Behavior of Finite Difference Schemes for Hyperbolic Conservation Laws on Adaptive Redistributed Spatial Grids

Ch. Arvanitis * & A. I. Delis †

Abstract

In this work we consider finite difference schemes combined with dynamic spatial mesh redistribution. We study whether appropriate mesh redistribution is a satisfactory mechanism for increasing the resolution of numerical solutions for problems of scalar and systems of Conservation Laws (CL) in one space dimension, while being at the same time a stabilization mechanism for selecting the appropriate entropy solution. In order to increase the resolution around shocks areas and keep the computational cost low, our redistribution policy is to reconstruct spatially the numerical solution on a new mesh, where the solutions curvature is almost uniformly distributed, while the nodes cardinality is kept constant. We examine the stabilization properties of that redistribution process by adding it as a substep on the time evolution step of some classical schemes with known (unstable) characteristics. Testing the resulting method for several such schemes and on a large number of CL problems that have solutions with special characteristics (shocks, rarefaction areas, steady states) and compare the results with those produced by schemes with extra stabilization mechanisms (like slope/flux limiters, entropy corrections) we conclude that indeed the proposed redistribution adds such stabilization properties while at the same time increases the resolution.

Keywords: Finite Difference Methods, Adaptive Grid Redistribution, Hyperbolic Conservation Laws

AMS subject classifications: 35L65, 65M06

*Department of Applied Mathematics, University of Crete, Heraklion 71409, Crete, Greece, arvas@tem.uoc.gr

†Department of Sciences, Division of Mathematics, Technical University of Crete University Campus, Chania 73100, Crete, Greece, adelis@science.tuc.gr

1 Introduction

The application of finite difference schemes is a very popular choice for computing solutions of systems of Conservation Laws in the context: *find* $u : \mathbb{R}^d \times [0,T] \to \mathbb{R}^M$ *such that*

$$\begin{aligned} & u(\,\cdot\,,0) = u_0 \quad \textit{given} \\ & \partial_t u + \sum_{i=1}^{d} \partial_{x_i} F_i(u) = 0\,. \end{aligned} \tag{1.1}$$

Some classical schemes, of first or second order in space, when applied directly to this system will result to computational solutions with diffusive or oscillatory behavior, especially close to shocks. To overcome this difficulty several modifications of such schemes have been proposed in the literature where the necessary stability and viscosity mechanisms are imposed "by hand". Mesh adaptation is a main current stream towards to efficiently compute numerical solutions of complex systems by increasing the resolution of the essential solution. Several redistribution techniques have been introduced in the near past for solving the problem of proper mesh selection, starting with the Self-Adjusting method of Harten, Hyman [11] to the Moving Mesh methods of Azarenok, Fazio, Ivanenko, Leveque, Mackenzie, Stockie, Tao Tang, Huazhong Tang, and others (see bibliography [4], [22], [11], [7], [16], [14], [5], [6], [8], [9], [10], [12], [13], [15], [17], [18], [19], [20], [21], [23], [24]). These methods calculate the spatial positions of the nodes of the new mesh, some of them by solving an Euler-Lagrange equation, where others by optimizing proper energy metrics. All of them have as common factor that they can be combined with any numerical scheme (making appropriate modifications) for increasing its resolution.

In this work the evolving mesh is constructed such that it's spatial resolution is controlled via selective characteristics of the computing solution. Our main aim here is to experimentally study the behavior of finite difference schemes, with or without stabilization mechanisms, but under the regime of this adaptive evolving mesh. Our choice of classical schemes include for example the first order Roe scheme, the second order Lax-Wendroff and MacCormack schemes and also some TVD schemes. The adaptive procedure studied in this work is based on a mesh redistribution policy that evolves within every computational time step. The basic principle of the suggested mesh redistribution is to (re)distribute the nodes of the generated partition with respect to geometrical characteristics of the solution. These characteristics are defined through a positive functional of the solution, the so called *estimator function* [1], [2], [3]. Among other estimator functions for dynamic PDE's, like the arc-length and variance, we choose the curvature of the solution as such function, for its diffuse-less behavior. Experiments in [3] with finite element schemes, which approximate CL solutions like central finite difference schemes, have shown that this redistribution procedure has stabilization properties by its own. For example in this work we show that schemes like Lax Wendroff combined with our redistribution policy provide surprising stable solutions free of oscillations. From the numerical experiments produced in the course of this work we concluded to the following

advantages of the use of this particular Adaptive Grid Redistribution (**AGR**) method.

- The method when applied to classical second order schemes, that produce oscillating solutions, suppress the oscillations, producing TVD like approximations. Classical schemes like the Lax-Wendroff or the Mac-Cormack scheme become stable and produce reliable solutions.

- When applied to numerical schemes that do not satisfy entropy conditions, for example the original Roe scheme, the scheme approximates the entropy satisfying solution.

- The method works well for hyperbolic problems with source terms producing stable solutions for first and second order balanced schemes and can converge to correct steady states.

- The method can automatically detect, resolve and track steep wave fronts and discontinuities, without having to resort to finer grids.

- The **AGR** method is of linear complexity as we will see in the next section cf. Remark 3, so, its computational cost is in favor when compared e.g. to the stabilization mechanisms for high resolution TVD schemes, where we have to solve a number of Riemann problems in order to compute the corresponding stabilization limiters.

- The mechanism has been proved robust for all the applications presented and schemes used.

Following from the above, our main conclusion from the present work is that: numerical schemes when combined with the proposed mesh adaptation yields stable solutions free of oscillations.

The paper is organized as follows. In the next section we define the **AGR** algorithm. In Section 3 we present finite difference schemes for scalar CL, adjusted to non-uniform grids, which will be tested in the regime of the adaptive redistribution algorithm. Section 4 is devoted to numerical results. In Section 5 numerical schemes for systems of CL, with or without source terms, are detailed, and in Section 6 numerical results for the shallow water system of equations are presented.

2 The Adaptive Grid Redistribution method in one-dimension

Let X be a partition of a domain $[a,b]$ and $\{x_i\}_{i=0}^{N}$ are the corresponding nodes in increasing order, i.e. $a = x_0 < x_1 < .. < x_{N-1} < x_N = b$, then we introduce the following notation:

- with $\sigma(X)$ we denote the σ-algebra of X, which for finite partitions coincides with the superset of unions of sets from X.

- with **resolution** of X, we denote the measure which for any measurable set A is defined by

$$\textbf{resolution}(A) = \text{card}\{K \in X : K \subset A\},$$

i.e., the **resolution** of the partition X over some set A is the number of partition elements that A contains.

- if $U = \{u_i\}_{i=0}^N$ is a vector of values defined on elements of the partition X, then it shall be notated as $U(X)$. For compatibility with approximations given from finite difference schemes, in the following we shall identify any vector of function values $\{u_i\}_{i=0}^N$ evaluated on partition nodes $\{x_i\}_{i=0}^N$, i.e $U = U(X)$, with the locally constant function U, defined on $[a, b]$ by:

$$U = \sum_{i=0}^{N} u_i \ \mathbb{X}_{[x_{i-1}^+, x_i^+)} + u_N \ \mathbb{X}_{\{x_N^+\}} \tag{2.1}$$

where $\mathbb{X}_A$ denotes the characteristic function of the set A, and $\{x_i^+\}_{i=-1}^N$ denotes the points

$$x_{-1}^+ = x_0, \quad x_i^+ = x_{i+\frac{1}{2}}, i = 0, 1, .., N-1, \quad x_N^+ = x_N.$$

In the case of dynamic PDE's, where the numerical solution is constructed like a sequence of spatial approximations of solutions instances, the redistribution process could be applied before every evolution step. Let **Solver** denote a (general) finite difference scheme for solving some dynamic PDE. For $n = 1, 2, ...$ **Solver** gives sequentially the approximations U^n of the solution instances $u(t^n, \cdot)$ at given time moments $t^0 < t^1 < ... < t^n \in \mathbb{R}^+$, starting at $t^0 = 0$. The given initial data U^0 is defined on the uniform partition X^0 of the domain. In the case of uniform partition, the evolution step can be represented for $n = 1, 2, ...$ as an equation on the vector space $\mathbb{R}^{2(N+1)}$:

$$(X^n, \ U^n(X^n)) = (X^{n-1}, \ \textbf{Solver}(X^{n-1}, \ U^{n-1}(X^{n-1}))).$$

On meshes generated by the **AGR** process, it can represented by the system of vector equations:

$$\begin{aligned} (\tilde{X}, \ \tilde{U}(\tilde{X})) &= \textbf{AGR}(X^{n-1}, \ U^{n-1}(X^{n-1})), \\ (X^n, \ U^n(X^n)) &= (\tilde{X}, \ \textbf{Solver}(\tilde{X}, \ \tilde{U}(\tilde{X}))), \end{aligned} \tag{2.2}$$

where $\tilde{X}, \tilde{U}$ are temporal vectors. Observe that in the case where the **AGR** process returns the same vectors, i.e. the **AGR** coincide with the identity linear transformation then (2.2) reduces to the uniform evolution step. The **AGR** procedure, for a given vector of values $U = \{u_i\}_{i=0}^N$ defined on the nodes $\{x_i\}_{i=0}^N$ of the partition X, i.e. $U = U(X)$, is described by the following two sequential steps,

$$\begin{aligned} \tilde{X} &= \textbf{GMesh}(X, \ U(X)), \\ \tilde{U}(\tilde{X}) &= \textbf{Rec}(X, \ U(X), \ \tilde{X}). \end{aligned} \tag{2.3}$$

In the sequel we describe in detail the two steps **GMesh**, **Rec** of our redistribution algorithm **AGR**.

2.1 The GMesh step.

At the **GMesh** step of the **AGR** procedure (2.3), a new partition $\tilde{X}$ of spatial nodes $\{\tilde{x}_i\}_{i=0}^N$ is formed, with **resolution** controlled by selected characteristics of the numerical solution U. The step is accomplished in two phases.

At the *first phase* the selected characteristics are defined on the domain $[a,b]$ through a strictly positive functional g of the approximate solution, the *estimator function*. Since we are mainly interested on increasing the **resolution** over areas with discontinuities and, taking into account the results in [3], in this work we will use some power p of proper approximation of the solutions curvature defined by

$$(g \circ u)(x) = \frac{|u''(x)|}{(1+u'(x)^2)^{\frac{3}{2}}},$$

as estimator function for selecting the **resolutions** density of the new partition. The power p is left as free parameter taking values in the range $[0,1]$. At the discrete level, given the partition $\{x_i\}_{i=0}^N$, we approximate $g(u(x_i))$ with the inverse outer radius g_i of the plane points $A_j = (x_j, u_j), j = i-1, i, i+1$ of the numerical solution, that is $g_0 = g_N = \delta^p$ and for $i = 1, .., N-1$,

$$g_i = \Big(max\Big\{\delta,\ 2\frac{\|(A_{i+1}-A_i)\times(A_i-A_{i-1})\|}{\|A_{i+1}-A_i\|\ \|A_{i+1}-A_{i-1}\|\ \|A_i-A_{i-1}\|}\Big\}\Big)^p, \tag{2.4}$$

where $\|\cdot\|$ denotes the Euclidean norm, and δ is a very small positive number which ensures that g_i is strictly positive. In addition, at computational level, δ must be chosen such that g_i behaves monotonically as a function of p in (2.4), even for negligible values of p. In our experiments where the calculations was made with ANSI-C computational libraries, δ was fixed to the value 10^{-32}. In case of multidimensional solution $u = (u_1, .., u_M)^{\mathrm{T}}$, we intend to use the average of the coordinate curvatures as estimator function. Therefore in general, instead of the values g_i we keep the normalized version $g_i / \sum_{j=0}^N g_j$.

At the *second phase*, the partition $\tilde{X}$ is actually produced through $g_h \circ U$, the corresponding function of type (2.1) to values (2.4). Since we expect a new mesh $\tilde{X}$ with **resolution** that locally follows the function $g_h \circ U$, we conclude that the **resolution** measure must distributes in space like the measure $G_U(A) = \int_A (g_h \circ U)\, d\mu$ introduced by the positive function $g_h \circ U$, at least for sets from $\sigma(\tilde{X})$. Therefore the new partition $\tilde{X}$ is defined such that

$$\forall A \in \sigma(\tilde{X}) \quad G_U(A) = C\ \mathbf{resolution}(A), \quad \text{with } C = \frac{G_U([a,b])}{N}. \tag{2.5}$$

Applying sequentially (2.5) with the sets $A_i = [a, \tilde{x}_i], i = 1, 2, .., N$, where, their indexes declare also the **resolution** of the partition $\tilde{X}$ over them, we induce the so called in the literature of moving meshes *equidistribution principal*:

$$\tilde{x}_0 = a \text{ and for } i = 1, \ldots, N,\ \tilde{x}_i \text{ is such that } \int_a^{\tilde{x}_i} g_h(U(x))\, dx = \frac{i}{N}\int_a^b g_h(U(x))\, dx. \tag{2.6}$$

In the present work, the new mesh $\tilde{X}$ is defined through (2.6) but we note that other inverting algorithms could also be constructed through a more elegant treatment of the generic equidistribution principal (2.5). Let $G(x) = G_{U}([a,x])$ be the distribution function which is an increasing local linear function defined on the grid X since, $g_{h} \circ U$ is a positive function of type (2.1). Thus, G is completely defined from the values $G_i = G(x_i)$ of the G_{U} measure of the intervals $[a, x_i]$ that is, $G_0 = 0$ and for $i = 1, .., N$

$$G_i = G_{U}([a,x_i]) = \int_{[a,x_i]} (g_{h} \circ U)\, d\mu = G_{i-1} + \int_{x_{i-1}}^{x_i} g_{h}(U(x))\, dx \overset{(2.1)}{=} G_{i-1} + (g_{i-1} + g_i)(x_i - x_{i-1})/2. \tag{2.7}$$

From (2.6) it comes out that the $\tilde{x}_i$ node is given by inverting the equation $G(\tilde{x}_i) = \frac{i}{N} G(\tilde{x}_N)$ so the nodes of the new mesh $\tilde{X}$ are, $\tilde{x}_0 = a,\ \tilde{x}_N = b$ and, for $i = 1, .., N-1,\ k_0 := 0$

$$\tilde{G}_i = \frac{i}{N} G_N, \quad k_i = \max_{k_{i-1} \le \ell \le N} \{\ell : G_\ell \le \tilde{G}_i\}, \quad \tilde{x}_i = x_{k_i} + \frac{x_{k_i+1} - x_{k_i}}{G_{k_i+1} - G_{k_i}} (\tilde{G}_i - G_{k_i}). \tag{2.8}$$

2.2 The Rec step.

At the **Rec** step of the **AGR** procedure (2.3), a new numerical solution $\tilde{U} = \{\tilde{u}_i\}_{i=0}^{N}$ of type (2.1) is defined on the new grid $\tilde{X}$, by reconstructing U from the old grid X to the new one. Since the finite difference schemes used in this work produce conservative solutions, the reconstruction step in the current work is done so that it is locally conservative on each interval $[\tilde{x}_{i-1}^{+}, \tilde{x}_i^{+}]$ of the new mesh. That is, for $i = 0, .., N$, we require

$$\int_{\tilde{x}_{i-1}^{+}}^{\tilde{x}_i^{+}} \tilde{U}(x)\, dx = \int_{\tilde{x}_{i-1}^{+}}^{\tilde{x}_i^{+}} U(x)\, dx.$$

Since the functions $U, \tilde{U}$ are of type (2.1) defined on the partitions $X, \tilde{X}$,

$$\tilde{u}_i(\tilde{x}_i^{+} - \tilde{x}_{i-1}^{+}) = \int_{\tilde{x}_{i-1}^{+}}^{\tilde{x}_i^{+}} U(x)\, dx = \int_{x_{k_i}^{+}}^{\tilde{x}_i^{+}} U(x)\, dx + \int_{x_{k_{i-1}}^{+}}^{x_{k_i}^{+}} U(x)\, dx - \int_{x_{k_{i-1}}^{+}}^{\tilde{x}_{i-1}^{+}} U(x)\, dx$$

where, using the real line ordering of the domain $[a,b]$, $x_{k_i}^{+}$ denotes the larger node from the old partition which does not exceed the node $\tilde{x}_i^{+}$ of the new partition. Consequentially the new values $\tilde{U}$ are, $\tilde{u}_0 = u_0$ and for $i = 1, .., N,\ k_0 := 0$:

$$\begin{aligned} k_i &= \max_{k_{i-1} \le \ell \le N} \{\ell : x_\ell^{+} \le \tilde{x}_i^{+}\}, \\ \tilde{u}_i &= \big\{(\tilde{x}_i^{+} - x_{k_i}^{+}) u_{k_i+1} + \sum_{\ell=k_{i-1}+1}^{k_i} (x_\ell^{+} - x_{\ell-1}^{+}) u_\ell - (\tilde{x}_{i-1}^{+} - x_{k_{i-1}}^{+}) u_{k_{i-1}+1} \big\} / \big(\tilde{x}_i^{+} - \tilde{x}_{i-1}^{+}\big) \end{aligned} \tag{2.9}$$

The **Gmesh** step of the **AGR** process can be graphically represented like in Figure 1 (left). Starting with some initial data (solid line) we calculate the estimator function (dotted line) through (2.4) and then, the corresponding distribution function (increasing dotted line)

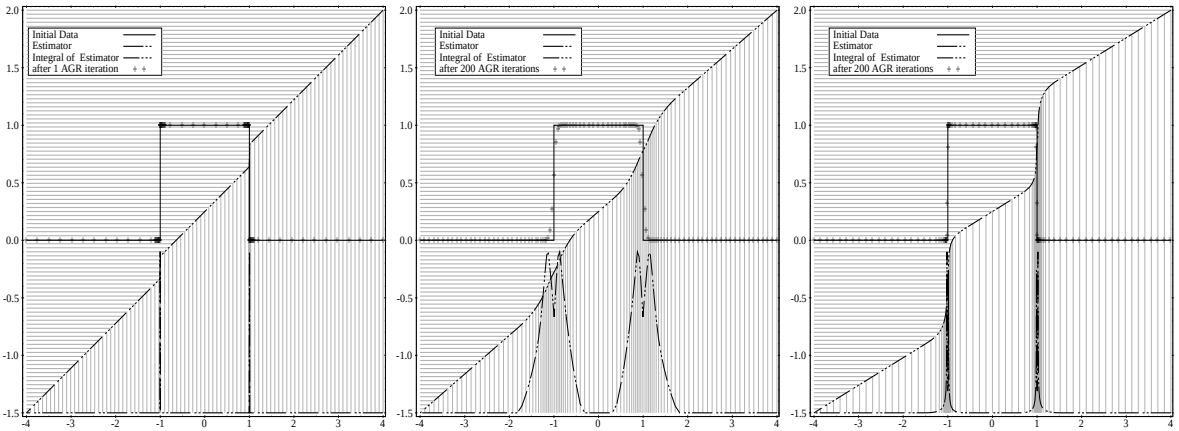

Figure 1: Iteration of AGR procedure applied on Riemann Data. After 1 iteration (left) and after 200 iter's for $p = 0.023$ (middle) and $p = 0.1$ (right).

using (2.7). The proposed mesh (x position of the vertical lines, under the distribution function) is given by inverting a uniform mesh through the distribution function using (2.8) (y position of the horizontal lines, over the distribution function). Notice that y-coordinates are valid only for the data while either the estimator or its distribution function was scaled vertically to fit on the viewing graph area.

Remark 1. *The action of the parameter p.*

The action of the parameter p in the **AGR** procedure, can also be seen in Figure 1 (middle, right). For $p = 0$ observe from (2.7) that $G_i = x_i - x_0$, so the proposed partition from (2.8) coincide with the uniform partition of $[a, b]$. As the value of p increases, the proposed partition becomes more dense on the discontinuous areas but, over some value of p the process becomes unreliable because the partition lacks of nodes over smooth areas. In our experiments with finite difference schemes a valid range for the parameter p was $[0.02, 0.05]$ for first order and $[0.04, 0.12]$ for second order schemes.

Remark 2. *Smoothing effect of the AGR process.*

Results in [3] show that the reconstructed function $\tilde{U}$ of the **AGR** procedure, is more diffusive than the initial function U. Indeed this diffusive behavior can be observed by applying the **AGR** procedure iteratively on some initial data, i.e. starting with some initial data U^0 on the uniform partition X^0 of $[a, b]$, we define the sequence $(X^n, U^n(X^n)) = \mathbf{AGR}(X^{n-1}, U^{n-1}(X^{n-1})), n = 1, 2, ...$ Experiments with Riemann initial data show that the above sequence attains a limit pair $(X, U(X))$. In Figure 1 (middle, right) we present the results after 200 iterations, for $p = 0.023$ (middle) and for $p = 0.1$ (right). In Figure 2 (middle, right) we present the trajectories of the nodes for the above iterative process. Notice that after about 20 iterations for $p = 0.023$ and about 80 for $p = 0.1$ the position of all the nodes X became constant. But from (2.9) one can observe that reconstructing any function

on the same grid leads to the same function, i.e. $U(X) = \mathbf{Rec}(X,\ U(X),\ X)$, so we conclude that also the reconstructed values $U(X)$ after the above iterations became constant. Figure 2 (left) show the smoothing effect of the **AGR** process on the initial data after 200 iterations, and for various values of p.

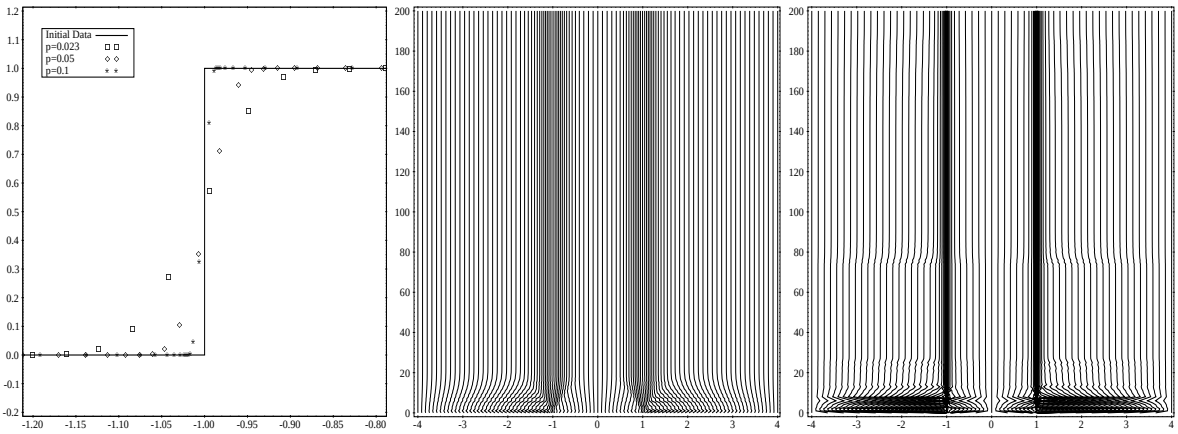

Figure 2: Focus in the sock area (left), after 200 iter's of the AGR procedure and for various values of parameter p. Trajectories of mesh nodes, for $p = 0.023$ (middle) and for $p = 0.1$ (right).

Remark 3. *Complexity of the AGR process.*

Observe that either the **Gmesh** step (2.7), (2.8) or the **Rec** step (2.9), of the **AGR** process is of linear complexity.

2.3 Redistribution in the steady state regime.

The diffusive behavior of the **AGR** process might be one of the reasons that second order schemes when combined with the redistribution step produce non-oscillatory approximations; however, in the case where the current and the new mesh are almost the same the redistribution step it should be avoided. In addition, in the case of very sensitive CL problems with steady states we noticed that the repeated application of **AGR** process prevents the conservation of the steady states. Taking in to account these observations, in the numerical experiments of this work we adopt a new version of the **AGR** process:

$$\begin{aligned} &\tilde{X} = \mathbf{GMesh}(X, U) \\ &\text{if } (|X - \tilde{X}| > D) \quad \tilde{U} = \mathbf{Rec}(X, U, \tilde{X}) \\ &\text{else} \qquad\qquad\quad \tilde{X} = X,\ \tilde{U} = U \end{aligned}$$

where, D is a "cutoff" level for the relative mean displaysmend $|X - \tilde{X}|$ between the current and the proposed meshes, under of which the **Rec** step is avoided. The relative mean displaysmend is measured in the ℓ_1 vector norm that is, $|X - \tilde{X}| = \frac{\frac{1}{N+1}\sum_{i=0}^{N}|x_i - \tilde{x}_i|}{b-a}$. Experiments

show that in the case of first order schemes or those with extra stabilization mechanisms, the **AGR** process produce results with higher resolution when the cutoff parameter D range values of order 10^{-2} while, due to the oscillatory behavior of the second order schemes in that case the cutoff level should be of order 10^{-3} or less. In all of the cases, when the problem impose steady states a non-zero value of D is crucial for the solving process in order to reach the time invariant solution.

3 Numerical schemes on non-uniform grids (1D scalar equations)

In this section we review some well known classical numerical schemes, implemented at the **Solver** step, for the solution of a scalar conservation law,

$$u_t + f(u)_x = 0, \tag{3.1}$$

and present them in their non-uniform grid formulation. We will also state some well known disadvantages that these schemes exhibit.

We will approximate the solution $u(t,x), x \in \mathbb{R}, t \geq 0$, of (3.1) by the discrete values $u_i^n, i \in \mathbb{Z}, n \in \mathbb{N}$ and in order to do so we consider a grid of points $x_{i+\frac{1}{2}}$, and define the computational cells and their lengths as

$$C_i = [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}], \qquad \Delta x_i = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}} > 0.$$

We also denote $x_i = (x_{i-\frac{1}{2}} + x_{i+\frac{1}{2}})/2$ the centers of the partition cells. The values u_i^n will be approximations of the averages of the exact solution over the cell

$$u_i^n \approx \frac{1}{\Delta x_i} \int_{C_i} u(t^n, x) dx,$$

with t^n define the discrete times.

For consistency we present the schemes in the usual conservative formulation with explicit time stepping,

$$u_i^{n+1} = u_i^n - \frac{\Delta t}{\Delta x_i} \left(f_{i+\frac{1}{2}}^n - f_{i-\frac{1}{2}}^n \right) \tag{3.2}$$

where $f_{i+\frac{1}{2}}^n, i = 1, 2, \ldots, N-1$ is the numerical flux function that determines the scheme.

First Order Roe Scheme. The numerical flux of the well known Roe's first order upwind scheme is given by

$$f_{i+\frac{1}{2}}^{ROE} = \frac{1}{2} \left(f_{i+1} + f_i - |a_{i+\frac{1}{2}}| \Delta u_{i+\frac{1}{2}} \right), \tag{3.3}$$

where $\Delta u_{i+\frac{1}{2}} = u_{i+1} - u_i$ and $|a_{i+\frac{1}{2}}|$ is the characteristic speed defined by

$$a_{i+\frac{1}{2}} = \begin{cases} \dfrac{f_{i+1} - f_i}{u_{i+1} - u_i}, & u_{i+1} \neq u_i \\ a_i = \dfrac{\partial f}{\partial u}, & u_{i+1} = u_i \end{cases}. \tag{3.4}$$

A well known drawback of Roe's schemes is that it may resolve non-physical solutions by admitting stationary entropy-violating expansion shocks. Various *entropy corrections* have been proposed, see [15]. For example we can replace the moduli of a in (3.3) with

$$\psi(a) = \max(\delta, |a|)$$

or the smoother form

$$\psi(a) = \begin{cases} |a| & |a| \geq \delta \\ (a^2 + \delta^2)/2\delta & |a| < \delta \end{cases},$$

with

$$\delta_{i+\frac{1}{2}} = \max[0, a_{i+\frac{1}{2}} - a_i, a_{i+1} - a_{i+\frac{1}{2}}], \quad \delta_{i-\frac{1}{2}} = \max[0, a_{i-\frac{1}{2}} - a_{i-1}, a_i - a_{i-\frac{1}{2}}].$$

The Roe numerical flux is then defined as

$$f^{ROE}_{i+\frac{1}{2}} = \frac{1}{2}\left(f_{i+1} + f_i - \psi(a_{i+\frac{1}{2}})\Delta u_{i+\frac{1}{2}}\right), \tag{3.6}$$

In our numerical examples to follow we will see that when the adaptive mechanism is applied the above entropy corrections are not necessary for the scheme to produce approximations of entropy solutions. This is a first order diffusive scheme, but it serves as a key ingredient in developing higher order methods.

The Local Lax-Friedrichs Scheme (LF). A well known first order scheme that produce approximations of entropy solutions is the local Lax-Friedrichs scheme. Its numerical flux function is given by

$$f^{LF}_{i+\frac{1}{2}} = \frac{1}{2}\left(f_{i+1} + f_i - a_{max}\Delta u_{i+\frac{1}{2}}\right), \tag{3.7}$$

where $a_{max} = \max\{|f'(u_i)|, f'(u_{i+1})|\}$. This scheme produces diffusive numerical solutions and smears discontinuities and was used for comparison reasons only.

The Lax-Wendroff Scheme (LW). A natural generalization of the classical second order Lax-Wendroff scheme for non-uniform grids can be given as a combination of the first order Roe flux plus a correction flux, see [15], as,

$$f^{LW}_{i+\frac{1}{2}} = f^{ROE}_{i+\frac{1}{2}} + \frac{1}{2}\left(\frac{\Delta x_i}{0.5\left(\Delta x_{i+1} + \Delta x_i\right)} - \frac{\Delta t}{0.5\left(\Delta x_{i+1} + \Delta x_i\right)}|a_{i+\frac{1}{2}}|\right)|a_{i+\frac{1}{2}}|\Delta u_{i+\frac{1}{2}}, \tag{3.8}$$

with $a_{i+\frac{1}{2}}$ as defined in (3.4). It is easy to see that for a uniform grid numerical flux (3.8) is reduced to the classical Lax-Wendroff numerical flux. As for uniform grids this scheme has highly oscillatory behavior near discontinuities and without an entropy correction can not produce entropy solutions.

The Mac-Cormack Scheme. One of the most classical second order schemes is the well known two step Mac-Cormack scheme

$$u_i^{(1)} = u_i^n - \frac{\Delta t}{\Delta x_i}\left(f_{i+1}^n - f_i^n\right), \tag{3.9}$$

$$u_i^{(2)} = u_i^n - \frac{\Delta t}{\Delta x_i}\left(f_{i+1}^{(1)} - f_i^{(1)}\right), \tag{3.10}$$

$$u_i^{n+1} = \frac{1}{2}\left(u_i^{(1)} + u_i^{(2)}\right). \tag{3.11}$$

This simple scheme also exhibits oscillatory behavior near discontinuities and it can not produce the correct entropy solutions.

The MUSCL-TVD Scheme. Here we present a second order slope-limiting scheme based on the MUSCL-TVD interpolation formula on non-uniform grids. We define

$$u_{i+\frac{1}{2}}^L = u_i + \frac{h_{i+1}}{2}\left(\frac{u_i - u_{i-1}}{h_i}\right)\Phi\left(\theta_i\right)$$

$$u_{i+\frac{1}{2}}^R = u_{i+1} - \frac{h_{i+1}}{2}\left(\frac{u_{i+2} - u_{i+1}}{h_{i+2}}\right)\Phi\left(\frac{1}{\theta_{i+1}}\right),$$

where $h_i = x_i - x_{i-1}$ and θ_i given by

$$\theta_i = \frac{\left(u_{i+1} - u_i\right)/h_{i+1}}{\left(u_i - u_{i-1}\right)/h_i}.$$

The Φ is a limiter function, and there are several options on choosing Φ from. Some of the most popular limiters are, the MinMod (MM) limiter

$$\Phi(\theta) = \max(0, \min(1, \theta)),$$

the Superbee (SB) limiter

$$\Phi(\theta) = \max(0, \min(2\theta, 1), \min(\theta, 2)),$$

the VanLeer (VL) limiter

$$\Phi(\theta) = \frac{|\theta| + \theta}{1 + |\theta|},$$

and the Monotonized Central (MC) limiter

$$\Phi(\theta) = \max(0, \min((1+\theta)/2, 2, 2\theta)).$$

The numerical flux for the MUSCL scheme can then be expressed, based on the Roe flux (without an entropy fix), as

$$f_{i+\frac{1}{2}}^{MUSCL} = \frac{1}{2}\left(f(u_{i+\frac{1}{2}}^R) + f(u_{i+\frac{1}{2}}^L) - |a_{i+\frac{1}{2}}|(u_{i+\frac{1}{2}}^R - u_{i+\frac{1}{2}}^L)\right), \tag{3.12}$$

here $a_{i+\frac{1}{2}}$, is evaluated as before, with u_{i+1} and u_i replaced by $u_{i+\frac{1}{2}}^R$ and $u_{i+\frac{1}{2}}^L$. In a similar way one can define a MUSCL flux based on the LF flux.

Second order TVD schemes reduce to first order at extrema and there are also differences in the behavior for each limiter, for example the last three limiters have been shown to exhibit sharper resolution of discontinuities, since they do not reduce the slope as severely as MM near a discontinuity.

4 Numerical examples for scalar conservation laws

Because all the schemes presented above are explicit in time, stability requires the time step Δt to satisfy the CFL condition, as to ensure that the time step is small enough that waves from neighboring cell do not interact, with $CFL \leq 1$. Based on that we use a variable time step calculated from

$$CFL = \Delta t \max_i \left\{ \left| \frac{a^+_{i-\frac{1}{2}}}{\Delta x_i} \right|, \left| \frac{a^-_{i+\frac{1}{2}}}{\Delta x_i} \right| \right\} \tag{4.1}$$

with $a^+ = \max(0, a)$ and $a^- = \min(0, a)$.

4.1 Burger's Equation

We first consider numerical experiments for the Burgers equation in the inviscid limit

$$u_t + \left(\frac{u^2}{2} \right)_x = 0 \tag{4.2}$$

The analytical solutions to this problem can be found for example in [12].

The first problem is an academic test case with initial conditions, for $x \in [-5, 5]$,

$$u(x,0) = \begin{cases} -1.0 & x < 0 \\ 1.0 & x \geq 0 \end{cases}, \tag{4.3}$$

and has the following exact solution also known as a *transonic rarefaction*,

$$u(x,0) = \begin{cases} -1.0 & x < -t \\ x/t & -t \leq x \leq t \\ 1.0 & x > t \end{cases}. \tag{4.4}$$

As pointed out in [15], dealing with transonic rarefactions properly is an important component in the development of successful methods. Results for this problem are presented in Figs 3 and 4 at $t = 2s$. A grid of 61 points was used for all schemes with CFL number equal to 0.9. All the schemes presented in the previous section, with the exception of the LF scheme, can not produce the correct entropy solution. By the application of the adaptive mechanism all the schemes calculate the correct solution. The most accurate results were produced with the MUSCL adaptive scheme using the SB limiter. We note here that when the adaptive mechanism was applied to the MUSCL scheme we were able to produce similar results for all limiters.

The second problem was presented in [19] and has initial conditions given by

$$u(x,0) = \begin{cases} 1.0 & x \in [0.2, 2.0] \\ -0.5 & x \in (2.0, 3.0] \\ -1.0 & x \in (3.0, 4.8] \\ 0.0 & \text{else} \end{cases}. \tag{4.5}$$

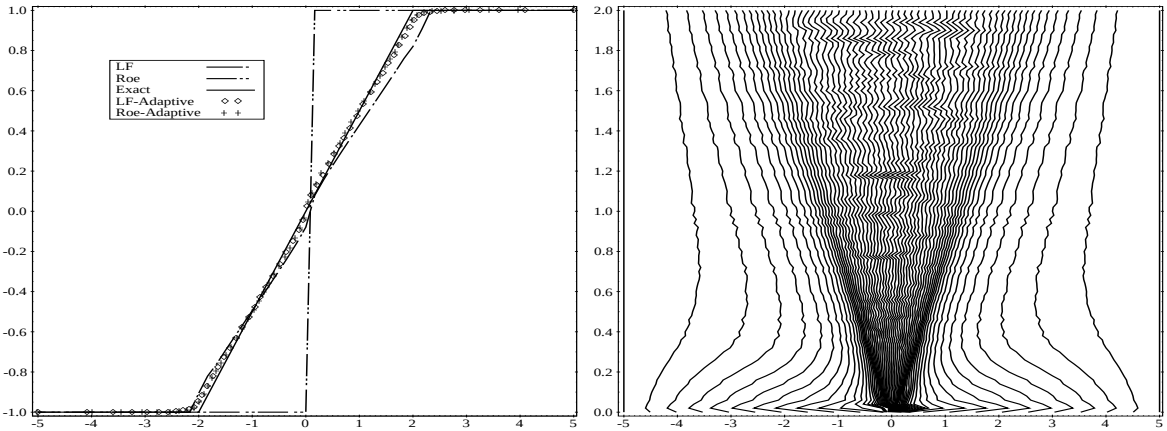

Figure 3: Transonic rarefaction problem: Numerical solution (left) and grid point trajectories(left) for the adaptive Roe scheme ($p = 0.09, D = 0$).

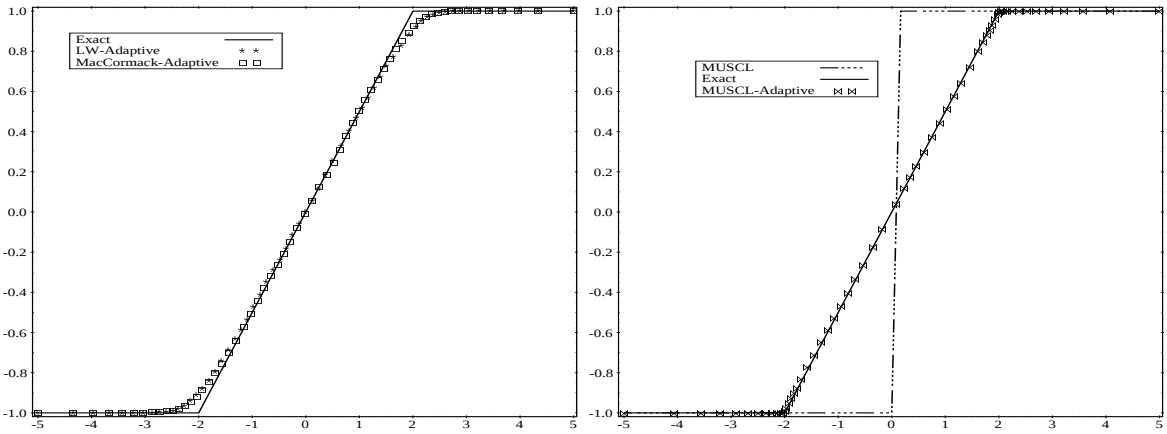

Figure 4: Transonic rarefaction problem: Numerical solution (left) for the LW and MacCormack schemes ($p = 0.062, D = 0$) and (right) for the MUSCL scheme ($p = 0.10, D = 0$).

The solution domain is for $x \in [0, 5]$ with homogeneous Dirichlet conditions. This problem includes two shocks initially at $x = 2$ and $x = 3$, moving to the right and left respectively, and two expansion discontinuities at $x = 0.2$ and $x = 4.8$ also expanding to the right and left respectively. The two shocks collide at time $t = 1s$ and form a single shock moving to the left. The numerical results are presented in Figs. 5 and 6 and for $t = 2$ when the shocks have combined into a single one. A grid of 121 points was used for all schemes with CFL number equal to 0.9. All the schemes produce improved results when the adaptive mechanism is imposed, when compared to those produced in a uniform mesh. It is impressive that even the second order oscillatory LW and MacCormack schemes are now able to produce accurate solutions. The performance of the MUSCL using the MC limiter scheme is also greatly improved, since in the non-adaptive case the scheme fails to produce the entropy correct

solution at expansions. Similar observations were made for the other limiters as well. In Fig.6 one can also see the ability of the mesh to capture and follow the evolution of the solution as demonstrated by the grid point trajectories.

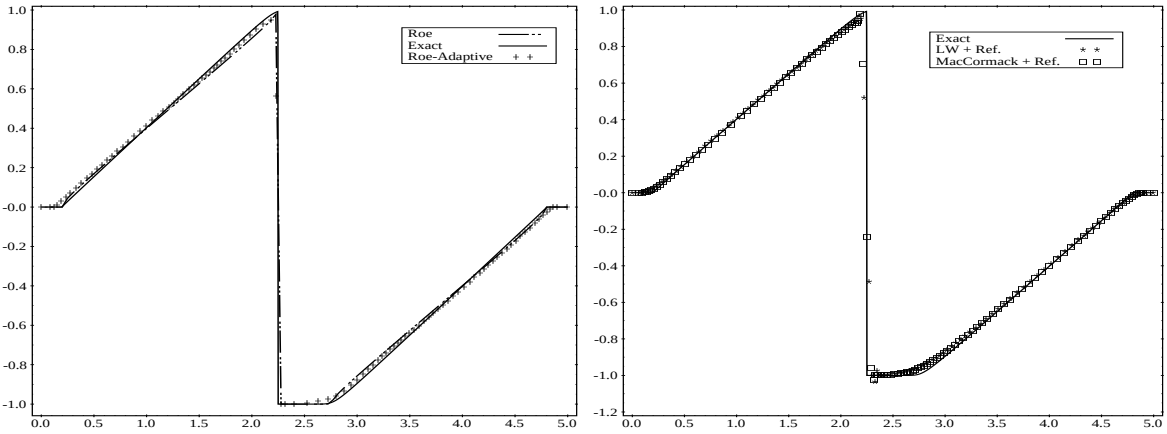

Figure 5: The second Burger's problem: Numerical solution Roe's scheme (left) ($p = 0.079, D = 0.0224$) and the LW and MacCormack schemes (right) ($p = 0.10, D = 0$).

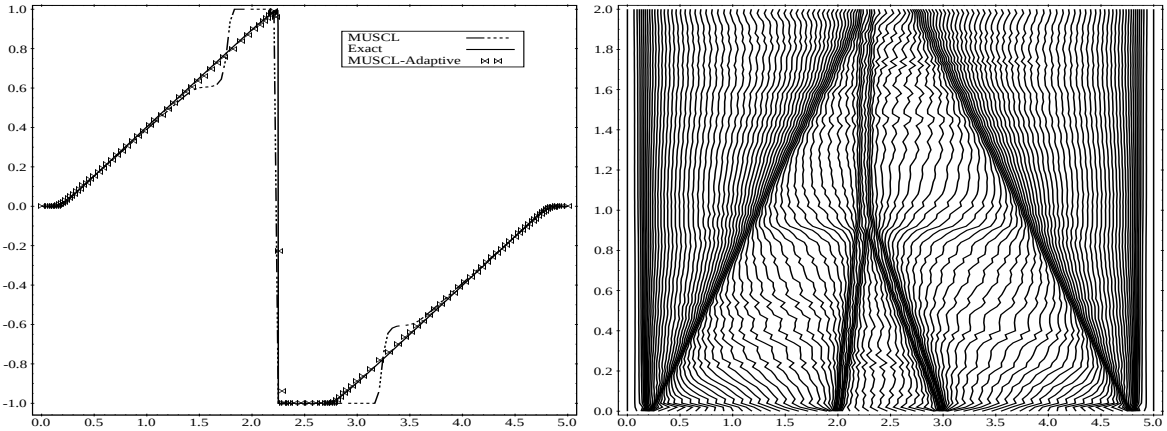

Figure 6: The second Burger's problem: Numerical solution (left) and grid point trajectories (left) for the adaptive MUSCL scheme ($p = 0.09, D = 0$).

The third problem has initial profile of a smooth wave given by

$$u(x,0) = 0.5\sin(\pi x) + \sin(2\pi x), \qquad x \in [0,1],$$

with periodic boundary conditions imposed. The solution propagates to the right, steepening until the singularity time $t_c = 64/(129\pi)$. Results are presented in Figs. 7 and 8 for the second order schemes for a 61 point grid and with a CFL number of 0.6. at $t = 1s$, and verify the observations made for the second example. Considerably better results are obtained for the adaptive methods, compared to the ones in uniform grid, especially in shock resolution

because of the clustering of grid points near the shock. Again the grid trajectories show the formation and following of the shock layer.

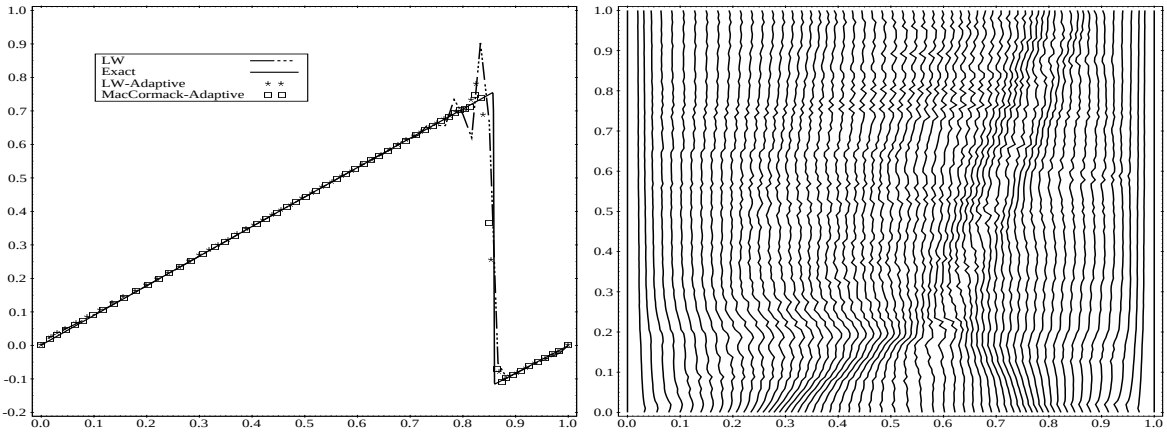

Figure 7: The third Burger's problem: Numerical solution for the LW and MacCormack schemes (left) and grid point trajectories (left) for the LW scheme ($p = 0.06, D = 0$).

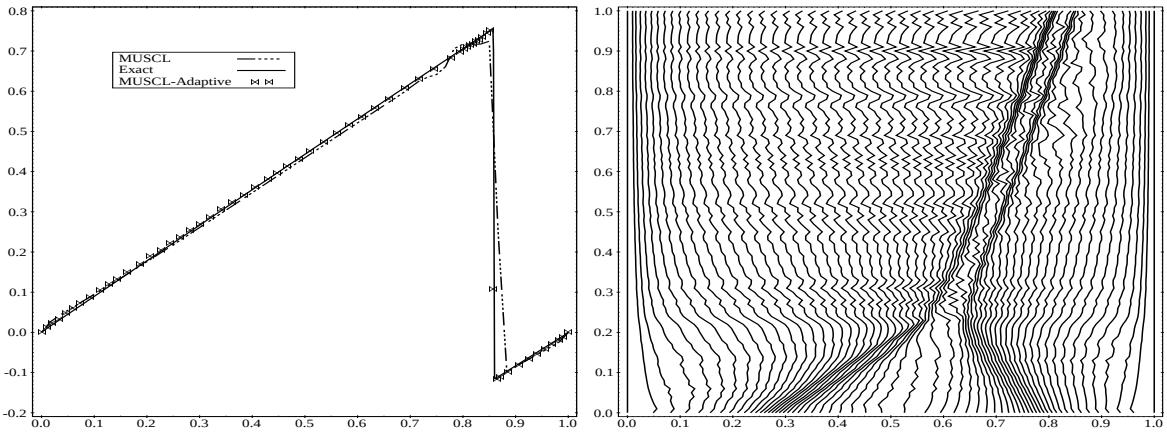

Figure 8: The third Burger's problem: Numerical solution (left) and grid point trajectories (left) for the MUSCL scheme ($p = 0.118, D = 0$).

The fourth problem has initial conditions

$$u(x, 0) = 1 - 0.02x, \qquad x \in [0, 100],$$

and has been selected to illustrate convergence to a discontinuous steady state. The parameters for the calculations were 61 grid points and a CFL number equal to 0.9. The stabilizing tendency of the adaptive method can be clearly seen since it does not produce oscillatory results, as shown in Figs 9 and 10 for $t = 125s$. One can also observe the improvement in all the calculations when the adaptive mechanism is imposed. The effect in the grid movement

of a nonzero value for the "cutoff" parameter D, for first order schemes, can be seen in Fig. 9, following the remarks made in Section 2.

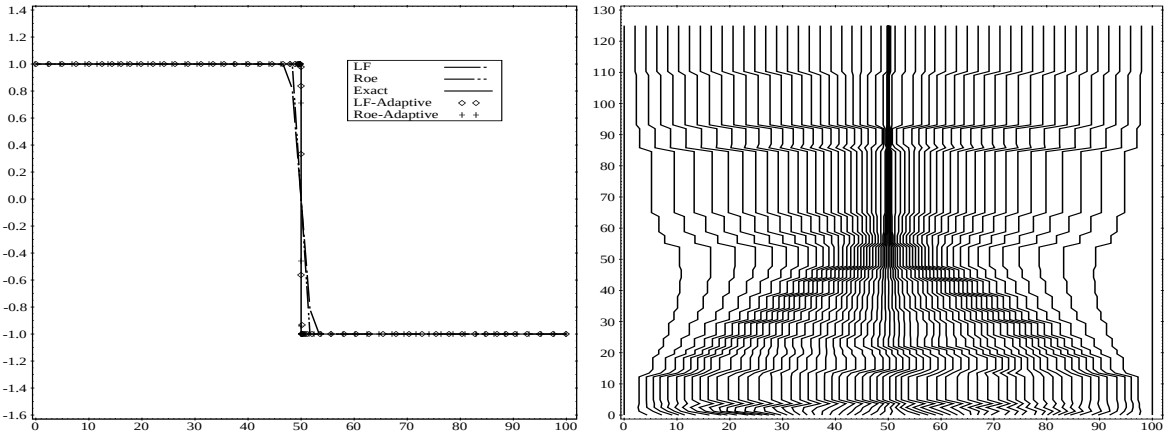

Figure 9: The fourth Burger's problem: Numerical solutions for the Roe and LF schemes (left) and grid point trajectories (left) for the adaptive Roe scheme ($p = 0.10, D = 0.013$).

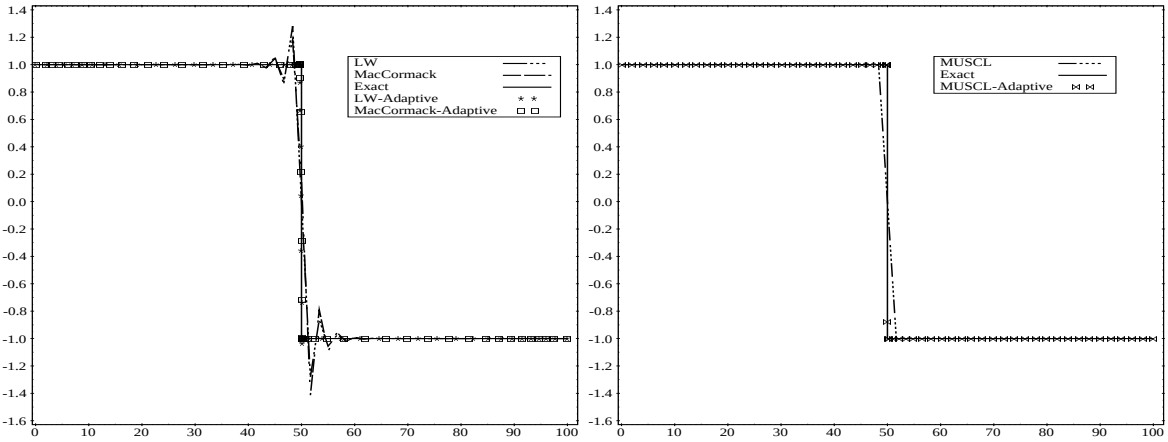

Figure 10: The fourth Burger's problem: Numerical solutions for the LW and MacCormack schemes (left) and MUSCL scheme (right) ($p = 0.13, D = 0$).

4.2 Nonconvex conservation law and the Buckley-Leverett equation

In this section we first apply the adaptive grid method to the scalar conservation law with nonlinear nonconvex flux,

$$f(u) = (u^2 - 1)(u^2 - 4)/4, \quad \text{for} \quad x \in [0, 1]. \tag{4.6}$$

The initial profile is $u(x, 0) = -2\text{sign}(x)$. As pointed out in [24] solving this Riemann problem for a scalar conservation law with nonconvex flux leads to difficulties with some numerical

schemes. Numerical solutions for all schemes are shown in Fig. 11 at $t = 1.2$, in a 101 point grid and for CFL number equal to 0.8. Again by imposing the adaptive mechanism we are able to produce quality results for all schemes.

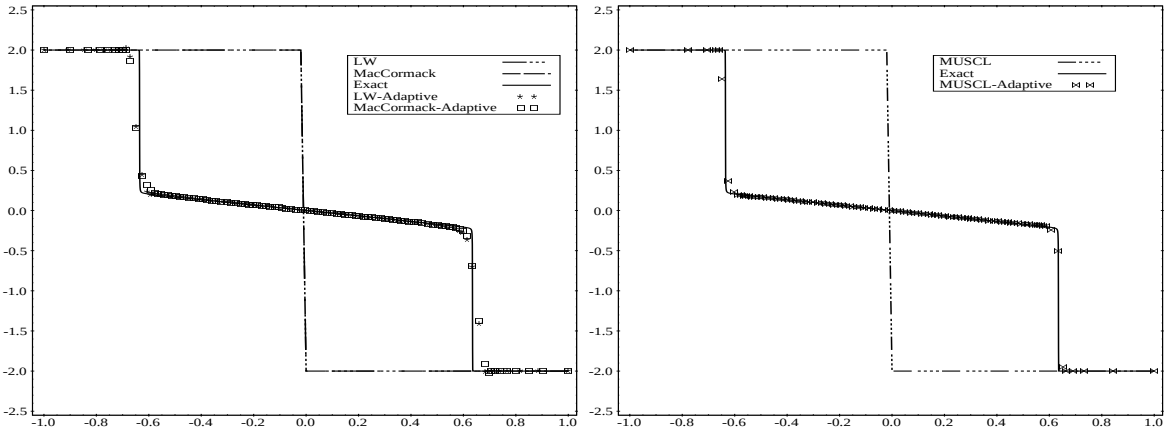

Figure 11: Nonconvex flux problem: Numerical solutions for the LW and MacCormack schemes (left) and MUSCL scheme (right) ($p = 0.069, D = 0$).

Next we consider the scalar Buckley-Leverett equation with the flux function

$$f(u) = \frac{u^2}{u^2 + 0.5(1-u)^2}$$

with initial and boundary conditions for $x \in [0, 2]$,

$$u(x,0) = \frac{1}{1+10x}, \qquad u(0,t) = 1, \qquad u(1,t) = \frac{1}{21}.$$

The solution for the adaptive and uniform LW and MacCormack schemes is shown in Fig. 12, in a 101 point grid. The highly oscillatory behavior of these schemes in totally suppressed by the adaptive mechanism resolving and following the shock layer much more accurately. Improvement in the shock resolution can also be observed for the MUSCL scheme in Fig. 13.

5 Numerical schemes for non-uniform grids (1D systems of equations)

We consider the general system of conservation laws with a source term added, i.e.

$$\mathbf{U}_t + \mathbf{F}(\mathbf{U})_x = \mathbf{G}(\mathbf{U}), \quad \mathbf{U} \in \mathbb{R}^M. \tag{5.1}$$

Where $\mathbf{F}(\mathbf{U})$ is the flux function and $\mathbf{G}$ is the source term. We can numerically approximate system (5.1) by using the explicit conservative numerical scheme with a source term approximation within the finite difference-volume frame, as

$$\mathbf{U}_i^{n+1} = \mathbf{U}_i^n - \frac{\Delta t}{\Delta x_i}\left[\mathbf{F}^n_{i+\frac{1}{2}} - \mathbf{F}^n_{i-\frac{1}{2}}\right] + \frac{\Delta t}{\Delta x_i}\widetilde{\mathbf{G}}_i^n, \tag{5.2}$$

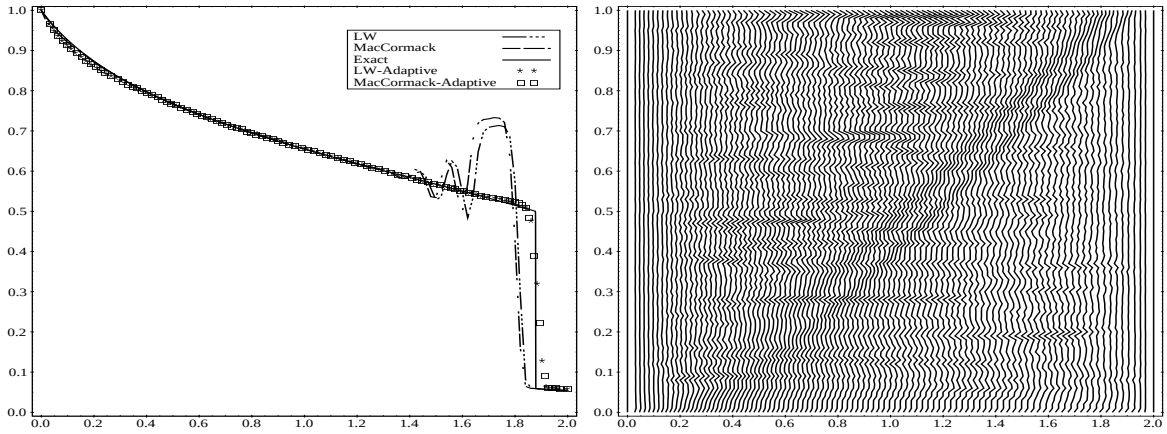

Figure 12: Buckley-Leverett problem: Numerical solutions for the LW and MacCormack schemes (left) and grid point trajectories for the adaptive LW scheme ($p = 0.076, D = 0$).

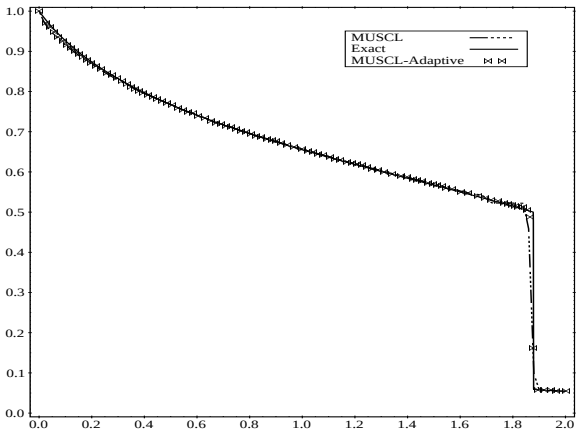

Figure 13: Buckley-Leverett problem: Numerical solutions for the MUSCL scheme ($p = 0.10, D = 0.0088$)

where again $\mathbf{F}^n_{i+\frac{1}{2}}, i = 1, 2, \ldots, N-1$ is the numerical flux function, and the approximation of the source term $\widetilde{\mathbf{G}}^n_i$ is considered as the cell average value (numerical source integral over the computational cell) satisfying the relationship

$$\widetilde{\mathbf{G}}^n_i = \frac{1}{\Delta x_i} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \mathbf{G}(x, \mathbf{U}^n) dx, \tag{5.3}$$

and after choosing the numerical flux function it remains to choose an appropriate approximation (depending on the scheme used) for the numerical source integral $\widetilde{\mathbf{G}}^n_i$.

For the stationary case, $\mathbf{U}_t = 0$, the flux function and source term balance

$$\mathbf{F}(\mathbf{U})_x = \mathbf{G}(\mathbf{U}).$$

Therefore, an accurate numerical scheme should also balance the numerical flux with the

source term approximation,

$$\mathbf{F}^n_{i+\frac{1}{2}} - \mathbf{F}^n_{i-\frac{1}{2}} = \widetilde{\mathbf{G}}^n_i \tag{5.4}$$

The numerical schemes presented next incorporate the numerical source integral in such a way that satisfy (5.4).

5.1 Roe's scheme for systems of conservation laws with source terms

The numerical flux for the Roe upwind scheme can be written in the form

$$\mathbf{F}^{ROE}_{i+\frac{1}{2}} = \frac{1}{2}\left[\mathbf{F}_{i+1} + \mathbf{F}_i - \mathbf{J}_{i+\frac{1}{2}}(\mathbf{U}_{i+1} - \mathbf{U}_i)\right]. \tag{5.5}$$

The matrix $\mathbf{J}_{i+\frac{1}{2}}$ is the Roe-type linearization and satisfies

$$\mathbf{F}_{i+1} - \mathbf{F}_i = \mathbf{J}_{i+\frac{1}{2}}(\mathbf{U}_{i+1} - \mathbf{U}_i), \tag{5.6}$$

and has real eigenvalues ($\tilde{a}^k_{i+\frac{1}{2}}$), $k = 1, \ldots, M$ and a complete set of eigenvectors ($\tilde{\mathbf{e}}^k_{i+\frac{1}{2}}$). Representing now $\mathbf{U}_{i+1} - \mathbf{U}_i$ in terms of the eigenvectors of the Roe linearization, i.e.

$$\mathbf{U}_{i+1} - \mathbf{U}_i = \sum_{k=1}^{M} \alpha^k_{i+\frac{1}{2}} \tilde{\mathbf{e}}^k_{i+\frac{1}{2}} \tag{5.7}$$

then from (5.6)

$$\mathbf{J}_{i+\frac{1}{2}}(\mathbf{U}_{i+1} - \mathbf{U}_i) = \sum_{k=1}^{M} \tilde{a}^k_{i+\frac{1}{2}} \alpha^k_{i+\frac{1}{2}} \tilde{\mathbf{e}}^k_{i+\frac{1}{2}}. \tag{5.8}$$

Then the Roe numerical flux can be written as

$$\mathbf{F}^{ROE}_{i+\frac{1}{2}} = \frac{1}{2}\left[\mathbf{F}_{i+1} + \mathbf{F}_i - \sum_{k=1}^{M} |\tilde{a}^k_{i+\frac{1}{2}}| \alpha^k_{i+\frac{1}{2}} \tilde{\mathbf{e}}^k_{i+\frac{1}{2}}\right]. \tag{5.9}$$

The $\alpha_{i+\frac{1}{2}}$ (wave strengths) values follow from (5.7).

Source term Upwinding. The integral of $\mathbf{G}$, in equation (5.3), over the cell $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$ is first written into a sum of two integrals in the $[x_{i-\frac{1}{2}}, x_i]$, $[x_i, x_{i+\frac{1}{2}}]$ as,

$$\frac{1}{\Delta x_i}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \mathbf{G}(x, \mathbf{U}^n)dx = \frac{1}{\Delta x_i}\left[\int_{x_{i-\frac{1}{2}}}^{x_i} \mathbf{G}(x, \mathbf{U}^n)dx + \int_{x_i}^{x_{i+\frac{1}{2}}} \mathbf{G}(x, \mathbf{U}^n)dx\right]. \tag{5.10}$$

The upwind definition of the approximated source terms $\widetilde{\mathbf{G}}^n_i$ for nonuniform grids is given by

$$\widetilde{\mathbf{G}}^n_i = \frac{1}{\Delta x_i}\left[\frac{(x_i - x_{i-1})}{2}\widetilde{\mathbf{G}}^L(x_{i-1}, x_i, \mathbf{U}^n_{i-1}, \mathbf{U}^n_i) + \frac{(x_{i+1} - x_i)}{2}\widetilde{\mathbf{G}}^R(x_i, x_{i+1}, \mathbf{U}^n_i, \mathbf{U}^n_{i+1})\right] \tag{5.11}$$

where $\widetilde{\mathbf{G}}^L$ and $\widetilde{\mathbf{G}}^R$ are continuous functions, giving the averages of $\mathbf{G}(x, \mathbf{U}^n)$ on the subcells $[x_{i-\frac{1}{2}}, x_i]$, $[x_i, x_{i+\frac{1}{2}}]$ respectively. These functions give the upwind values of the source terms and following [6] can be defined as

$$\widetilde{\mathbf{G}}^L_{i-\frac{1}{2}} = \left[\mathbf{I} + |\mathbf{J}_{i-\frac{1}{2}}|\mathbf{J}^{-1}_{i-\frac{1}{2}}\right]\widetilde{\mathbf{G}}_{i-\frac{1}{2}}, \quad \widetilde{\mathbf{G}}^R_{i+\frac{1}{2}} = \left[\mathbf{I} - |\mathbf{J}_{i+\frac{1}{2}}|\mathbf{J}^{-1}_{i+\frac{1}{2}}\right]\widetilde{\mathbf{G}}_{i+\frac{1}{2}}. \tag{5.12}$$

The definition of the $\mathbf{J}_{i\pm\frac{1}{2}}$ and $|\mathbf{J}_{i\pm\frac{1}{2}}|$ matrices follow from equation (5.6) for the Roe linearization and are defined in the Appendix. The terms $\widetilde{\mathbf{G}}_{i\pm\frac{1}{2}}$ represent approximations of $\mathbf{G}$ at cells $(i-1,i)$ and $(i,i+1)$ respectively.

Alternatively, we re-write (5.12) in decomposed form

$$\widetilde{\mathbf{G}}^L_{i-\frac{1}{2}} = \sum_{k=1}^{M} \beta^k_{i-\frac{1}{2}} \tilde{\mathbf{e}}^k_{i-\frac{1}{2}} \left(1 + \mathrm{sgn}(\tilde{a}^k_{i-\frac{1}{2}})\right), \quad \widetilde{\mathbf{G}}^R_{i+\frac{1}{2}} = \sum_{k=1}^{M} \beta^k_{i+\frac{1}{2}} \tilde{\mathbf{e}}^k_{i+\frac{1}{2}} \left(1 - \mathrm{sgn}(\tilde{a}^k_{i+\frac{1}{2}})\right) \tag{5.13}$$

where the values of $\beta^k_{i\pm\frac{1}{2}}$ are determined from

$$\sum_{k=1}^{M} \beta^k_{i\pm\frac{1}{2}} \tilde{\mathbf{e}}^k_{i\pm\frac{1}{2}} = \widetilde{\mathbf{G}}_{i\pm\frac{1}{2}}. \tag{5.14}$$

5.2 The Lax-Wendroff scheme

The numerical flux for the classical second order Lax-Wendroff scheme for nonlinear systems, on non-uniform grids, can be written as

$$\mathbf{F}^{LW}_{i+\frac{1}{2}} = \mathbf{F}^{ROE}_{i+\frac{1}{2}} + \frac{1}{2}\sum_{k=1}^{M} \left(\frac{\Delta x_i}{\Delta x_{i+\frac{1}{2}}} - \frac{\Delta t}{\Delta x_{i+\frac{1}{2}}} |\tilde{a}^k_{i+\frac{1}{2}}| \right) |\tilde{a}^k_{i+\frac{1}{2}}| \alpha^k_{i+\frac{1}{2}} \tilde{\mathbf{e}}^k_{i+\frac{1}{2}} \tag{5.15}$$

where $\Delta x_{i+\frac{1}{2}} = (\Delta x_{i+1} + \Delta x_i)/2$.

A flux-limited second order TVD version of the LW scheme is then given by

$$\mathbf{F}^{LWTVD}_{i+\frac{1}{2}} = \mathbf{F}^{ROE}_{i+\frac{1}{2}} + \frac{1}{2}\sum_{k=1}^{M} \left(\frac{\Delta x_i}{\Delta x_{i+\frac{1}{2}}} - \frac{\Delta t}{\Delta x_{i+\frac{1}{2}}} |\tilde{a}^k_{i+\frac{1}{2}}| \right) \Phi(\theta^k_{i+\frac{1}{2}}) |\tilde{a}^k_{i+\frac{1}{2}}| \alpha^k_{i+\frac{1}{2}} \tilde{\mathbf{e}}^k_{i+\frac{1}{2}} \tag{5.16}$$

where

$$\theta^k_{i+\frac{1}{2}} = \frac{\alpha^k_{I+\frac{1}{2}}}{\alpha^k_{i+\frac{1}{2}}} \qquad \text{with} \qquad I = \begin{cases} i-1, & \text{if} \quad \tilde{a}^k_{i+\frac{1}{2}} > 0 \\ i+1, & \text{if} \quad \tilde{a}^k_{i+\frac{1}{2}} < 0, \end{cases} \tag{5.17}$$

where Φ can be any of the limiter function presented in Section 3.

The Roe averaged values can be used for the source term approximations and obtain a flux-limited second order approximation of the source term in the numerical source term (5.10) by setting

$$\begin{aligned}
\widetilde{\mathbf{G}}^L_{i-\frac{1}{2}} &= \sum_{k=1}^{M} \beta^k_{i-\frac{1}{2}} \tilde{\mathbf{e}}^k_{i-\frac{1}{2}} \left[1 + \mathrm{sgn}(\tilde{a}^k_{i-\frac{1}{2}}) \left(1 - \Phi(\theta^k_{i-\frac{1}{2}}) \left(\frac{\Delta x_i}{\Delta x_{i-\frac{1}{2}}} - \frac{\Delta t}{\Delta x_{i-\frac{1}{2}}} |\tilde{a}^k_{i-\frac{1}{2}}| \right) \right) \right], \\
\widetilde{\mathbf{G}}^R_{i+\frac{1}{2}} &= \sum_{k=1}^{M} \beta^k_{i+\frac{1}{2}} \tilde{\mathbf{e}}^k_{i+\frac{1}{2}} \left[1 - \mathrm{sgn}(\tilde{a}^k_{i+\frac{1}{2}}) \left(1 - \Phi(\theta^k_{i+\frac{1}{2}}) \left(\frac{\Delta x_i}{\Delta x_{i+\frac{1}{2}}} - \frac{\Delta t}{\Delta x_{i+\frac{1}{2}}} |\tilde{a}^k_{i+\frac{1}{2}}| \right) \right) \right].
\end{aligned}$$

5.3 The MacCormack scheme

The MacCormack scheme adapted to approximate systems of conservation laws with a source term present as

$$\mathbf{U}_i^{n+1} = \frac{1}{2}\left(\mathbf{U}_i^n + \mathbf{U}_i^{(1)}\right) - \frac{\Delta t}{2\Delta x_i}\left(\mathbf{F}_i^{(1)} - \mathbf{F}_{i-1}^{(1)}\right) + \frac{\Delta t}{2\Delta x_i}\widetilde{\mathbf{G}}_{i-\frac{1}{2}}^{(1)} \tag{5.18}$$

where

$$\mathbf{U}_i^{(1)} = \mathbf{U}_i^n - \frac{\Delta t}{\Delta x_i}\left(\mathbf{F}_{i+1}^n - \mathbf{F}_i^n\right) + \frac{\Delta t}{\Delta x_i}\widetilde{\mathbf{G}}_{i+\frac{1}{2}}^n.$$

This scheme is a predictor-corrector scheme and has the advantage that we do not need to approximate the eigenvalues and eigenvectors of the Jacobian matrix. However, spurious oscillations may occur in the solution, even when we choose the $\widetilde{\mathbf{G}}$ discritisations in (5.18) in such a way as to satisfy (5.4).

6 The shallow water model

We consider the well known one-dimensional shallow water (SW) system, with a geometrical source term (the bottom topography) added, written in differential conservation law form as a single vector equation

$$\mathbf{U}_t + \mathbf{F}(\mathbf{U})_x = \mathbf{G}(\mathbf{U}), \tag{6.1}$$

with

$$\mathbf{U} = \begin{bmatrix} h \\ hu \end{bmatrix}, \qquad \mathbf{F}(\mathbf{U}) = \begin{bmatrix} hu \\ hu^2 + \frac{g}{2}h^2 \end{bmatrix}, \qquad \mathbf{G}(\mathbf{U}) = \begin{bmatrix} 0 \\ -ghZ' \end{bmatrix}.$$

System (6.1) describes the flow at time $t \geq 0$ at point $x \in \mathbb{R}$, where $h(x,t) \geq 0$ is the total water height above the bottom, $u(x,t)$ is the average horizontal velocity, $Z(x)$ is the bottom height function and g the gravitational acceleration. In the following we will denote as $q = hu$ the water unit discharge.

An important property of system (6.1) is related to the source term: the shallow water system admits non-trivial steady states. They are characterized by

$$(hu)_x = 0, \tag{6.2}$$

$$(hu^2 + \frac{g}{2}h^2)_x = ghZ', \tag{6.3}$$

i.e,

$$q = (hu) = \text{constant}, \tag{6.4}$$

$$hu^2 + g(h+Z) = \text{constant}. \tag{6.5}$$

A particular case that provides a benchmark for many approximating schemes is the still water steady state (flow at rest), i.e, when $u = 0$ and $h + Z =$ constant.

In [6] the concept of Property C was introduced. A given scheme would satisfy Property C if in the case of a flow at rest, there is an exact balance between the discrete components

of the flux and a given source term treatment based on (5.4), approximating this stationary solution. Thus, when numerically treating the source terms, for the Property C to be satisfied exactly or approximately (to order $O(\Delta x^2)$) one must ensure that this equilibrium solution would not be perturbed. The source term discretizations presented in the previous sections along with these presented in the Appendix for the shallow water system satisfy this property.

In the numerical examples presented next we use again a variable time step as to satisfy the $CFL \leq 1$ stability condition, as,

$$CFL = \Delta t \max_{i,k} \left\{ \left| \frac{\tilde{a}^{k+}_{i-\frac{1}{2}}}{\Delta x_i} \right|, \left| \frac{\tilde{a}^{k-}_{i+\frac{1}{2}}}{\Delta x_i} \right| \right\}, \tag{6.6}$$

where for the shallow water model the values $\tilde{a}^{k+}_{i-\frac{1}{2}}$ and $\tilde{a}^{k-}_{i+\frac{1}{2}}$ are defined in the Appendix.

6.1 Idealized Dam-Break Flow

We first consider a non-stationary case, the dam break problem in a rectangular channel with flat bottom, $Z = 0$. We computed the solution on a channel of length $L = 2000m$ for time $t = 50s$ and with initial conditions:

$$u(x,0) = 0,$$
$$h(x,0) = \begin{cases} h_1 & x \leq 1000, \\ h_0 & x > 1000, \end{cases}$$

with $h_1 > h_0$. This is the corresponding Riemann problem for the homogeneous problem. The water depth ratio is given by h_0/h_1. The dam collapses at $t = 0s$ and the resulting flow consists of a shock wave (bore) traveling downstream and a rarefaction wave (depression wave) traveling upstream. The upstream depth h_1 was set at 10m. When the depth ratio is greater than 0.5, the flow throughout the channel remains subcritical. For depth ratios smaller than 0.5, the flow downstream of the dam position is supercritical while remaining subcritical upstream. For very small values of the ratio h_0/h_1 the flow regime becomes strongly supercritical downstream and the shock wave can be difficult to capture, see [9] for example.

Comparative results are presented in a 101 grid points and depth ratio 0.005, for all schemes in Figs. 14-16. All the results produced with the adaptive method are significantly better in terms of shock resolution and non-oscillatory behavior in both components of the solution. No entropy problems appeared for all adaptive schemes. It can also be seen the grid clustering and the close following of the downstream shocks. It is impressive that the LW and MacCormack schemes are able to produce quality results with no extra dissipative mechanism imposed other than the adaptive one. The performance of the TVD scheme was also improved.

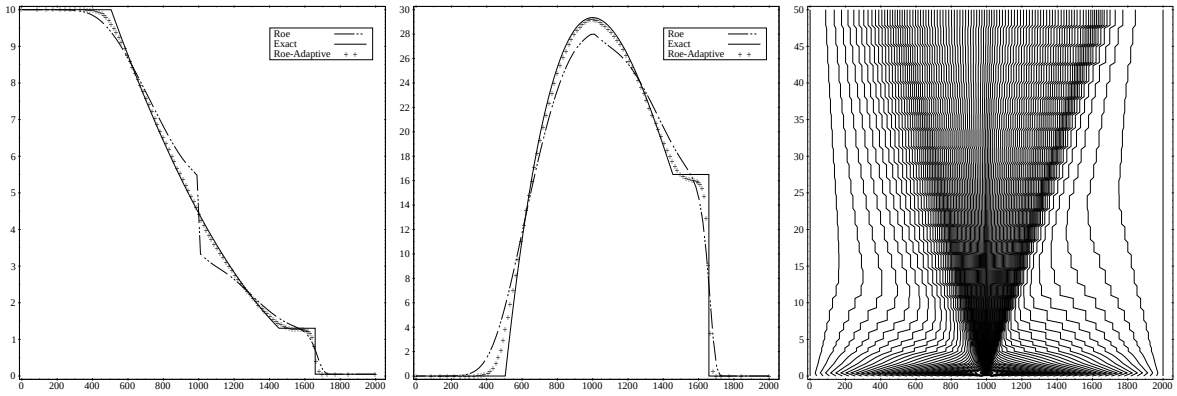

Figure 14: Idealized Dam-Break flow: Numerical results with Roe's scheme for depth (h) (left), discharge (q) (middle) and grid point trajectories (left) for the adaptive Roe scheme ($p = 0.0518, D = 0.0065$).

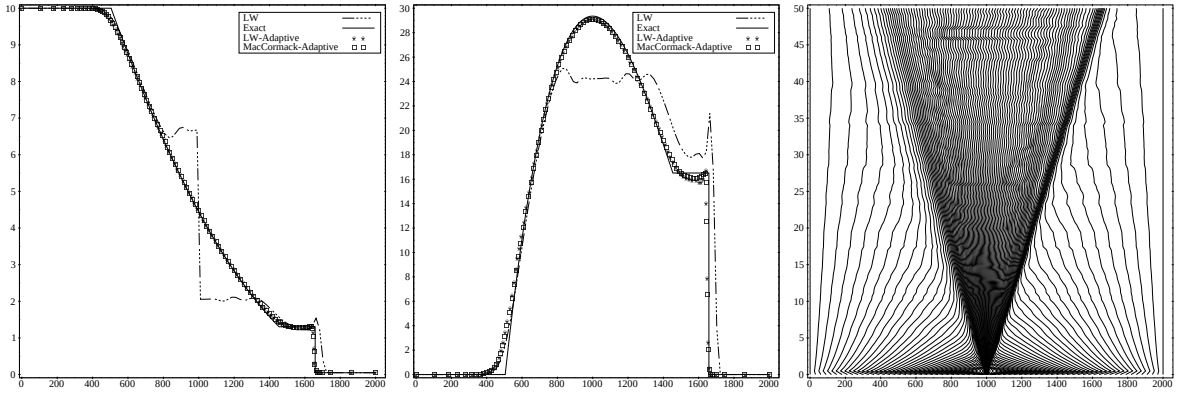

Figure 15: Idealized Dam-Break flow: Numerical results with the LW and MacCormack schemes for depth (h) (left), discharge (q) (middle) and grid point trajectories (left) for the adaptive LW scheme ($p = 0.10, D = 0$).

6.2 Flow at rest over topography

We consider system (6.1) with initial conditions

$$u(x,0) = 0, \quad \forall x \in \mathbb{R},$$
$$h(x,0) + Z(x) = H, \quad \forall x \in \mathbb{R},$$

then clearly

$$u(x,t) = 0, \quad \forall x \in \mathbb{R}, t \geq 0,$$
$$h(x,t) + Z(x) = H, \quad \forall x \in \mathbb{R}, t \geq 0,$$

is a solution to (6.1).

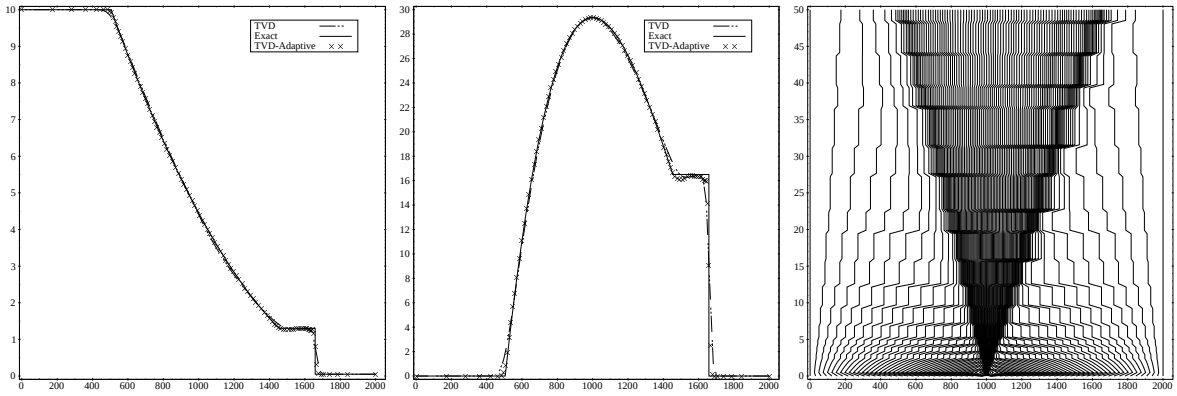

Figure 16: Idealized Dam-Break flow: Numerical results with the TVD scheme for depth (h) (left), discharge (q) (middle) and grid point trajectories (left) for the adaptive TVD scheme ($p = 0.10, D = 0.0125$).

We test the schemes with the adaptive mechanism, in order to study its behavior, to this benchmark problem, [8, 10], with $Z(x)$ given by

$$Z(x) = \begin{cases} 0.2 - 0.05(x-10)^2, & 8 \leq x \leq 12, \\ 0, & \text{otherwise}, \end{cases} \tag{6.7}$$

in channel of length $L = 25m$ and $H = 2m$. Figs. 17 and 18 display the final water level and the final unit discharge values respectively, for the Roe and MacCormack schemes at time $t = 200s$ and in grid of 101 points. Both schemes with the adaptation method perfectly preserve this steady state, up to machine accuracy. Grid points are concentrated were most needed, in the topography area. The movement of the grid points is stopping much faster for the MacCormack scheme being a second order method. Similar results and observations were made for the TVD scheme and are not presented here for brevity.

6.3 Steady transcritical flow with shock

The initial conditions in this case was taken to be as in Section 6.2 with H being the constant water level downstream provided by the boundary condition. In this test case the upstream boundary condition for the discharge was $q = 0.18m^2/s$ and the downstream boundary condition for the water level was $H = 0.33m$.

Results for coarse 51 point grid for the Roe scheme and on a 101 point grid TVD scheme with the VL limiter used are shown in Figs.19 and 20. The improvement in the results compared to the uniform grid results is clear. The shock resolution in very good and the steady state is perfectly preserved for a long time calculation of $t = 800s$. Finally in Fig. 21 the numerical result for the depth in shown for the adaptive MacCormack scheme, where in spite the formation of a strong shock the scheme can produce a non-oscillatory steady state solution. The CFL value used in all test was 0.8.

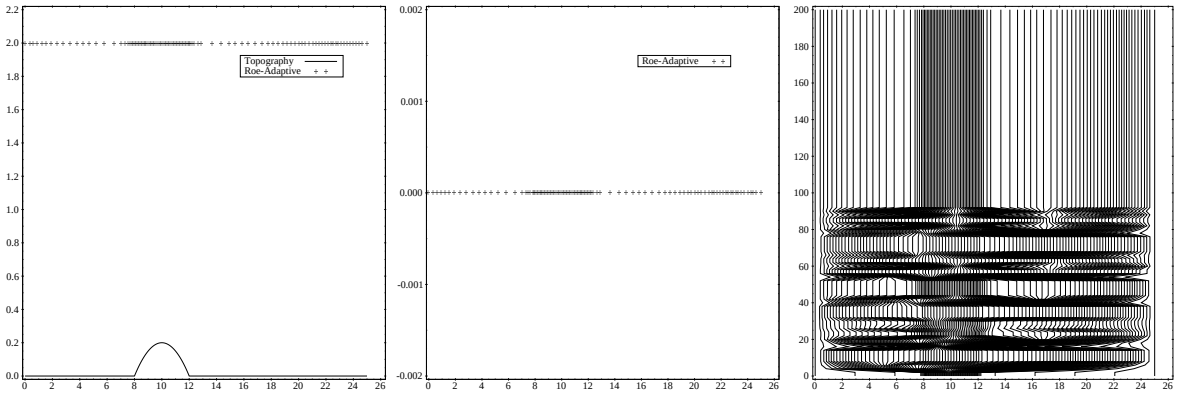

Figure 17: Flow at rest over topography: Numerical results with Roe's scheme for depth (h) (left), discharge (q) (middle) and grid point trajectories (left) for the adaptive Roe scheme ($p = 0.06, D = 0.0525$).

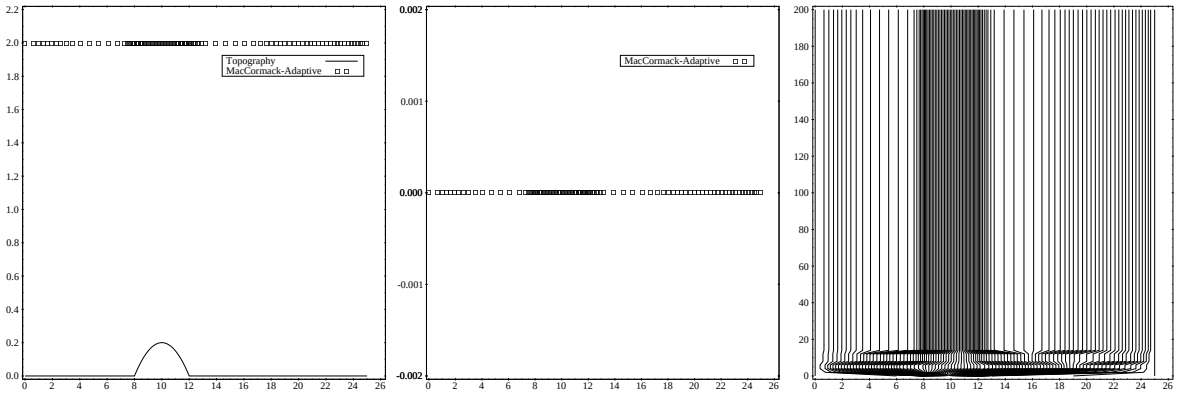

Figure 18: Flow at rest over topography:Numerical results with the MacCormack scheme for depth (h) (left), discharge (q) (middle) and grid point trajectories (left) for the adaptive MacCormack scheme ($p = 0.06, D = 0.0525$).

6.4 Dam-Break flow over topography

In this example we solve the SW equations with a wavy bottom $Z(x)$

$$Z(x) = \begin{cases} 0.3(\cos(\pi(x-1)/2))^{30}, & |x-1| \leq 1 \\ 0, & \text{else.} \end{cases} \tag{6.8}$$

and initial conditions

$$h(x,0) = \begin{cases} 2.0 - Z(x), & -10 \leq x < 1 \\ .35 - Z(x), & 1 \leq x < 10 \end{cases}, \qquad u(x,0) = \begin{cases} 1, & -10 \leq x < 1 \\ 0, & 1 \leq x < 10, \end{cases} \tag{6.9}$$

in order to test the adaptive behavior of the schemes in an unsteady flow over topography, this test was also presented in [23]. The solution profiles for the water depth and velocity profile

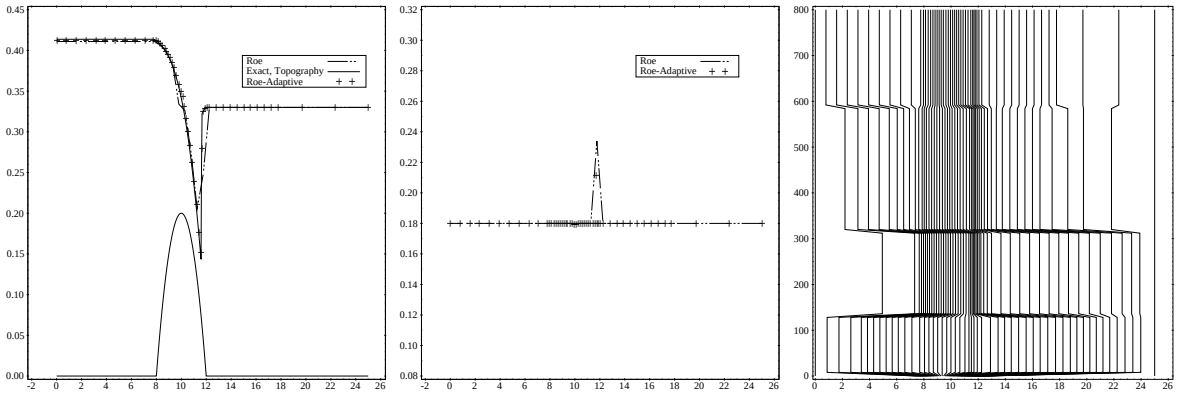

Figure 19: Steady transcritical flow with shock: Numerical results with Roe's scheme for depth (h) (left), discharge (q) (middle) and grid point trajectories (left) for the adaptive Roe scheme ($p = 0.08, D = 0.07$).

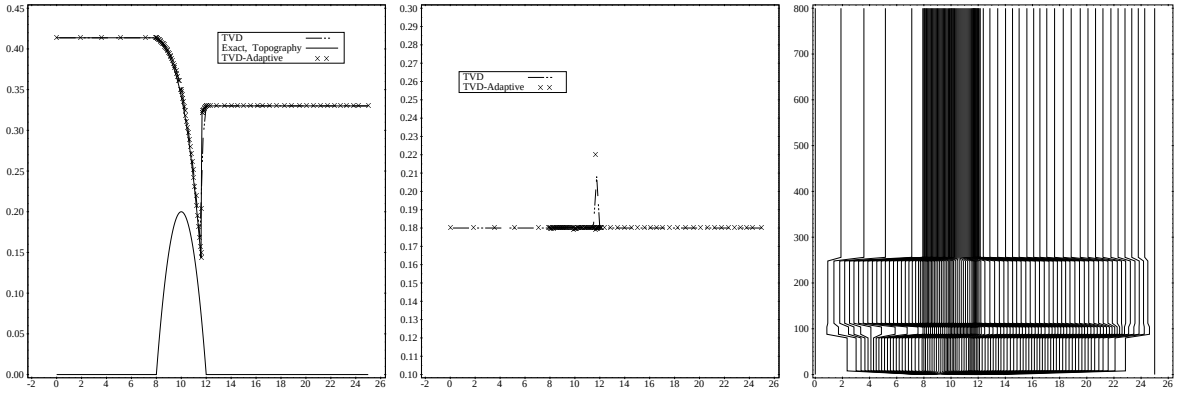

Figure 20: Steady transcritical flow with shock: Numerical results with the TVD scheme for depth (h) (left), discharge (q) (middle) and grid point trajectories (left) for the adaptive TVD scheme ($p = 0.08, D = 0.075$).

at time $t = 1s$ are shown in Figs 22-24. A grid of 101 points was used in all calculations. The exact solution for this problem is a reference solution calculated in a uniform grid with 4001 grid points. The improvements in the numerical solution can be observed for all the schemes, even the MacCormack scheme can produce a stable solution when compared with the one calculated in a uniform grid.

7 Conclusions

In this work we have rigorously investigated, in a general framework, the numerical behavior of an adaptive grid redistribution **AGR** mechanism, when combined with classical numerical methods, with well known characteristics, for solving one dimensional hyperbolic conservation

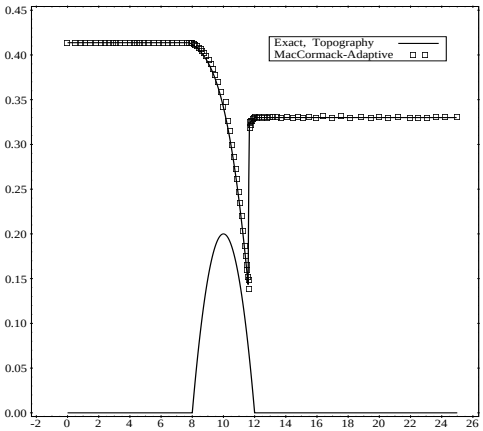

Figure 21: Steady transcritical flow with shock: Numerical result for depth (h) with the adaptive MacCormack scheme ($p = 0.12, D = 0$).

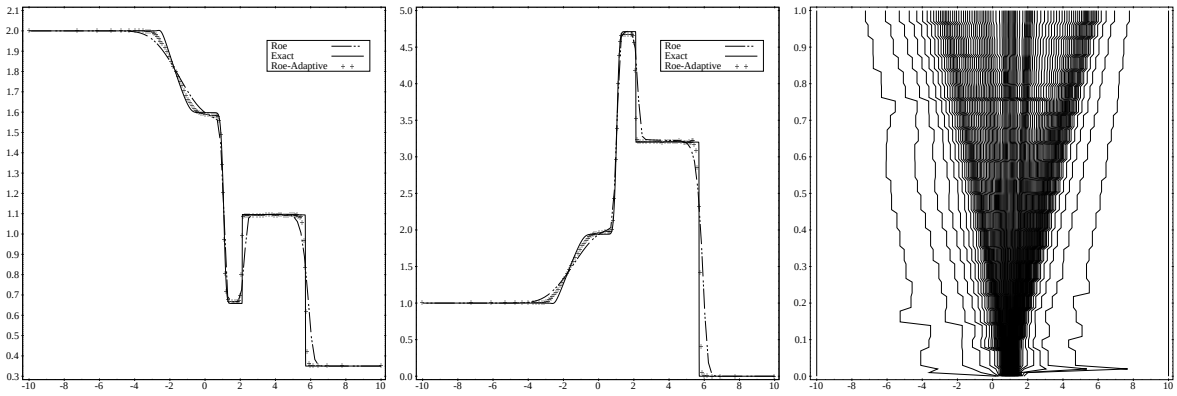

Figure 22: Dam-Break flow over topography:Numerical results with Roe's scheme for depth (h) (left), velocity (u) (middle) and grid point trajectories (right) for the adaptive Roe scheme ($p = 0.1093, D = 0.0055$).

laws. The evolving mesh is constructed such that it's spatial resolution is controlled via selective geometrical characteristics of the numerical solution, by choosing some power of the curvature of the solution as the estimator function. A conservative reconstruction procedure for the numerical solution is also applied at each evolution step.

From the numerical experiments produced in the course of this work we concluded to the following advantages of the use of this particular **AGR** method. The method, when applied to classical second order schemes that compute oscillatory solutions, suppress the oscillations produced, and as such, classical schemes like the Lax-Wendroff or the Mac-Cormack scheme become stable and can produce reliable solutions. When applied to numerical schemes that do not satisfy entropy conditions the scheme can converge to the entropy satisfying solution. The method performs well with hyperbolic problems with source terms by producing stable

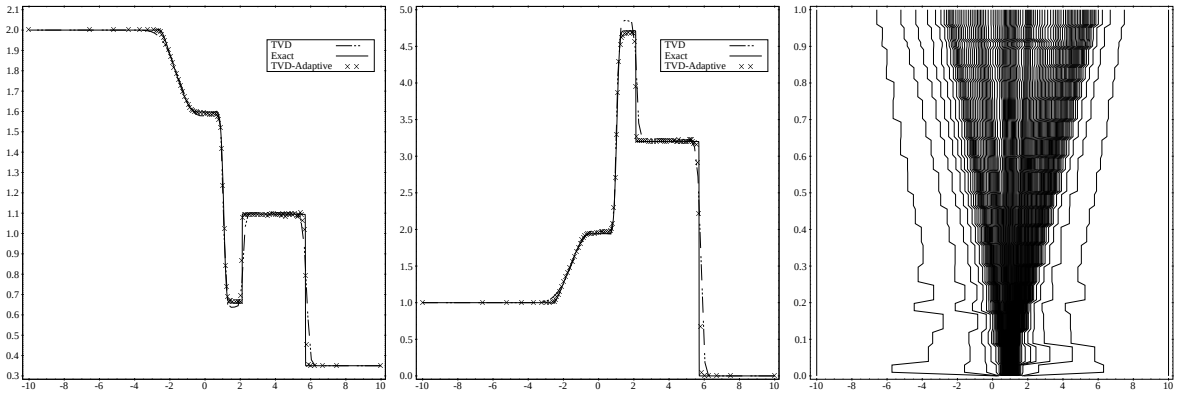

Figure 23: Dam-Break flow over topography: Numerical results with the TVD scheme for depth (h) (left), velocity (u) (middle) and grid point trajectories (right) for the adaptive TVD scheme ($p = 0.1126, D = 0.006$).

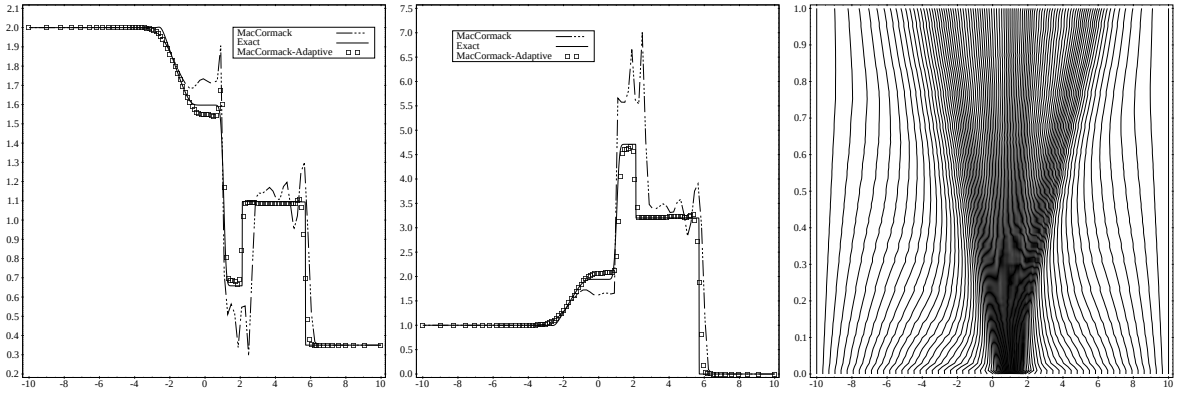

Figure 24: Dam-Break flow over topography:Numerical results with the MacCormack scheme for depth (h) (left), velocity (u) (middle) and grid point trajectories (right) for the adaptive MacCormack scheme ($p = 0.0345, D = 0.001$).

solutions for first and second order balanced schemes that converge to the correct steady states. The method can automatically detect, resolve and track steep wave fronts and discontinuities, without having to resort to finer grids. It can be combined with high resolution flux or slope limiting TVD scheme and improve the numerical solutions obtained. Only two computational parameter have to be tuned for each problem, the power of the estimator function and the tolerance that measures the relative mean displaysmend between two consecutively grid distributions, under which the reconstruction step is avoided. The mechanism has been proved robust for all the applications presented and schemes used.

There is a need for further theoretical work and development of the method, as well as extension of the ideas presented in this work in higher dimensions. These will provide a more in depth understanding of the method's dehavior, the advantages gained when combine

the adaptive mechanism with existing numerical schemes, and more general, the influence of evolving adaptive grids to the numerical solution of dynamic PDEs.

Acknowledgments

The authors wish to thank Prof. Ch. Makridakis for his helpful discussions during the course of this work. Ch.A. was partially supported by the European Union RTN-network HYKE, HPRN-CT-2002-00282 and the program Pythagoras of EPEAEK II.

References

[1] Arvanitis Ch., Katsaounis Th., and Makridakis Ch. Adaptive finite element relaxation schemes for hyperbolic conservations laws. *M2AN* 2001; **35**:17–33.

[2] Arvanitis Ch., Makridakis Ch. and Tzavaras A Stability and convergence of a class of finite element schemes for hyperbolic systems of conservation laws *SIAM J. Numer. Anal.* 2004; **42**: 1357–1393.

[3] Arvanitis Ch. Mesh Redistribution Strategies and Finite Element Schemes for Hyperbolic Conservation Laws, (submitted).

[4] Azarenok BN, Ivanenko SA, and Tang T. Adaptive mesh redistribution method based on Godunov's scheme. *Comm. Math. Sci.* 2003; **1**:152–179.

[5] Beckett G., Mackenzie JA. Convergence analysis of finite difference approximations on equidistributed grids to a singularly perturbed boundary value problem. *Applied Numerical Mathematics* 2000; **35**:87–109.

[6] Bermudez A, Vázquez ME, Upwind methods for hyperbolic conservation laws with source terms. *Computers and Fluids* 1994; **23**:1049.

[7] Chen L, Sun P, and Xu J. Optimal Anisotropic meshes for minimizing interpolation errors in L^p-Norm. *Mathematics of of Computation* 2004;

[8] Delis AI. Improved application of the HLLE Riemann solver for the shallow water equations with source terms. *Comm. Numer. Meth. Engin.* 2003; **19**:59–83.

[9] Delis AI, Skeels CP. TVD schemes for open channel flow. *Int. J. Numer. Meth. Fluids* 1998; **26**:791–809.

[10] Delis AI, Katsaounis Th. Relaxation schemes for the shallow water equations. *Int. J. Numerical Methods in Fluids*, 2003; **41**: 695–719.

[11] Harten S, Hyman JM. Self Adjusting Grid Methods for One-Dimensional Hyperbolic Conservation Laws. *Journal of Computational Physics* 1983; **50**:17–33.

[12] Hirch C. Numerical Computation of Internal and External Flows, Wiley, 1988.

[13] Hubbard ME, Garcia-Navarro P. Flux Difference Splitting and the Balancing of Source terms and Flux Gradients. *J. Comp. Phys.* 2000; **165**:89–125.

[14] Hyman JM, Li S, and Linda Petzold L. An Adaptive Moving Mesh Method with Static Rezoning for Partial Differential Equations.

[15] LeVêque RJ. Finite Volume Methods for Hyperbolic Problems, Cambridge University Press, 2002.

[16] Li R, Tang T, and Zhang P. Moving Mesh Methods in Multiple Dimensions Based on Harmonic Maps *Journal of Computational Physics* 1983; **170**:562–588.

[17] Li S, Petzold L. Moving Mesh methods with upwind schemes for time dependent PDEs. *J. Comp. Phys.* 1997; **131**:368–377.

[18] Fazio R, Leveque RJ. Moving-Mesh methods for One-Dimensional Hyperbolic Problems Using CLAWPACK *Computers and Mathematics with Applications* 2003; **45**:273–298.

[19] Pennington SV, Berzins M. New NAG Software for first order Partial Differential Equations. *ACM Transactions on Mathematical Software* 1994; **20**:63-99.

[20] Li S, Petzold L, and Ren Y. Stability of Moving Mesh systems of Partial Differential Equations. *SIAM J. Sci. Comput.* 1998; **20**:719-738.

[21] Stockie JM, Mackenzie, JA and Russel, RD. A Moving mesh method for one-dimensional hyperbolic conservation laws. *SIAM J. Sci. Comput.* 2001; **22**:1791-1813.

[22] Tan Z, Zhang Z, Huang Y, Tang T. Moving mesh methods with locally varying time steps. *Journal of Computational Physics* 2004; **35**:17–33.

[23] Tang H. Solution of the shallow-water equations using an adaptive moving mesh method *Int. J. Numer. Meth. Fluids* 2004; **44**:789-810.

[24] Tang H, Tang T. Adaptive mesh methods for one- and two-dimensional hyperbolic conservation laws. *SIAM J. Numer. Anal.* 2003; **41**: 487–515.

APPENDIX

Following [6] and [8] the average velocity and celerity, $\tilde{u}_{i+\frac{1}{2}}$ and $\tilde{c}_{i+\frac{1}{2}}$, for the eigenvalues $\tilde{a}^{1,2}_{i+\frac{1}{2}} = \tilde{u}_{i+\frac{1}{2}} \pm \tilde{c}_{i+\frac{1}{2}}$ and eigenvectors, and $\tilde{\mathbf{e}}^{1,2}_{i+\frac{1}{2}} = \left(1, \tilde{a}^{1,2}_{i+\frac{1}{2}}\right)^{\mathrm{T}}$ of Roe's linearised Jacobian $\mathbf{J}_{i+\frac{1}{2}}$ for the SW system, are calculated as

$$\tilde{u}_{i+\frac{1}{2}} = \frac{u_{i+1}\sqrt{h_{i+1}} + u_i\sqrt{h_i}}{\sqrt{h_{i+1}} + \sqrt{h_i}}, \qquad \tilde{c}_{i+\frac{1}{2}} = \sqrt{g\tilde{h}}, \tag{7.1}$$

with $\tilde{h} = (h_{i+1} + h_i)/2$.

The wave strengths $\alpha^{1,2}_{i+\frac{1}{2}}$ follow solving equation (5.7) and they are:

$$\alpha^{1,2}_{i+\frac{1}{2}} = \frac{1}{2}\Delta_{i+\frac{1}{2}}h \pm \frac{1}{2\tilde{c}_{i+\frac{1}{2}}}(\Delta_{i+\frac{1}{2}}(uh) - \tilde{u}_{i+\frac{1}{2}}\Delta_{i+\frac{1}{2}}h) \tag{7.2}$$

where the operator $\Delta_{i+\frac{1}{2}}(\cdot) = (\cdot)_{i+1} - (\cdot)_i$.

For a general system of equations and temporarily droping the subscript indeces we define $\mathbf{\Lambda}$ to be the diagonal form of $\mathbf{J}$ given by its eigenvalues, with

$$\mathbf{J} = \mathbf{R}\mathbf{\Lambda}\mathbf{R}^{-1}, \tag{7.3}$$

where $\mathbf{R}$ is the matrix of the eigenvectors of $\mathbf{J}$. We can now define the $|\mathbf{\Lambda}|$, $\mathbf{\Lambda}^+$ and $\mathbf{\Lambda}^-$ matrices as

$$|\mathbf{\Lambda}| = \mathrm{diag}(|a^1|, \ldots, |a^M|), \quad \mathbf{\Lambda}^{\pm} = \mathrm{diag}(a^{1^{\pm}}, \ldots, a^{M^{\pm}}), \tag{7.4}$$

where $a^{k^+} = \max(0, a^k)$ and $a^{k^-} = \min(0, a^k)$. We can now define

$$|\mathbf{J}| = \mathbf{R}|\mathbf{\Lambda}|\mathbf{R}^{-1}, \tag{7.5}$$

$$\mathbf{J}^{\pm} = \mathbf{R}\mathbf{\Lambda}^{\pm}\mathbf{R}^{-1}. \tag{7.6}$$

Therefore,

$$|\mathbf{J}| = \mathbf{J}^+ - \mathbf{J}^-, \quad \mathbf{J}^{\pm} = \frac{1}{2}(\mathbf{J} \pm |\mathbf{J}|). \tag{7.7}$$

Now in the discrete level the $\mathbf{J}_{i+\frac{1}{2}}$ and $|\mathbf{J}_{i+\frac{1}{2}}|$ are defined as

$$\mathbf{J}_{i+\frac{1}{2}} = \mathbf{R}_{i+\frac{1}{2}}\mathbf{\Lambda}_{i+\frac{1}{2}}\mathbf{R}^{-1}_{i+\frac{1}{2}}, \tag{7.8}$$

$$|\mathbf{J}_{i+\frac{1}{2}}| = \mathbf{R}_{i+\frac{1}{2}}|\mathbf{\Lambda}_{i+\frac{1}{2}}|\mathbf{R}^{-1}_{i+\frac{1}{2}}. \tag{7.9}$$

Using the above formulation to calculate te upwind approximation of the source term $\mathbf{G}$, we need to evaluate $\widetilde{\mathbf{G}}_{i\pm\frac{1}{2}}$ approximations, that is some average of the source term on the left and right of the cell interfaces. Then a well balanced approximation of the source term, at each computational cell, is

$$\widetilde{\mathbf{G}}_{i\pm\frac{1}{2}} = \left[0, -\frac{g}{2}(h_{i+1} + h_i)(Z_{i+1} - Z_i)\right]^{\mathrm{T}}. \tag{7.10}$$

The values of $\beta^{1,2}_{i\pm\frac{1}{2}}$ in (5.14) are the components of $\mathbf{R}^{-1}_{i+\frac{1}{2}}\widetilde{\mathbf{G}}_{i\pm\frac{1}{2}}$, and for the shallow water system are

$$\beta^{1,2}_{i\pm\frac{1}{2}} = \mp\frac{\tilde{c}_{i+\frac{1}{2}}(Z_{i+1} - Z_i)}{2}.$$