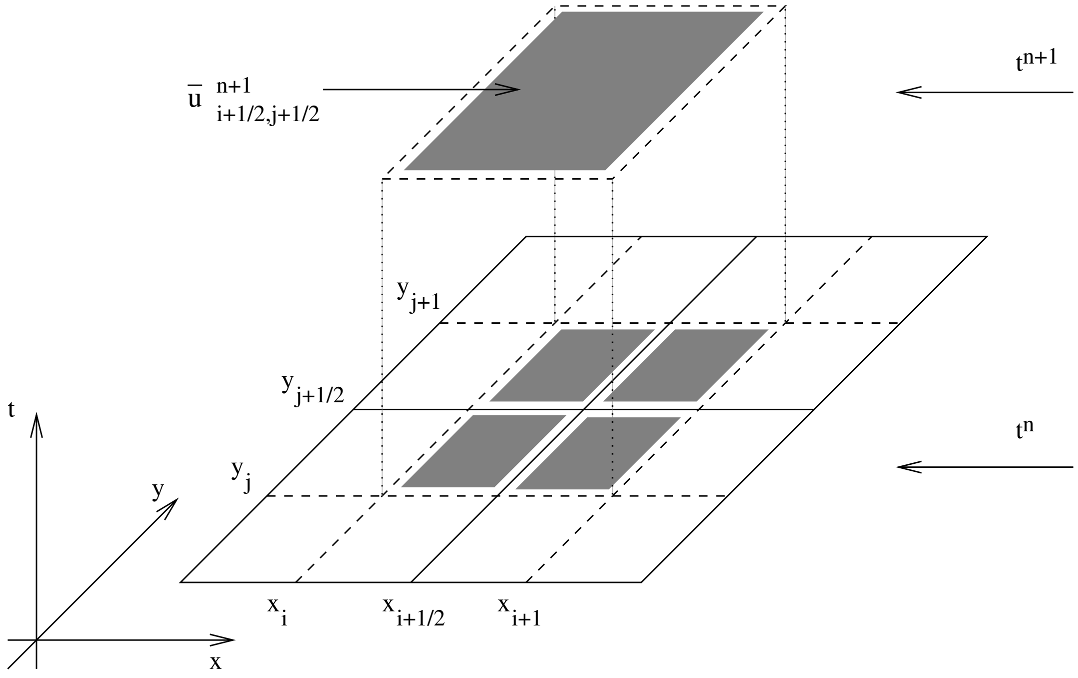

Figure 2.10 The Two-Dimensional Stencil

An exact integration of the system (2.55) with data $u^n(x,y)$ over the control volume $I_{i+1/2,j+1/2} \times [t^n,t^{n+1}]$ results with

$$\begin{aligned} \bar{u}^{n+1}_{i+\frac{1}{2},j+\frac{1}{2}} &= \frac{1}{h^2}\int\int_{I_{i+\frac{1}{2},j+\frac{1}{2}}} u^n(x,y)\,dx\,dy \\ &- \frac{1}{h^2}\int_{\tau=t^n}^{t^{n+1}} \left\{ \int_{y=y_j}^{y_{j+1}} \left[f\left(u(x_{i+1},y,\tau)\right) - f\left(u(x_i,y,\tau)\right)\right] dy \right\} d\tau \\ &- \frac{1}{h^2}\int_{\tau=t^n}^{t^{n+1}} \left\{ \int_{x=x_i}^{x_{i+1}} \left[g\left(u(x,y_{j+1},\tau)\right) - g\left(u(x,y_j,\tau)\right)\right] dx \right\} d\tau. \end{aligned} \tag{2.57}$$

The first integral on the RHS of (2.57) is the cell-average of the function $u^n(x,y)$ on the staggered cell $I_{i+1/2,j+1/2}$. Given the reconstructed function $u^n(x,y)$, (2.56), this term can be computed exactly: it will consist of a contribution of four terms, resulting from averaging $R_{i+1,j+1}(x,y)$, $R_{i,j+1}(x,y)$, $R_{i+1,j}(x,y)$, and $R_{i,j}(x,y)$, on the corresponding quarter-cells.

The advantage of the central framework appears in the evaluation of the time integrals appearing in (2.57). Since the solution remains smooth on the segments $(x_i,y_j) \times [t^n,t^{n+1}]$, we can evaluate the time integrals with a quadrature rule using only nodes lying in these segments.

A second order scheme is obtained by approximating the integral of the flux f as

$$\int_{\tau=t^n}^{t^{n+1}} \int_{y=y_j}^{y_{j+1}} f(x_i,y,\tau)\, dy\, d\tau \approx \frac{h\Delta t}{2} \left(f(u_{i,j}^{n+1/2}) + f(u_{i,j+1}^{n+1/2}) \right) \tag{2.58}$$

and likewise for the integral of the flux g. By applying the same discretization used for the Nessyahu-Tadmor scheme, one obtains the two dimensional counterpart, which has been introduced in [6] and, independently, in [25]. First, in each cell (i,j) the field u is reconstructed by a piecewise linear approximation,

$$L_{i,j}(x,y) = \bar{u}_{i,j}^n + u'_{i,j} \frac{x - x_i}{h} + u^{\backprime}_{i,j} \frac{y - y_j}{h}$$

where u'/h and $u^{\backprime}/h$ denote, respectively, first order approximations of x- and y-partial derivatives, and can be computed by using a suitable slope limiter, as in the one dimensional case. The resulting scheme has a compact form similar to the one dimensional one, and can be written as

$$\begin{aligned} \bar{u}_{i+1/2,j+1/2}^{n+1} \quad = \quad & \bar{u}_{i+1/2,j+1/2}^{n} \\ & -\frac{\lambda}{2}(f(u_{i+1,j}^{n+1/2}) + f(u_{i+1,j+1}^{n+1/2}) - f(u_{i,j}^{n+1/2}) - f(u_{i,j+1}^{n+1/2}) \\ & -\frac{\lambda}{2}(g(u_{i,j+1}^{n+1/2}) + g(u_{i+1,j+1}^{n+1/2}) - g(u_{i,j}^{n+1/2}) - g(u_{i+1,j}^{n+1/2}) \end{aligned} \tag{2.59}$$

where $\lambda = \Delta t/h$, and

$$\begin{aligned} \bar{u}_{i+1/2,j+1/2}^{n} \quad = \quad & \frac{1}{4}(\bar{u}_{i,j}^n + \bar{u}_{i+1,j}^n + \bar{u}_{i+1,j+1}^n + \bar{u}_{i,j+1}^n) + \frac{1}{16}(u'_{i,j} - u'_{i+1,j} \\ & + u'_{i,j+1} - u'_{i+1,j+1} + u^{\backprime}_{i,j} - u^{\backprime}_{i,j+1} + u^{\backprime}_{i+1,j} - u^{\backprime}_{i+1,j+1}) \end{aligned} \tag{2.60}$$

and the predictor values are evaluated as

$$u_{i,j}^{n+1/2} = \bar{u}_{i,j}^n - \frac{\lambda}{2} f'_{i,j} - \frac{\lambda}{2} f^{\backprime}_{i,j} \tag{2.61}$$

Once again, the first order approximation $f'_{i,j}$ and $g^{\backprime}_{i,j}$ can be computed either by a slope limiter acting on $f(\bar{u}_{i,j})$ and $g(\bar{u}_{i,j})$ or by

$$f' = A(\bar{u}_{i,j}) u'_{i,j}, \quad g^{\backprime} = B(\bar{u}_{i,j}) u^{\backprime}_{i,j},$$

where A and B are the Jacobian matrices $A = \nabla_u f$, $B = \nabla_u g$. The above second order central scheme is very simple to use, and it has been successfully applied to a large variety of problems, such as double Mach reflection [25], MHD equations [65]. Modifications and improvements of the above second order central scheme have been considered by several authors (see [28, 56, 43]).

Two dimensional central schemes have been extended to unstructured grids [7]. Two dimensional central schemes on unstructured grids are very flexible, and they provide good resolution even for complex flows. Theoretical convergence results for such schemes are available [8, 20].

Here we consider the development of high order central schemes in two space dimensions. For a fourth-order method one can use Simpson's rule for the time integrals

$$\int_{t^n}^{t^{n+1}} f(u(x_i,y_j,\tau))d\tau = \tag{2.62}$$
$$= \frac{\Delta t}{6}\left[f\left(u^n_{i,j}\right) + 4f\left(u^{n+1/2}_{i,j}\right) + f\left(u^{n+1}_{i,j}\right)\right] + O\left((\Delta t)^5\right),$$

and the following centered quadrature rule in space for the integrals in space,

$$\int_{x_i}^{x_{i+1}} f(x)dx = \frac{h}{24}[-f(x_{i+2}) + 13f(x_{i+1}) + 13f(x_i) - f(x_{i-1})] + O(h^5). \tag{2.63}$$

In this way, the quadrature rule for approximating the integrals of the fluxes involves only nodes on the segments $(x_i,y_j)\times[t^n,t^{n+1}]$.

The quadrature in time, (2.62), requires the prediction of the values of the solution at later times. In the case of Simpson's rule, one has to generate the values of $u_{i,j}$ at times $t^{n+1/2}$, t^{n+1}. (The point-value $u^n_{i,j}$ can be obtained directly from the reconstruction $u_{i,j}(t^n) = u^n(x_i,y_j)$). Once again we use the smoothness of the numerical solution along the segments $(x_i,y_j)\times[t^n,t^{n+1}]$ to consider the sequence of Cauchy problems

$$\begin{cases} v'_{i,j}(\tau) = F(\tau,v_{i,j}(\tau)) := -f_x(v(x_i,y_j,t^n+\tau)) - g_y(v(x_i,y_j,t^n+\tau)), \\ v_{i,j}(\tau=0) = u^n(x_i,y_j). \end{cases} \tag{2.64}$$

In order to obtain the mid-values at $t^{n+1/2}$ and t^{n+1}, all that is required is to solve (2.64) up to these times using a Runge-Kutta scheme. When more than one intermediate value is required (as in the case of Simpson's rule), it is possible to solve (2.64) once with the largest time required, and then reconstruct the other values with the required accuracy using the Natural Continuous Extension (NCE), [70]. More details will follow below.

2.3.1 Multidimensional CWENO reconstruction

In this section we will describe in detail our new reconstruction step. We start with the reconstruction from cell-averages, (2.56), which is needed at the beginning of each time-step. We then proceed with the reconstruction from point-values which is used for evaluating the fluxes in the RK step (2.64). This section ends with a discussion of the modifications to the algorithm which are required for solving systems of equations.

2.3.2 The Reconstruction from Cell Averages

In every cell $I_{i,j}$ we reconstruct a bi-quadratic polynomial, $R_{i,j}(x,y)$, which is written as a convex combination of nine bi-quadratic polynomials, $P_{i,j}(x,y)$, centered in the cells around $I_{i,j}$,

$$R_{i,j}(x,y) = \sum_{l,k=-1}^{1} w_{i,j}^{l,k} P_{i+l,j+k}(x,y). \tag{2.65}$$

The bi-quadratic polynomials $P_{i,j}(x,y)$, which serve as the building blocks for the reconstruction (2.65), interpolate the data $\{\bar{u}^n\}$ in the sense of cell averages (see below). They approximate the function $u(x,y)$ whose cell averages are $\{\bar{u}^n\}$ with third order accuracy. The combination (2.65) is designed to increase accuracy *and* to avoid spurious oscillations.

The weights $w_{i,j}^{l,k}$ in (2.65) are computed using a nonlinear algorithm which satisfy a positivity requirement, $w_{i,j}^{l,k} \geq 0$, and a conservation requirement, $\sum_{l,k=-1}^{1} w_{i,j}^{l,k} = 1$.

For simplicity of notation, let us introduce the 3×3 matrices:

$$(\Omega_{i,j})_{l,k} = w_{i,j}^{l,k}, \qquad l,k = -1,0,1.$$

Thus each matrix $\Omega_{i,j}$ contains the nine non-constant weights needed to compute the reconstruction on the cell $I_{i,j}$. Note that the first index, l, is associated with the x-variable, while the index k is associated with the y-variable.

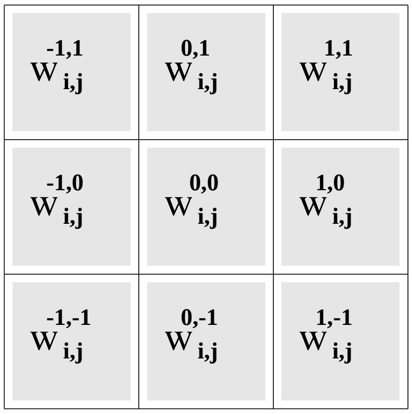

Figure 2.11 The Weight Matrix $\Omega_{i,j}$

Let $I_{i,j}^m$, $m = 1,\ldots,4$, denote the four quarters of the cell $I_{i,j}$, with $I_{i,j}^1$ being the upper-right quarter while the other three quarters are numbered clock-wise (see figure 2.12). In order to obtain a fourth-order computation of the first term on the RHS of (2.57), the reconstructed polynomial, $R_{i,j}(x,y)$, must recover the averages over the four quarter cells with fourth-order accuracy,

$$\bar{R}_{i,j}^{(m)} := \frac{1}{4h^2} \int_{I_{i,j}^m} R_{i,j}(x,y,t^n)dxdy = \frac{1}{4h^2} \int_{I_{i,j}^m} u(x,y,t^n) + O(h^4), \quad m = 1,\ldots,4, \tag{2.66}$$

where $u(x,y,t^n)$ denotes the exact solution of the equation at time t^n. On the other hand, the derivatives of the fluxes should be recovered with third-order accuracy. We therefore need to accurately evaluate the intermediate values, $u(x,y,t^n + \beta_i \Delta t)$, with $\beta_1 = 1/2$ and $\beta_2 = 1$, (see (2.26),(2.28) and (2.62)), and in particular, we need an

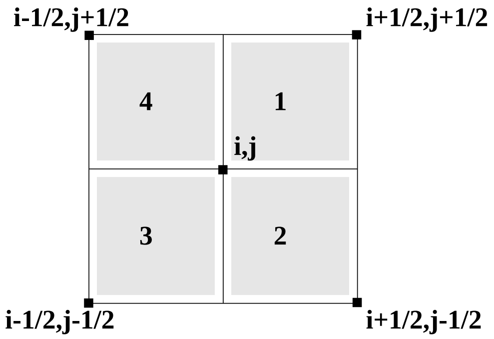

Figure 2.12 The Quarter Cells

accurate reconstruction of the point-values of the solution at the integer grid points (i,j) at time t^n.

The output of the reconstruction routine from cell averages at the beginning of the time step must therefore provide a fourth-order approximation of

(a) the four quarter-cell averages

$$\bar{R}^{(m)}_{i,j} = \frac{1}{4h^2} \sum_{l,k=-1}^{1} w^{l,k}_{i,j} \int_{I^m_{i,j}} P_{i+l,j+k}(x,y)\, dxdy, \qquad m = 1,\dots,4. \tag{2.67}$$

(b) the point-values at the integer grid-points

$$R_{i,j}(x_i,y_j) = \sum_{l,k=-1}^{1} w^{l,k}_{i,j} P_{i+l,j+k}(x_i,y_j). \tag{2.68}$$

The reconstruction routine from point values called at each evaluation of the Runge-Kutta fluxes must provide a third order approximation of the derivatives of the flux at the integer grid points

$$R^x_{i,j}(x_i,y_j) = \sum_{l,k=-1}^{1} w^{l,k}_{i,j} \partial_x P_{i+l,j+k}(x_i,y_j), \tag{2.69}$$

$$R^y_{i,j}(x_i,y_j) = \sum_{l,k=-1}^{1} w^{l,k}_{i,j} \partial_y P_{i+l,j+k}(x_i,y_j), \tag{2.70}$$

where the polynomials $P_{i+l,j+k}$ interpolate the data $f(u(\cdot,\cdot))$ in (2.69), while in (2.70) the polynomials $P_{i+l,j+k}$ interpolate the data $g(u(\cdot,\cdot))$. Generally, the weights $w^{l,k}_{i,j}$ in (2.67) and in (2.68) will be different from the weights in (2.69) and (2.70) due to the different accuracy requirements.

We would like to stress that there is no need to explicitly compute all the coefficients of the polynomial $R_{i,j}(x,y)$. All that is needed are the point-values and the quarter-cell averages of these polynomials, or their derivatives at the grid points.

We are now ready to present the construction of the fundamental bi-quadratic polynomials, $P_{i,j}(x,y)$.

2.3.3 The Bi-Quadratic Polynomials

In this section we explicitly give the coefficients of the interpolating polynomials $P_{i,j}(x,y)$, which serve as the building block for the reconstruction of $R_{i,j}(x,y)$ in (2.65). In each cell, $I_{i,j}$, we write the polynomial $P_{i,j}(x,y)$ as

$$\begin{aligned} P_{i,j}(x,y) &= b_0 + b_1(x-x_i) + b_2(y-y_j) + b_3(x-x_i)(y-y_j) + \\ &\quad + b_4(x-x_i)^2 + b_5(y-y_j)^2 + b_6(x-x_i)^2(y-y_j) + \\ &\quad + b_7(x-x_i)(y-y_j)^2 + b_8(x-x_i)^2(y-y_j)^2, \end{aligned} \tag{2.71}$$

where for simplicity we have omitted the indices (i,j) from the coefficients $\{b_m\}$. The nine coefficients b_m are uniquely determined by the interpolation conditions

$$\frac{1}{h^2}\int_{x_i-\frac{h}{2}+lh}^{x_i+\frac{h}{2}+lh}\int_{y_j-\frac{h}{2}+kh}^{y_j+\frac{h}{2}+kh} P_{i,j}(x,y)dydx = \bar{u}_{i+l,j+k}, \qquad l,k=-1,0,1,$$

i.e., the polynomials $P_{i,j}(x,y)$ interpolate the data $\{\bar{u}_{i,j}\}$ in the sense of cell-averages. The resulting expressions of the coefficients are

$$\begin{aligned} & b_0 = \bar{u} - \tfrac{h^2}{24}(\hat{u}_{xx}+\hat{u}_{yy}) + \tfrac{h^4}{24^2}\hat{u}_{xxyy}, && b_1 = \hat{u}_x - \tfrac{h^2}{24}\hat{u}_{xyy}, \\ & b_2 = \hat{u}_y - \tfrac{h^2}{24}\hat{u}_{xxy}, && b_3 = \hat{u}_{xy}, \\ & b_4 = \tfrac{1}{2}\hat{u}_{xx} - \tfrac{h^2}{48}\hat{u}_{xxyy}, && b_5 = \tfrac{1}{2}\hat{u}_{yy} - \tfrac{h^2}{48}\hat{u}_{xxyy}, \\ & b_6 = \tfrac{1}{2}\hat{u}_{xxy}, && b_7 = \tfrac{1}{2}\hat{u}_{xyy}, \\ & b_8 = \tfrac{1}{4}\hat{u}_{xxyy}, && \end{aligned}$$

where the following notation for divided differences was used,

$$\begin{aligned} \hat{u}_{x_{i,j}} &= \frac{\bar{u}_{i+1,j}-\bar{u}_{i-1,j}}{2h}, \qquad \hat{u}_{y_{i,j}} = \frac{\bar{u}_{i,j+1}-\bar{u}_{i,j-1}}{2h}, \\ \hat{u}_{xx_{i,j}} &= \frac{\bar{u}_{i+1,j}-2\bar{u}_{i,j}+\bar{u}_{i-1,j}}{h^2}, \qquad \hat{u}_{yy_{i,j}} = \frac{\bar{u}_{i,j+1}-2\bar{u}_{i,j}+\bar{u}_{i,j-1}}{h^2}, \\ \hat{u}_{xy_{i,j}} &= \frac{\bar{u}_{i+1,j+1}-\bar{u}_{i+1,j-1}-\bar{u}_{i-1,j+1}+\bar{u}_{i-1,j-1}}{4h^2}, \\ \hat{u}_{xyy_{i,j}} &= \frac{(\bar{u}_{i+1,j+1}-2\bar{u}_{i+1,j}+\bar{u}_{i+1,j-1})-(\bar{u}_{i-1,j+1}-2\bar{u}_{i-1,j}+\bar{u}_{i-1,j-1})}{2h^3}, \\ \hat{u}_{xxy_{i,j}} &= \frac{(\bar{u}_{i+1,j+1}-2\bar{u}_{i,j+1}+\bar{u}_{i-1,j+1})-(\bar{u}_{i+1,j-1}-2\bar{u}_{i,j-1}+\bar{u}_{i-1,j-1})}{2h^3}, \\ \hat{u}_{xxyy_{i,j}} &= \frac{1}{h^4}\Big[(\bar{u}_{i+1,j+1}-2\bar{u}_{i+1,j}+\bar{u}_{i+1,j-1})-2(\bar{u}_{i,j+1}-2\bar{u}_{i,j}+\bar{u}_{i,j-1}) + \\ &\quad +(\bar{u}_{i-1,j+1}-2\bar{u}_{i-1,j}+\bar{u}_{i-1,j-1})\Big]. \end{aligned}$$

2.3.4 The Weights

The weights $w_{i,j}^{l,k}$ in the reconstruction (2.65) are computed following the WENO/CWENO ideas presented in [47], [26] and [38]. The goal is to choose weights such that

(a) In smooth regions maximum accuracy is obtained.

(b) In non-smooth regions, information coming from non-smooth stencils should be switched off in order to prevent the onset of spurious oscillations.

In order to achieve these goals, the weights $w_{i,j}^{l,k}$ are written as

$$w_{i,j}^{l,k} = \frac{\alpha_{i,j}^{l,k}}{\sum_{l,k=-1}^{1} \alpha_{i,j}^{l,k}}, \tag{2.72}$$

where

$$\alpha_{i,j}^{l,k} = \frac{C^{l,k}}{(\epsilon + \mathrm{IS}_{i,j}^{l,k})^p}. \tag{2.73}$$

Here, $C^{l,k}$ are the constants which are chosen in order to maximize accuracy in smooth regions, $\mathrm{IS}_{i,j}^{l,k}$ are the "smoothness indicators" (see below), p is a constant and ϵ is introduced in order to prevent division by zero. Following previous works (e.g. [38]), in all our numerical experiments we use $p = 2$ and $\epsilon = 10^{-6}$.

The "smoothness indicators", $\mathrm{IS}_{i,j}^{l,k}$, are designed to measure the smoothness (or, more precisely, the roughness) of the polynomials $P_{i+l,j+k}$ in the cell $I_{i,j}$. This is done by evaluating a suitable function of the norms of the derivatives of the polynomial on the cell $I_{i,j}$, namely

$$\mathrm{IS}_{i,j}^{l,k} = \int_{I_{i,j}} \left(|\partial_x P_{i+l,j+k}|^2 + |\partial_y P_{i+l,j+k}|^2 + h^2 |\partial_{xx}^2 P_{i+l,j+k}|^2 + h^2 |\partial_{yy}^2 P_{i+l,j+k}|^2 \right) \, dx\, dy. \tag{2.74}$$

The integrals in (2.74) can be computed exactly, but they involve a large number of function evaluations. We compute the integrals were evaluated with a Gaussian quadrature with four nodes on the rectangle $I_{i,j}$.

All is left is to compute are the constants $C^{l,k}$ in (2.73).

We seek the values of a set of constants, $C^{l,k}$, such that the integral of the reconstruction on each quarter-cell is fourth-order accurate.

We start with the upper-right quarter cell and use symmetry considerations to label $C^{l,k}$ as $q_1, \dots q_6$, such that, $C^{1,1} = q_1$, $C^{1,0} = C^{0,1} = q_2$, $C^{-1,1} = C^{1,-1} = q_3$, $C^{-1,0} = C^{0,-1} = q_4$, $C^{-1,-1} = q_5$ and $C^{0,0} = q_6$, (see figure 2.13). Since $C^{l,k} \geq 0$ and $\sum_{l,k=-1}^{1} C^{l,k} = 1$, we have

$$q_6 = 1 - q_1 - 2q_2 - 2q_3 - 2q_4 - q_5, \qquad q_m \geq 0, \quad m = 1, \cdots, 6.$$

Imposing the accuracy requirements (2.66) for the upper-right quarter cell, results with the following system

$$\begin{cases} q_2 = -q_1 + q_4 + q_5, \\ \\ q_3 = \frac{3}{16} - q_4 - q_5, \end{cases}$$

while q_1,q_4 and q_5 remain arbitrary.

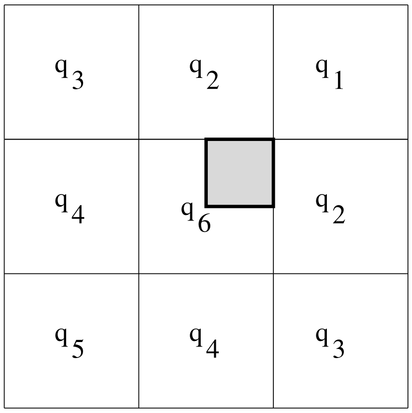

Figure 2.13 The Nine Weights

One possible solution is to choose $q_1 = q_4 = q_5 = \frac{1}{16}$, from which it follows that $q_2 = q_3 = \frac{1}{16}$ and $q_6 = \frac{1}{2}$. This gives a symmetric combination which can be therefore used for all four quarter-cell averages (and not only for the upper-right quarter cell),

$$\mathbb{C} = \begin{pmatrix} 1/16 & 1/16 & 1/16 \\ 1/16 & 1/2 & 1/16 \\ 1/16 & 1/16 & 1/16 \end{pmatrix}. \tag{2.75}$$

By symmetry, this specific choice of $\mathbb{C}$ also gives fourth-order accuracy for the computation of point-values at the center of the cell.

As in the one dimensional case, the smoothness indicators are computed only once, at the beginning of each time step.

2.3.5 The Reconstruction of Flux Derivatives

In order to compute each RK flux in (2.64), it is necessary to evaluate the function

$$F(u)_{i,j} := -f_x(u) - g_y(u)\Big|_{i,j}, \tag{2.76}$$

where u is evaluated at each intermediate time $t_i = t^n + \Delta t\, c_i$ of the RK scheme, (2.26). It is therefore necessary to compute the intermediate values of u:

$$u_{i,j}^{(l)} = u_{i,j}^n + \Delta t \sum_{k=1}^{l-1} a_{l,k} K^{(k)}, \quad l = 1, \dots, \nu. \tag{2.77}$$

Given the intermediate values in (2.77) we can evaluate $f(u_{i,j}^{(l)})$ and $g(u_{i,j}^{(l)})$, which can be then used to compute the discrete derivatives of f and g required in (2.76). These derivatives can be calculated using a procedure which is equivalent to the reconstruction procedure that was used earlier. This time, however, we require that the point-values of the derivative of the reconstruction will be third-order accurate. For simplicity, assume that we start with the function $u_{i,j}$. As before, we write the final reconstruction as a convex combination of interpolating polynomials (compare with (2.65)),

$$R_{i,j}(x,y) = \sum_{l,k=-1}^{1} \tilde{w}_{i,j}^{l,k} \tilde{P}_{i+l,j+k}(x,y). \tag{2.78}$$

This time, the polynomials interpolate the data in the sense of point-values

$$\tilde{P}_{i,j}(x_{i+l},y_{j+k}) = \tilde{u}_{i+l,j+k} \quad l,k = -1,0,1.$$

where $\tilde{u}_{i+l,j+k}$ denotes either $f(u_{i+l,j+k})$ or $g(u_{i+l,j+k})$.

The non-constant coefficients in (2.78) are (compare with (2.72)),

$$\tilde{w}_{i,j}^{l,k} = \frac{\tilde{\alpha}_{i,j}^{l,k}}{\sum_{l,k=-1}^{1} \tilde{\alpha}_{i,j}^{l,k}},$$

where for the derivative in the x-direction, one has

$$\tilde{\alpha}_{i,j}^{l,k} = \frac{\tilde{C}_x^{l,k}}{(\epsilon + \mathrm{IS}_{i,j}^{l,k})^p},$$

and a similar expression holds for the derivative in the y-direction. The smoothness indicators are the same as those computed at the beginning of the time-step. Since we are interested in an accurate reconstruction of the derivatives in (2.76), the constants $\tilde{C}_x^{l,k}$ must be chosen in order to satisfy

$$|\partial_x R_{i,j} - u_x(x_i,y_j)| = O(h^3).$$

A straightforward computation results with the possible choice of $\tilde{C}_x^{l,k}$ as

$$\tilde{C}_x^{l,k} = \begin{pmatrix} 0 & 0 & 0 \\ 1/6 & 2/3 & 1/6 \\ 0 & 0 & 0 \end{pmatrix}. \tag{2.79}$$

For the y-derivative one can choose the transpose of (2.79), $\tilde{C}_y^{l,k} = (\tilde{C}_x^{l,k})^t$. With this choice, the mixed terms of the bi-quadratic polynomials do not play any role and the differentiation formulas become very simple:

$$\begin{aligned} \left.\frac{\partial R_{i,j}}{\partial x}\right|_{(x_i,y_j)} &= \sum_{l=-1}^{1} \tilde{w}_{i,j}^{l,0} \left.\frac{\partial \tilde{P}_{i+l,j}}{\partial x}\right|_{(x_i,y_j)} \\ \left.\frac{\partial R_{i,j}}{\partial y}\right|_{(x_i,y_j)} &= \sum_{k=-1}^{1} \tilde{w}_{i,j}^{0,k} \left.\frac{\partial \tilde{P}_{i,j+k}}{\partial x}\right|_{(x_i,y_j)} \end{aligned} \tag{2.80}$$

2.3.6 The Algorithm

We would like to summarize the different stages of the algorithm obtained in the previous sections. Given $\bar{u}_{i,j}^n$, compute $\bar{u}_{i+1/2,j+1/2,}^{n+1}$ according to (2.57), i.e.,

$$\bar{u}^{n+1}_{i+1/2,j+1/2} = \mathcal{I}_1 + \mathcal{I}_2,$$

where

$$\mathcal{I}_1 = \frac{1}{h^2}\int\int_{I_{i+\frac{1}{2},j+\frac{1}{2}}} u^n(x,y)\,dx\,dy,$$

and

$$\begin{aligned}\mathcal{I}_2 = \quad & - \frac{1}{h^2}\int_{\tau=t^n}^{t^{n+1}}\left\{\int_{y=y_j}^{y_{j+1}}\left[f\left(u(x_{i+1},y,\tau)\right) - f\left(u(x_i,y,\tau)\right)\right]dy\right\}d\tau \\ & - \frac{1}{h^2}\int_{\tau=t^n}^{t^{n+1}}\left\{\int_{x=x_i}^{x_{i+1}}\left[g\left(u(x,y_{j+1},\tau)\right) - g\left(u(x,y_j,\tau)\right)\right]dx\right\}d\tau.\end{aligned}$$

$\mathcal{I}_1$ is the sum of the four quarter-cell averages defined in (2.67),

$$\mathcal{I}_1 = \bar{R}^{(1)}_{i,j} + \bar{R}^{(2)}_{i,j+1} + \bar{R}^{(3)}_{i+1,j+1} + \bar{R}^{(4)}_{i+1,j}$$

where the polynomials $P_{i+l,j+k}(x,y)$ appearing in (2.67) are given by (2.71) and the weights $w^{l,k}_{i,j}$ are given by (2.72).

The integrals in $\mathcal{I}_2$ are replaced by the quadrature (2.62) and (2.63).

$$\begin{aligned}&\int_{t^n}^{t^{n+1}} f(u(x_i,y_j,z))dz = \\ &\quad = \frac{\Delta t}{6}\left[f\left(u^n_{i,j}\right) + 4f\left(u^{n+1/2}_{i,j}\right) + f\left(u^{n+1}_{i,j}\right)\right] + O\left((\Delta t)^5\right),\end{aligned}$$

and

$$\int_{x_i}^{x_{i+1}} f(x)dx = \frac{h}{24}[-f(x_{i+2}) + 13f(x_{i+1}) + 13f(x_i) - f(x_{i-1})] + O(h^5).$$

The time quadrature required the prediction of the mid-values, which can be obtained with the RK scheme, (2.26), and their NCE (2.28). This ODE solver requires on the RHS the values of the derivatives of the fluxes given by (2.80), which are evaluated at the integer grid-points (and therefore utilizes the point-values recovered by (2.68)).

2.3.7 Systems of Equations

There are not that many modifications required in order to solve systems of equations instead of solving scalar equations. Basically, one has to extend the algorithm to systems using a straightforward component-wise approach.

The only delicate point is the computation of the smoothness indicators. A component-wise evolution of the smoothness indicators where each component may relay on a different stencil has some disadvantages, as already pointed out in the one dimensional case. The simplest and most robust way to compute the smoothness indicators is to apply *global smoothness indicators*: all components have the same indicator, which is computed as an average of the smoothness indicators of each component,

$$S_{i,j}^{l,k} = \frac{1}{d}\sum_{m=1}^{d}\left\{\int_{I_{i,j}}\left(|\partial_x P_{i+l,j+k}^m|^2 + |\partial_y P_{i+l,j+k}^m|^2 + \right.\right. \tag{2.81}$$
$$\left.\left. h^2|\partial_{xx}^2 P_{i+l,j+k}^m|^2 + +h^2|\partial_{yy}^2 P_{i+l,j+k}^m|^2\right) dxdy\right\}\left(\|\bar{u}^{(m)}\|_2 + \epsilon\right)^{-1}.$$

Here $P_{i,j}^m$ denotes the m-th component of the vector valued interpolation polynomial, centered on the cell $I_{i,j}$, and:

$$\|\bar{u}^{(m)}\|_2^2 = \sum_{i,j}|\bar{u}_{i,j}^{(m)}|^2\, h^2,$$

where (m) denotes the m-th component of the vector $\bar{u}_{i,j}$. Therefore, the Global Smoothness Indicator is an average of all componentwise Smoothness Indicators, each of which is normalized with respect to the norm of the corresponding field.

2.3.8 Numerical Examples

In order to show the non-oscillatory and high resolution properties of the fourth order CWENO scheme, we consider three test problems. The first is a convergence test performed on the linear equation

$$u_t + u_x + u_y = 0$$

with initial condition $u(x,y,0) = \sin^2(2\pi x)\sin^2(2\pi y)$, with periodic boundary conditions. The convergence results of this test are reported in Table 2.3.

Linear advection, Nonlinear weights

N	L^1 error	L^1 order	L^∞ error	L^∞ order
10	8.763E-03	-	2.464E-02	-
20	5.092E-04	4.10	1.632E-03	3.92
40	3.001E-05	4.08	8.747E-05	4.22
80	1.828E-06	4.04	4.836E-06	4.18
160	1.135E-07	4.01	2.802E-07	4.11

Table 2.3 $T = 1$, $\lambda = 0.45$, $u_0(x) = \sin^2(\pi x)\sin^2(\pi y)$

It is evident that the scheme shows an accuracy of fourth order, as expected.

The second test consists in the solution of the 2D Burgers equation, with flux $f(u) = g(u) = -u^2/2$. The initial data, taken from [25], consists in a two dimensional Riemann problem, and is given by

0.8	0.5
-1	-0.2

The configuration is centered at $(1/2, 1/2)$, and the computational region is $[0,1]\times[0,1]$. The boundary conditions are $\partial u/\partial n = 0$. Such conditions are perfectly justified until the signal reaches the boundary. The flux is Burgers', $f(u) = g(u) = -u^2/2$. The number of grid points in each direction is $N = 80$, and $\lambda = .25$.

The solution at $T = .5$ is shown in Figure 2.14. The figure shows the absence of spurious oscillations in a problem involving shock interaction. All discontinuities are very sharp.

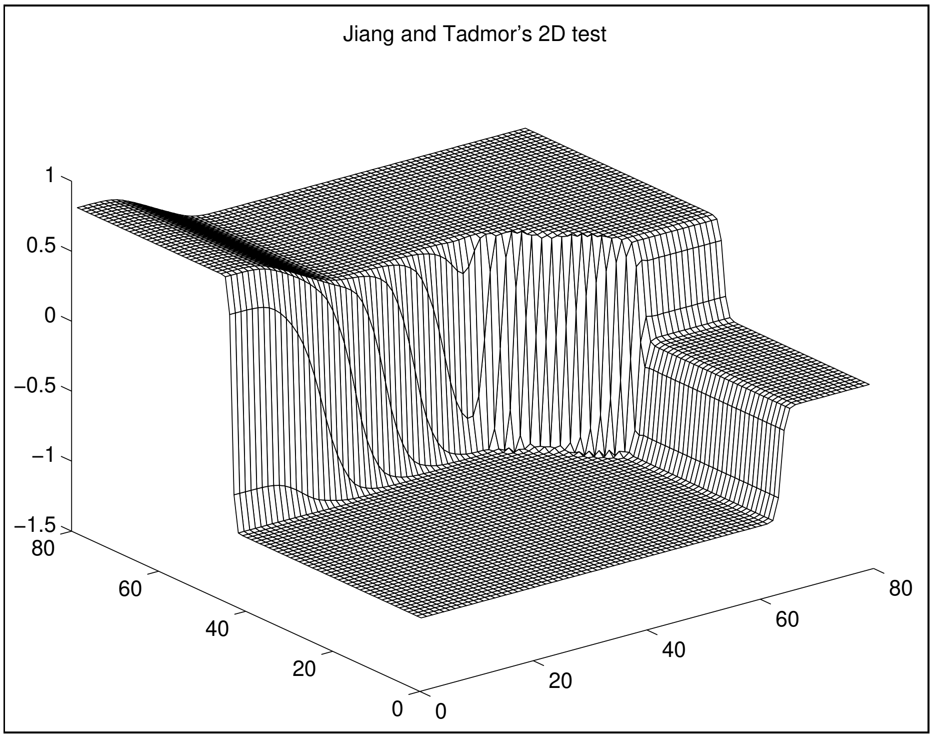

Figure 2.14 Burgers equation. Solution at $T = .5$. $N = 80$, $\lambda = .25$

The third test is described below.

Two-Dimensional Gas Dynamics Equations

We consider the system of equations for gas dynamics in 2D

$$U_t + F(U)_x + G(U)_y = 0,$$

where

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

Here, ρ is the density, u and v are the two components of the velocity, $E = \rho e + \frac{1}{2}\rho(u^2 + v^2)$ is the total energy per unit volume, and e is the internal energy of the gas. The system is closed by defining the pressure p through the equation of state. For a polytropic gas $p = \rho e(\gamma - 1)$, where the constant γ is the ratio of specific heats. In all tests considered, $\gamma = 1.4$.

$\begin{pmatrix} \rho \\ u \\ v \\ p \end{pmatrix} = \begin{pmatrix} 2 \\ -0.75 \\ 0.5 \\ 1 \end{pmatrix}$	$\begin{pmatrix} \rho \\ u \\ v \\ p \end{pmatrix} = \begin{pmatrix} 1 \\ -0.75 \\ -0.5 \\ 1 \end{pmatrix}$
$\begin{pmatrix} \rho \\ u \\ v \\ p \end{pmatrix} = \begin{pmatrix} 1 \\ 0.75 \\ 0.5 \\ 1 \end{pmatrix}$	$\begin{pmatrix} \rho \\ u \\ v \\ p \end{pmatrix} = \begin{pmatrix} 3 \\ 0.75 \\ -0.5 \\ 1 \end{pmatrix}$

Table 2.4 Initial condition for Configuration 5 (Figure 2.3.8

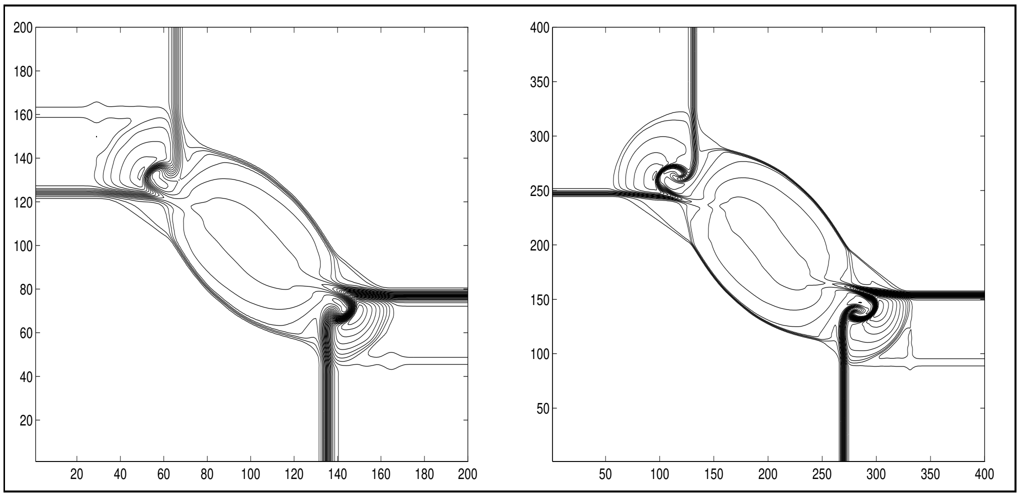

Figure 2.15 Two dimensional Riemann Problem. Solution at time $T = 0.23$ for the initial data reported in Table 2.4. Comparison with two different grids. On the left, the grid is 200×200, on the right, the grid is 400×400

The test problems shown below are two dimensional Riemann problems, taken from [36]. The initial is given by a piecewise constant data in the four quadrants of $\mathbb{R}^2$. The initial condition for the first one is given in Table 2.4 (denoted as Configuration 5 in [36]). These initial data result in four interacting contact discontinuities.

The results of the computation are shown in Figure 2.3.8, at time T=0.23. On the left, we show the density obtained with a 200×200 grid, while on the right we show the density on a 400×400 grid. The mesh ration one half the one chosen in [36], due to our more restrictive CFL, namely $\lambda = 0.5 * 0.2494$.

We first note that there is a very strong increase in resolution as the cell dimensions are halved due to the high-order accuracy of the scheme. When we compare the results obtained on the fine grid with the corresponding ones in Figure 5 of [36], we find that the two pictures are almost identical. Although the positive schemes used by Lax and Liu in [36] are only second order accurate, we believe that our results are quite striking. In fact, while the positive scheme makes use of the Jacobian and the matrix of eigenvectors of the system of gas dynamics, our scheme requires only the definition of the fluxes. Still, the physics of the problem, apparently, is perfectly caught.

$\begin{pmatrix} \rho \\ u \\ v \\ p \end{pmatrix} = \begin{pmatrix} 1.0222 \\ -0.6179 \\ 0.1 \\ 1 \end{pmatrix}$	$\begin{pmatrix} \rho \\ u \\ v \\ p \end{pmatrix} = \begin{pmatrix} 0.5313 \\ 0.1 \\ 0.1 \\ 0.4 \end{pmatrix}$
$\begin{pmatrix} \rho \\ u \\ v \\ p \end{pmatrix} = \begin{pmatrix} 0.8 \\ 0.1 \\ 0.1 \\ 1 \end{pmatrix}$	$\begin{pmatrix} \rho \\ u \\ v \\ p \end{pmatrix} = \begin{pmatrix} 1 \\ 0.1 \\ 0.8276 \\ 1 \end{pmatrix}$

Table 2.5 Initial condition for Configuration 16 (Figure 2.16)

We end our discussion showing the results obtained for Configuration 16 of [36]. The initial condition can be found in Table. 2.3.8. The resulting solution is composed of two contact discontinuities, a rarefaction and a shock wave. We show the results for the density at $T = 0.2$ on a 400×400 grid in Figure 2.16. The CFL number is $\lambda = 0.5 * 0.2494$. These results should be compared with the corresponding ones in [36], Figure 16. We note that the shock is sharp, and the resolution of the two contact discontinuities is also good. Moreover, there are no spurious oscillations even though the wave pattern is complex.

2.3.9 Improvements

The same improvements considered in one dimension can be extended in the two dimensional case. We discuss here only the better use of Runge-Kutta schemes, and we do not consider the problem of the negative weights.

System (2.55) can be written in the form (2.48)

$$\frac{\partial u}{\partial t} = F(u)$$

where $F(u) = -\partial f/\partial x - \partial g/\partial y$. By applying a ν-stage explicit Runge-Kutta scheme pointwise at the different stages, and taking the average of the solution at time t_{n+1}, one obtains

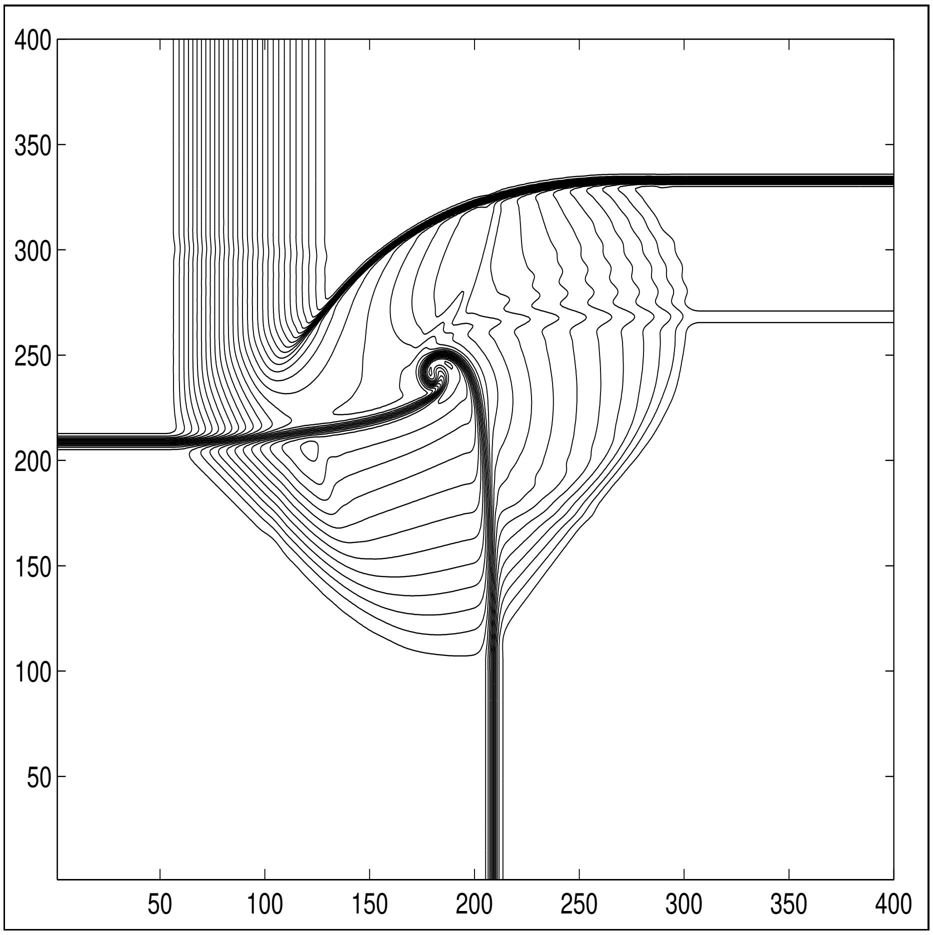

Figure 2.16 Two dimensional Riemann problem. 400×400 grid. Solution at time $T = 0.2$ for the initial data reported in Figure 2.3.8

$$u^{(k)}(x,y) = u^n(x,y) + \Delta t \sum_{l=1}^{k-1} a_{kl} F(u^{(l)}(x,y))$$

$$\begin{aligned} \bar{u}^{n+1}(x+h/2,y+h/2) &= \bar{u}^n(x+h/2,y+h/2) \\ &-\lambda \sum_{k=1}^{\nu} b_k \Big[\int_y^{y+h} (f(u^{(k)}(x+h,\tilde{y})) - f(u^{(k)}(x,\tilde{y})))\, d\tilde{y} \\ &- \int_x^{x+h} (g(u^{(k)}(\tilde{x},y+h)) - g(u^{(k)}(\tilde{x},y)))\, d\tilde{x} \Big] \end{aligned} \tag{2.82}$$

The above relation can be discretized on a grid, by performing a WENO reconstruction of the function. Using the WENO reconstruction, the staggered cell average can be computed. About the computation of the integral of the flux appearing in (2.82), one can use either the quadrature formula (2.63) or a Gaussian quadrature formula with nodes inside the cell. The latter approach appears more robust, but is more expensive, since it requires more function evaluations.

2.4 Treatment of the source

Several models in mathematical physics are described by quasilinear hyperbolic systems with source term. The source may be of relaxation type, or it may represent a diffusion term. In this case it is more appropriate to talk about balance laws, rather than conservation laws.

Hyperbolic systems with relaxation appear in discrete velocity models in kinetic theory [18], gas with vibrational degrees of freedom [68], gas in Extended Thermodynamics [49], hydrodynamical models for semiconductors [2, 3], shallow water equations [62], and in several other physical models.

In this section we consider systems of balance laws of the form

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = -\frac{1}{\epsilon} g(u) \tag{2.83}$$

The parameter ϵ represents the relaxation time. If it is very small than we say the the relaxation term is *stiff*.

Several schemes have been proposed for the treatment of systems with source term. The most common technique is based on fractional step methods.

In the simplest version, one step for (2.83) is obtained by solving, for a time step Δt, the sequence of the relaxation and convection equations. Given the numerical approximation of the solution, u^n, the new approximation u^{n+1} is obtained as

$$\frac{\partial \tilde{u}}{\partial t} = -\frac{1}{\epsilon} g(\tilde{u}), \quad \tilde{u}(x,0) = u^n(x) \tag{2.84}$$

$$\frac{\partial \hat{u}}{\partial t} + \frac{\partial f(\hat{u})}{\partial x} = 0, \quad \hat{u}(x,0) = \tilde{u}(x,\Delta t) \tag{2.85}$$

At the end of the two steps, one sets $u^{n+1}(x) = \hat{u}(x,\Delta t)$. Such splitting provides a first order (in time) approximation of the solution. A second order scheme can be obtained by using the so called *Strang splitting*: it is obtained by solving (2.84) for a time step $\Delta t/2$, then equation (2.85) for a time step Δt, and finally equation (2.84) for a time step $\Delta t/2$ again. Of course, each step (2.84) and (2.85) has to be solved at least with second order accuracy.

This approach can be generalized to obtain high order methods.

There are several cases, however, when the splitting approach is not very effective. When the relaxation term is stiff, for example, Strang splitting loses second order accuracy, and more sophisticated splitting techniques are necessary to obtain a second order scheme.

In [27], for example, a second order method for systems with stiff source is considered. In [13], a second order scheme for hyperbolic systems with relaxation is derived, which maintain second order accuracy both in the stiff and non stiff limit. These methods have been developed in the context of upwind schemes.

The same approach used in [13] can not be straightforwardly extended to central schemes.

A different approach has been proposed by Bereux and Sainsaulieu in [10], where a system with a possibly stiff relaxation term is considered. Their scheme maintains second order accuracy for stiff source.

A natural way to treat source term in central scheme is to include the source in the integration over the cell is space-time. The schemes we consider here are of this form.

Integrating Eq.(2.83) over the space-time (see Fig. 2.3) one has

$$
\begin{aligned}
\int_{x_j}^{x_{j+1}} u(x,t_{n+1}) &= \int_{x_j}^{x_{j+1}} u(x,t_n) - \int_{t_n}^{t_{n+1}} \left(f(u(x_{j+1},t)) - f(u(x_j,t))\right) dt \quad (2.86)\\
&+ \int_{t_n}^{t_{n+1}} \int_{x_j}^{x_{j+1}} g(x,t)\, dx\, dt \quad (2.87)
\end{aligned}
$$

Numerical schemes are obtained by a suitable discretization of the integrals.

Here we only consider second order schemes. We shall discuss later how to derive higher order schemes.

For a second order scheme we use a piecewise linear reconstruction in each cell, as in the Nessyahu-Tadmor scheme, and midpoint rule for the computation of the flux integral.

Different schemes are obtained, according to the discretization of the integral of the source term. If the source is not stiff, then a fully explicit time discretization can be used, resulting in the following scheme

$$
\begin{aligned}
u_{j+1/2}^{n+1} &= \bar{u}_{j+1/2}^{n} + \frac{\Delta t}{\Delta x}\left(f(u_j^{n+1/2}) - f(u_{j+1}^{n+1/2})\right)\\
&+ \frac{\Delta t}{2}\left(g(u_j^{n+1/2}) + g(u_{j+1}^{n+1/2}\right) \quad (2.88)\\
& \quad (2.89)
\end{aligned}
$$

where

$$
\bar{u}_{j+1/2}^{n} = \frac{1}{2}(u_j^n + u_{j+1}^n) + \frac{1}{8}(u_j' - u_{j+1}') \quad (2.90)
$$

and the predictor values $u_j^{n+1/2}$ are computed by

$$
u_j^{n+1/2} = u_j^n - \frac{\lambda}{2} f_j' + \frac{\Delta t}{2} g(\bar{u}_j^n) \quad (2.91)
$$

Note that this scheme is fully explicit, therefore it is subject to stability restriction due to both the flux and the source term. If the stability restriction of the source term is more severe than the one due to the flux, then not only efficiency, but also accuracy of the calculation will be affected. In this case, in fact, one has to use a Courant number much smaller than the one allowed by the CFL restriction. It is well known that in this case the numerical dissipation will be larger than necessary, and the accuracy of the scheme will be poor. This problem can be partially circumvented, for moderately stiff source, by the use of semidiscrete schemes. (For a discussion of this issue see, for example, [32]). When the stiffness increases it is better to treat the source implicitly. This can be done at the stage of the predictor, as

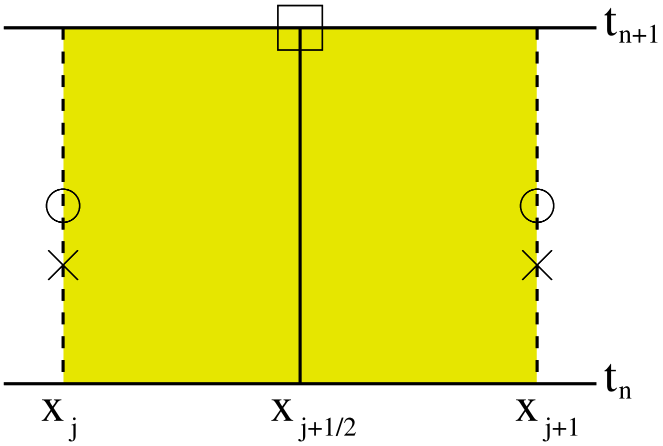

Figure 2.17 Nodes in space time for the second order Uniform Central Scheme

$$u_j^{n+1/2} = u_j^n - \frac{\lambda}{2} f_j' + \frac{\Delta t}{2} g(u_j^{n+1/2}) \tag{2.92}$$

The time discretization used for the source is the midpoint implicit scheme, which is a Gauss-collocation scheme with one level. Such scheme is A-stable, but not L-stable, and therefore it is not suitable for very stiff source (see [21]). A numerical scheme which is stable and accurate even for very stiff source has been proposed in [46]. It is obtained by using two predictor stages, one for the flux, and one which is needed for the source. The scheme can be written in the form

$$\begin{aligned}
u_j^{n+1/2} &= u_j^n - \frac{\lambda}{2} f_j' + \frac{\Delta t}{2} g(u_j^{n+1/2}) \\
u_j^{n+1/3} &= u_j^n - \frac{\lambda}{3} f_j' + \frac{\Delta t}{3} g(u_j^{n+1/3}) \\
u_{j+1/2}^{n+1} &= \bar{u}_{j+1/2}^n - \lambda(f(u_{j+1}^{n+1/2}) - f(u_j^{n+1/2})) \\
&\quad + \frac{\Delta t}{8}(3g(u_j^{n+1/3}) + 3g(u_{j+1}^{n+1/3}) + 2g(u_{j+1/2}^{n+1}))
\end{aligned} \tag{2.93}$$

where $\bar{u}_{j+1/2}^n$ is computed by 2.90. We shall call the above scheme Uniformly accurate Central Scheme of order 2 (UCS2).

Such scheme has the following properties. When applied to hyperbolic systems with relaxation, it is second order accurate in space and time both in the non stiff case (i.e. $\epsilon = 1$) and in the stiff limit (i.e. $\epsilon = 0$).

A small degradation of accuracy is observed for intermediate values of the relaxation parameter ϵ.

The time discretization used in the above scheme is a particular case of Runge-Kutta Implicit-Explicit (IMEX) scheme. Such schemes are particularly mportant when one ha to solve systems that contain the sum of e non stiff (possibly expensive to compute) and a stiff term. These systems may be convection-diffusion equations, or hyperbolic systems with stiff relaxation. In these cases it is highly desirable to use a scheme which is explicit in the non stiff term, and implicit in the stiff term.

Runge-Kutta IMEX schemes have been studied in [9] and in [53].

Let us consider a generic system

$$y' = f(t,y) + \frac{1}{\epsilon} g(t,y) \tag{2.94}$$

which may represent a system of ordinary differential equations or a discretization of a system of partial differential equations (method of lines).

An Implicit-Explicit Runge-Kutta scheme for system (2.94) is of the form

$$\begin{aligned} Y_i &= y_0 + h \sum_{j=1}^{i-1} \tilde{a}_{ij} f(t_0 + \tilde{c}_j h, Y_j) + h \sum_{j=1}^{\nu} a_{ij} \frac{1}{\varepsilon} g(t_0 + c_j h, Y_j), & (2.95) \\ y_1 &= y_0 + h \sum_{i=1}^{\nu} \tilde{w}_i f(t_0 + \tilde{c}_i h, Y_i) + h \sum_{i=1}^{\nu} w_i \frac{1}{\varepsilon} g(t_0 + c_i h, Y_i). & (2.96) \end{aligned}$$

The matrices $\tilde{A} = (\tilde{a}_{ij})$, $\tilde{a}_{ij} = 0$ for $j \geq i$ and $A = (a_{ij})$ are $\nu \times \nu$ matrices such that the resulting scheme is explicit in f, and implicit in g. An IMEX Runge-Kutta scheme is characterized by these two matrices and the coefficient vectors $\tilde{c} = (\tilde{c}_1, \ldots, \tilde{c}_\nu)^T$, $\tilde{w} = (\tilde{w}_1, \ldots, \tilde{w}_\nu)^T$, $c = (c_1, \ldots, c_\nu)^T$, $w = (w_1, \ldots, w_\nu)^T$. They can be represented by a double *tableau* in the usual Butcher notation,

$$\begin{array}{c|c} \tilde{c} & \tilde{A} \\ \hline & \tilde{w}^T \end{array}, \qquad \begin{array}{c|c} c & A \\ \hline & w^T \end{array}.$$

A sufficient condition to guarantee that f is always evaluated explicitly is that the scheme for g is diagonally implicit i.e. $a_{ij} = 0$, for $j > i$, and that the first column of a is zero.

Stability analysis of the scheme is studied in [46], and the analysis of other IMEX RK schemes is performed in [53]. In particular, the concept of A-stable scheme for a single complex linear equation is extended to IMEX schemes. Here we report the result found in [46].

By applying the generic IMEX scheme to the linear equation (2.94) with $f(y) = \lambda_1 y$, $g(y) = \lambda_2 y$, and $y_0 = 1$, one obtains the function of absolute stability

$$y_1 = \mathcal{R}(z_1,z_2) \equiv 1 + (z_1 \tilde{w}^T + z_2 w^T)(I - z_1 \tilde{A} - z_2 A)^{-1} e, \tag{2.97}$$

where $z_1 = \lambda_1 h$, $z_2 = \lambda_2 h$, $e = (1, \ldots, 1)^T \in \Re^\nu$. The function $\mathcal{R}(z_1,z_2)$ is the *function of absolute stability.*

The *region of absolute stability* S_A associated to scheme (2.95, 2.96) is defined as

$$S_A = \{(z_1,z_2) \in \mathbb{C}^2 : |\mathcal{R}(z_1,z_2)| \leq 1\}.$$

It is evident that the region does not contain the set $\mathbb{C}^- \times \mathbb{C}^-$. Our goal is to show that there exist two regions of the complex plane, $S_1 \subset \mathbb{C}$, $S_2 \subset \mathbb{C}$, with the following properties:

$$S_A \supset S_1 \times S_2, \tag{2.98}$$
$$S_2 \supset \mathbb{C}^- \equiv \{z \in \mathbb{C} : \Re(z) \leq 0\}, \tag{2.99}$$

and to compute them. In particular, one is interested in the largest set S_1 for which $S_2 \supset \mathbb{C}^-$. Such region is defined by

$$S_1 = \{z_1 \in \mathbb{C} : \sup_{z_2 \in \mathbb{C}^-} |\mathcal{R}(z_1,z_2)| \leq 1\}.$$

The region S_1 for scheme UCS2 is reported in Figure (2.18).

We remark that this stability analysis is valid only for the scalar equation. The extension to systems of equations is presently under investigation.

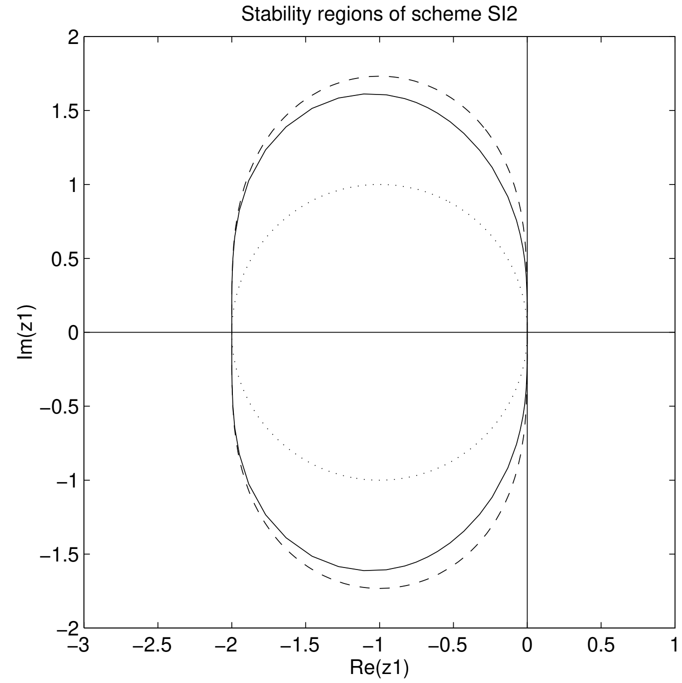

Figure 2.18 Stability region S_1 in the explicit parameter for scheme UCS2 (continuous line). The dotted and the dashed lines are the stability region of explicit Euler and explicit second order Runge-Kutta respectively.

To show the effectiveness of the scheme, we apply it to the monoatomic gas in Extended Thermodynamics.

Monatomic gas in Extended Thermodynamics As an application of the scheme UCS2, we consider the a Riemann problem for a gas in Extended Thermodynamics. In usual gas dynamics, a gas is described by the conservation laws for mass, momentum, end energy. Pressure, stress tensor, and heat flux are related to the conservative quantities by constitutive relations (i.e. the equation of state for the pressure and the Navier-Stokes-Fourier relations that express stress and heat flux in terms of the gradient of velocity and temperature. In Extended Thermodynamics stress tensor and heat flux are independent thermodynamical variables, for which additional balance laws can be written [49]. In one dimension the gas is described by a system of five balance equations, the usual three conservation equations for mass, momentum, and energy, and two additional equations for the balance of stress σ (just one component in one dimension) and heat flux q. The system is of the form

$$u_t + f(u)_x = -\frac{1}{\epsilon} g(u)$$

with

$$u = \begin{pmatrix} \rho \\ \rho v \\ \frac{1}{2}\rho v^2 + \frac{3}{2}p \\ \frac{2}{3}\rho v^2 + \sigma \\ \rho v^3 + 5vp + 2\sigma v + 2q \end{pmatrix}, g = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \rho\sigma \\ \frac{2}{3}\rho(2q + 3v\sigma) \end{pmatrix}$$

$$f = \begin{pmatrix} \rho v \\ \rho v^2 + p + \sigma \\ \frac{1}{2}\rho v^3 + \frac{5}{2}vp + \sigma v + q \\ \frac{2}{3}\rho v^3 + \frac{4}{3}vp + \frac{7}{3}v\sigma + \frac{8}{15}q \\ \rho v^4 + 5\frac{p^2}{\rho} + 7\frac{\sigma p}{\rho} + \frac{32}{5}qv + v^2(8p + 5\sigma) \end{pmatrix}$$

As $\epsilon \to 0 \;\Rightarrow\; \sigma \to 0, q \to 0$ and the equations reduce to the Euler equations for monoatomic gas.

Such system is hyperbolic in a suitable region of the field vector u, which contains the equilibrium manifold $\sigma = 0$, $q = 0$.

No simple expression for the eigenvalues and eigenvector is known for such system, therefore it would be expensive to use characteristic based methods.

We consider the following Riemann problem.

$$\rho = 1, \quad u = 0, \quad p = \frac{5}{3}, \quad \sigma = 0, \quad q = 0 \quad \text{for } x < 0.5,$$
$$\rho = \frac{1}{8}, \quad u = 0, \quad p = \frac{1}{6}, \quad \sigma = 0, \quad q = 0 \quad \text{for } x > 0.5.$$

The numerical solution to the Riemann problem is shown in figure (2.19). By comparison, the solution to the Euler equations corresponding to the same initial conditions is reported. The two solutions are very close for density and velocity, because the value of the relaxation parameter is very small.

Stress tensor and heat flux are mainly concentrated at the shock. Two schemes are compared: UCS2 and a scheme based on second order upwind + Strang splitting. It is evident that UCS2 gives a better resolution than the splitting scheme, because of the loss of accuracy of the latter for small values of the relaxation parameter.

The comparison is shown between the numerical solution of the Euler equations and of the gas in Extended Thermodynamics. It is

Further developments

The scheme UCS2 shown above is not optimal, since it makes uses of three implicit stages in order to obtain second order accuracy. More efficient time discretization can be obtained. A recently developed central scheme by Pareschi guarantees uniform accuracy in the stiff and non stiff limit with two explicit and two implicit stages [51].

There are several topics we did not have space to deal with in this chapter. We shall briefly mention them here.

We did not say anything about boundary conditions. In all calculations we used trivial boundary conditions, either periodic, or flat. Such boundary conditions can be easily treated by using enough ghost cells out of the computational domain. The cells will be then filled with a periodic replica of the computed solution (for periodic B.C.) or with constant values (for zero flux B.C.) The latter are correct until the signal reaches the boundary.

A case which is particularly important in the applications is given by reflecting boundary conditions for gas dynamics. This case is treated in the usual way, i.e. pressure and density are symmetric with respect to the boundary, and velocity is antisymmetric. Such boundary conditions are assigned both at even and odd time steps. At odd time step, the value of the field variables on the boundary cell can be computed if enough ghost cells are filled.

The treatment of more complex boundary conditions goes beyond the scope of the book.

About the derivation of schemes for hyperbolic systems with stiff source, we remark that semidiscrete central schemes can can be used as building blocks for second order IMEX time discretization of hyperbolic systems with source term. If one is interested in higher order schemes, however, this approach can not be straightforwardly used. Higher order semidiscrete central scheme based on WENO reconstruction are basically equivalent to WENO finite volume schemes developed and used by Shu [58]. However, the integration of a source term in space-time may pose some problems for stiff problems.

Higher order IMEX time discretization can be used by using *finite difference space discretization*, rather than *finite volume*. Finite difference schemes have the same structure of finite volume. They have a conservative form, and the basic unknown is the *pointwise value* rather that the cell average of the unknown field $u(x,t)$. They have the advantage of being more efficient that their finite volume counterpart. However, they have the disadvantage of requiring a uniform (or smoothly varying) grid, and therefore they can not be extended to unstructured grids. An extensive treatment of high order finite difference schemes is given in [60, 26, 58].

The development of high order finite volume schemes for systems with stiff source is still an open problem.

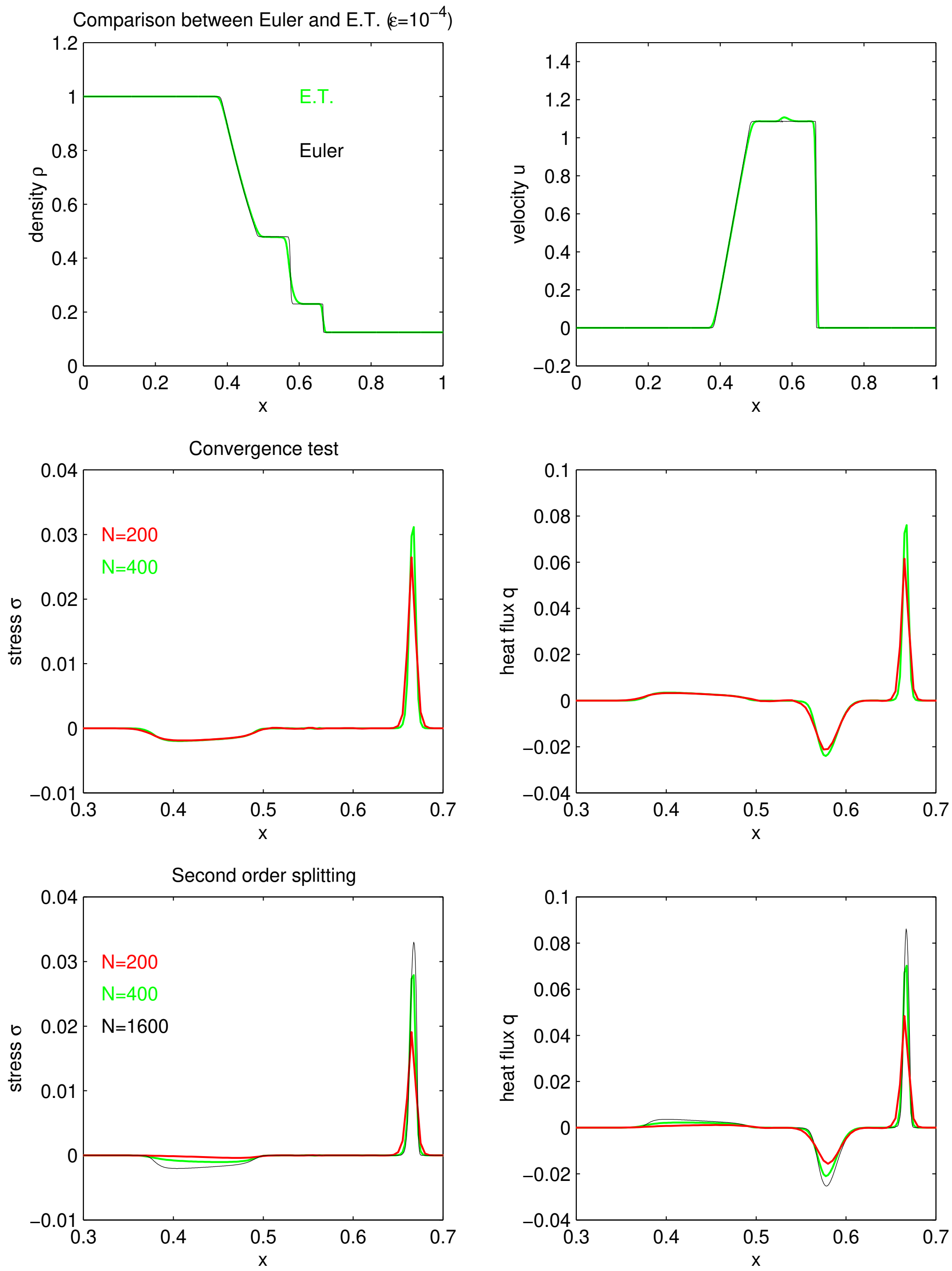

Figure 2.19 Riemann problem in Extended Thermodynamics.

Figure 2.20 Solutions of extended thermodynamics model with $\varepsilon = 10^{-4}$ for initial data 2.100 and comparison with the solution of the Euler equations of gas dynamics. Mesh ratio $\Delta t = \Delta x/9$. Top pictures: density (left) and velocity (right) computed with 200 grid points with UCS2 for Extended Thermodynamics (gray line) and NT scheme for Euler equation (black line). Bottom pictures: Shear stress (left) and heat flux (right) for Extended Thermodynamics model with $\varepsilon = 10^{-4}$. Note that the x-scale has been magnified. Mesh ratio $\Delta t = \Delta x/9$. Middle pictures: UCS2 scheme with 200 (gray line) and 400 (black line). Bottom pictures: upwind+splitting scheme with 200 (gray line) and 400 (black line). The thin line in the bottom pictures is the reference solution obtained by UCS2 with 1600 grid points.

Bibliography

[1] Anile, Angelo Marcello; Junk, Michael; Romano, Vittorio; Russo, Giovanni, *Cross-validation of numerical schemes for extended hydrodynamical models of semiconductors*, Math. Models Methods Appl. Sci., 10(6), 833-861 (2000).

[2] A. M. Anile and S. Pennisi, *Thermodynamic derivation of the hydrodynamical model for charge transport in semiconductors*, Phys. Rev B, **46**, 13186–13193 (1992).

[3] A. M. Anile and O. Muscato, *Improved hydrodynamical model for carrier transport in semiconductors* Phys. Rev B, **51**, 16728-16740 (1995).

[4] A. M. Anile, N. Nikiforakis and R. M. Pidatella, *Assessment of a high resolution centered scheme for the solution of hydrodynamical semiconductor equations*, SIAM Journal of Scientific Computing 22 (2000) 1533-1548.

[5] A. M. Anile, V. Romano and G. Russo, *Extended hydrodynamical model of carrier transport in semiconductors*, SIAM Journal Applied Mathematics 61 (2000) 74-101.

[6] Arminjon, Paul; Viallon, Marie-Claude, *Gnralisation du schma de Nessyahu-Tadmor pour une quation hyperbolique deux dimensions d'espace. (French) [Generalization of the Nessyahu-Tadmor scheme for hyperbolic equations in two space dimensions]* C. R. Acad. Sci. Paris Sr. I Math. 320 (1995), no. 1, 85–88.

[7] Arminjon, P.; Viallon, M.-C.; Madrane, A., *A finite volume extension of the Lax-Friedrichs and Nessyahu-Tadmor schemes for conservation laws on unstructured grids.* Int. J. Comput. Fluid Dyn. 9 (1997), no. 1, 1–22.

[8] Arminjon, Paul; Viallon, Marie-Claude, *Convergence of a finite volume extension of the Nessyahu-Tadmor scheme on unstructured grids for a two-dimensional linear hyperbolic equation*, SIAM J. Numer. Anal. 36 (1999), no. 3, 738–771

[9] U. Ascher, S. Ruuth, and R. J. Spiteri, *Implicit-explicit Runge-Kutta methods for time dependent Partial Differential Equations*, Appl. Numer. Math. **25**, 151–167 (1997).

[10] F. Bereux and L. Sainsaulieu, *A Roe-type Riemann solver for hyperbolic systems with relaxation based on time-dependent wave decomposition*, Numer. Math. **77** 143–185 (1997).

[11] 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.

[12] F. Bianco, G. Puppo, G. Russo, *High Order Central Schemes for Hyperbolic Systems of Conservation Laws*, HYP-98 Seventh International Conference on Hyperbolic Problems, Theory, Numerics, Applications, ETH Zuerich, Switzerland, February 9-13, 1998.

[13] R. E. Caflisch, S. Jin and G. Russo, *Uniformly accurate schemes for hyperbolic systems with relaxation*, SIAM J. Numer. Anal., **34**, 246–281 (1997).

[14] B. Cockburn, C. Johnson, Shu C.-W., E. Tadmor, *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations*, Lecture Notes in Mathematics (editor: A. Quarteroni), Springer, Berlin, 1998.

[15] B. Engquist and S.Osher, *One sided difference approximations for nonlinear conservation laws*, Math. Comp. **36** (1981), 321–351.

[16] B. Engquist and O. Runborg, *Multiphase computations in geometrical optics*, Journal of Computational and Applied Mathematics 74 (1996) 175-192.

[17] Engquist, Bjorn; Sjgreen, Bjrn, *The convergence rate of finite difference schemes in the presence of shocks.* SIAM J. Numer. Anal. 35 (1998), no. 6, 2464–2485.

[18] R. Gatignol, *Théorie cinétique des gaz à répartition discrète de vitesses*, Lectures Notes in Physics, vol. 36, Springer-Verlag, Berlin–New York (1975).

[19] Godlewski, Edwige; Raviart, Pierre-Arnaud, *Numerical approximation of hyperbolic systems of conservation laws.* Applied Mathematical Sciences, 118. Springer-Verlag, New York, 1996.

[20] B. Haasdonk, B. Kröner & D. Rohde, *Convergence of a staggered Lax-Friedrichs scheme for nonlinear conservation laws on unstructured two-dimensional grids*, Numerische Mathematik, 2000.

[21] E. Hairer and G. Wanner, *Solving ordinary differential equations, Vol.2 Stiff and differential-algebraic problems*, Springer-Verlag, New York (1987).

[22] Harabetian, Eduard; Pego, Robert, Nonconservative hybrid shock capturing schemes. J. Comput. Phys. **105** (1993), no. 1, 1–13.

[23] (UNO limiter) Harten A., Engquist B., Osher S., Chakravarthy S., *Uniformly High Order Accurate Essentially Non-oscillatory Schemes III*, J. Comput. Phys., **71**, (1987), pp.231–303.

[24] Jiang G.-S., Levy D., Lin C.-T., Osher S., Tadmor E., *High-Resolution Non-Oscillatory Central Schemes with Non-Staggered Grids for Hyperbolic Conservation Laws*, SIAM J. Numer. Anal. 35 (1998), no. 6, 2147–2168.

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

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

[27] S. Jin, *Runge-Kutta methods for hyperbolic systems with stiff relaxation terms*, J. Comp. Phys. **122**, 51–67 (1995).

[28] T. Katsaounis & D. Levy, *A modified structured central scheme for 2D hyperbolic conservation laws*, Applied Mathematics Letters **12** 89-96 (1999).

[29] R.Kupferman, *A numerical study of the axisymmetric Couette-Taylor problem using a fast high-resolution second-order central scheme*, SIAM Journal on Scientific Computing 20 (1998), 858-877

[30] R. Kupferman & E. Tadmor, *A fast high-resolution second-order central scheme for incompressible flows*, Proceedings of the National Academy of Sciences 94 (1997) 4848-4852.

[31] A. Kurganov and D. Levy, *A third-order semi-discrete central scheme for conservation laws and convection-diffusion equation*, SIAM Journal on Scientific Computing **22** (2000) 1461-1488.

[32] Kurganov A., Tadmor E., *New High-Resolution Central Schemes for Nonlinear Conservation Laws and Convection-Diffusion Equations*, J. Comput. Phys., **160** (2000), pp.214–282.

[33] A. Kurganov & E. Tadmor, *New High-Resolution Semi-Discrete Central Schemes for Hamilton-Jacobi Equations*, Journal of Computational Physics 160 (2000) 720-742.

[34] P.D.Lax, *Hyperbolic systems of conservation laws II*, Comm.Pure Appl. Math **10** (1957), 105–119.

[35] P.D.Lax, *Hyperbolic systems of conservation laws and the mathematical theory of shock waves*, (SIAM Philadelphia, 1973).

[36] Lax P.D., Liu X.D., *Solutions of Two-Dimensional Riemann Problems of Gas Dynamics by Positive Schemes*, SIAM J.Sci.Comp., **19**, No.2, (1998), pp.319–340.

[37] R. J. LeVeque, *Numerical Methods for Conservation Laws. Second edition*, Lectures in Mathematics ETH Zürich, Birkhäuser Verlag, Basel, 1992.

[38] 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.

[39] Levy D., Puppo G., Russo G., *A Third Order Central WENO Scheme for 2D Conservation Laws*, Appl. Num. Math., **33**, (2000), pp.407–414.

[40] Levy D., Puppo G., Russo G., *Compact Central WENO Schemes for Multidimensional Conservation Laws*, SIAM J. Sci. Comp., **22**, (2000), pp.656–672.

[41] Levy D., Puppo G., Russo G., *Central WENO Schemes for Multi-dimensional Hyperbolic Systems of Conservation Laws*, submitted.

[42] D. Levy & E. Tadmor, *Non-oscillatory central schemes for the incompressible 2-D Euler equations*, Mathematics Research Letters 4 (1997) 1-20.

[43] K.-A. Lie & S. Noelle, *Remarks on high-resolution non-oscillatory central schemes for multi-dimensional systems of conservation laws. Part I: An improved quadrature rule for the flux computation*, submitted.

[44] C.-T. Lin & E. Tadmor, *High-resolution non-oscillatory central scheme for Hamilton-Jacobi equations*, SIAM Journal on Scientific Computation 21 (2000) 2163-2186.

[45] LIU X.-D., TADMOR E., *Third Order Nonoscillatory Central Scheme for Hyperbolic Conservation Laws*, Numer. Math., **79**, (1998) 397-425.

[46] S. F. LIOTTA, V. ROMANO, G. RUSSO, *Central schemes for balance laws of relaxation type*, SIAM J. Numer. Anal.

[47] LIU X.-D., OSHER S., CHAN T., *Weighted Essentially Non-oscillatory Schemes*, JCP, **115**, (1994), pp.200-212.

[48] LIU X.-D., TADMOR E., *Third Order Nonoscillatory Central Scheme for Hyperbolic Conservation Laws*, Numer. Math., to appear.

[49] I. MÜLLER AND T. RUGGERI,, *Rational extended thermodynamics*, Springer-Verlag, Berlin, 1998.

[50] NESSYAHU H., TADMOR E., *Non-oscillatory Central Differencing for Hyperbolic Conservation Laws*, J. Comput. Phys., **87**, no. 2 (1990), pp.408–463.

[51] L.PARESCHI, *Central differencing based numerical schemes for hyperbolic conservation laws with relaxation terms*, Preprint (2000).

[52] L.PARESCHI, G.PUPPO, G.RUSSO,, *Improved central WENO schemes for conservation laws*, in preparation.

[53] L.PARESCHI, G.RUSSO, *Implicit-Explicit Runge-Kutta Schemes for Stiff Systems of Differential Equations*, In Recent Trends in Numerical Analysis, D.Trigiante Ed. Nova Science Publ., 2000, pp.269-288.

[54] ROE P. L., *Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes*, JCP, **43**, (1981), pp.357-372.

[55] V. ROMANO & G. RUSSO, *Numerical solution for hydrodynamical models of semiconductors*, Mathematical Models & Methods in Applied Sciences 10 (2000) 1099-1120.

[56] W.ROSENBAUM, M.RUMPF & S.NOELLE, *An adaptive staggered scheme for conservation laws*, "Hyperbolic Problems: Theory, Numerics, Applications", Proceedings of the 8th international conference held in Magdeburg, Feb. 2000.

[57] R. SANDERS AND W. WEISER, J. Comput. Phys., **10** 314 (1992).

[58] 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.

[59] J. SHI, C. HU AND C.-W. SHU, *A technique of treating negative weights in WENO schemes*, submitted to Journal of Computational Physics.

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

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

[62] STOKER, J. J. *Water waves: The mathematical theory with applications.* Pure and Applied Mathematics, Vol. IV. Interscience Publishers, Inc., New York; Interscience Publishers Ltd., London 1957.

[63] E. TADMOR, T. TANG, *Pointwise error estimates for scalar conservation laws with piecewise smooth solutions*, SIAM Journal on Numerical Analysis, 36 (1999), 1739-1756.

[64] E. TADMOR, T. TANG, *Pointwise convergence rate for nonlinear conservation laws*, Hyperbolic problems: theory, numerics, applications, Vol. II (Zrich, 1998), 925–934, Internat. Ser. Numer. Math., 130, Birkhuser, Basel, 1999.

[65] E. TADMOR AND C.C.WU, *Central scheme for the multidimensional MHD equations*, in preparation.

[66] TORO, ELEUTERIO F., *Riemann solvers and numerical methods for fluid dynamics. A practical introduction.* Second edition. Springer-Verlag, Berlin, 1999.

[67] M. TROVATO & P. FALSAPERLA, *Full nonlinear closure for a hydrodynamical model of transport in silicon*, Physical Review B-Condensed Matter 57 (1998) 4456-4471.

[68] W. G. VINCENTI AND C. H. KRUGER, JR., *Introduction to Physical Gas Dynamics*, Krieger Publishing Company, Malabar, Florida, 1982.

[69] G. B. WHITHAM, Linear and Nonlinear Waves, Wiley, 1974.

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