

Mesh Redistribution Strategies and Finite Element Schemes for Hyperbolic Conservation Laws

Christos Arvanitis *

Abstract

In this work we consider a new class of Relaxation Finite Element schemes for Conservation Laws, with more stable behavior on the limit area of the relaxation parameter. Combine this scheme with an efficient adapted spatial redistribution process, considered also in this work, we form a robust scheme of controllable resolution. The results on a number of test problems show that this scheme can produce entropic-approximations of high resolution even on the limit of the relaxation parameters. Since on the limit the scheme lack of the relaxation mechanism, we experimentally conclude that the proposed spatial redistribution can be a stabilization mechanism by its own for computational solutions of CL problems.

Keywords: Finite Element Methods, Relaxation Model, Adaptive Grid Redistribution, Hyperbolic Conservation Laws.

1 Introduction

In this paper we consider finite element schemes and adaptive strategies for the approximation of nonlinear hyperbolic systems of conservation laws, *find* $u: \mathbb{R}^d \times [0, T] \rightarrow \mathbb{R}^M$ *such that*

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

Finite element was not a very popular choice for computing solutions of (1.1). When applied directly to this system will result approximate solutions with oscillatory character close to the shock. This well known phenomenon is related to the fact that direct finite element discretization of (1.1) behave as dispersive approximation. The study of dispersive schemes for approximating conservation laws was an interesting issue since the works of Von Newman, see the very interesting work by Lax and Hou [17] and its references.

To overcome this difficulty several modifications of the standard finite element schemes have been proposed in the literature. Two main classes of these methods are the streamline diffusion type methods, and the Discontinuous Galerkin - Runge Kutta methods [21, 11, 19, 10]. In both cases the necessary stability and viscosity mechanisms are imposed by hand. Streamline diffusion methods include artificial viscosity and complicated shock capturing terms and Discontinuous Galerkin methods are stabilized by upwind discrete fluxes across the interfaces of elements and mainly by additional limiters. Recently a new class of finite element methods introduced in [2, 3]. These methods are based on relaxation models, they do not include additional stabilization terms but alternatively are designed to be used in conjunction with appropriate mesh refinement. Mesh adaptation is a main current stream towards to the efficient numerical solution of complex systems. Thus in our case of Hyperbolic Systems, the finite elements are still a natural choice since the development of supportive structures (finite element spaces of any order, flexibility in mesh construction, e.t.c.) in adaptive finite element literature and software implementation is at a remarkable level.

Our aim in this work is

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

- To propose and experimentally study new finite element schemes for (1.1).
- To introduce adaptive strategies for shock computations.
- To conclude new and rather unexpected observations for the behavior of dispersive-type finite element schemes under the regime of adaptive evolving mesh.

Our schemes are based on the idea of Relaxation Finite Element Schemes of [2] but are further developed to result an extremely robust class of finite element methods for Hyperbolic Systems. The idea of the schemes in [2, 3] was to use the simple and very appropriate for computational purposes model of Jin and Xin [20] to construct finite element schemes without additional stabilization mechanisms. In this work we suggest a modification of these schemes that computationally decouple completely the effect of relaxation parameter and the mesh sizes. The resulting scheme, called *Switched Relaxation Finite Element Scheme*, show remarkable stability even for extremely small values of ε . As a natural extension then we consider the limiting case $\varepsilon \rightarrow 0$. The resulted scheme is called *Limit Relaxation Finite Element Scheme*. This scheme although connected to the *Relaxed* difference schemes of Jin and Xin [20] have a major difference. Namely, in the case of Relaxed Schemes the main stabilization mechanism is of "upwinding" type, while in our case the Limit Relaxation Finite Element Scheme *are left with no stabilization mechanism* of viscosity or relaxation type. Still when these schemes are used in conjunction with the adaptive strategies of this paper yield computational solutions with surprisingly stable behavior.

The adaptive strategy studied in this work is based to a uniform distribution of appropriate measures which evolves with time (time steps). At a given time step, the basic principles of the suggested mesh redistribution are:

- a) Locate the regions of space where increased resolution is required, through a positive functional g .
- b) Find a partition of the space with density that follows the estimator function g .
- c) Reconstruct the solution on the FEM space which corresponds to that partition and advance to the next time step by applying the finite element scheme.

These steps are studied in detail in the sequel. Of course a crucial point is the choice of appropriate estimator functions that are used to relocate the nodes of the partition of the space domain. These meshes are called **G-uniform** since the Riemman-Stieltjes measure G induced from the estimator function g , is uniformly distributed on them. We use two different estimator functions that we believe that are appropriate for approximating hyperbolic conservation laws. One is motivated by available a posteriori estimates in the scalar case, [15, 2]. It turns out that a more successful one is based on *a discrete form of the curvature* of the approximate solution already computed at the given time step. The above steps are studied in detail in the one-dimensional with respect to x setting. Having in mind the extension of the algorithms in higher dimensions, we have chosen to present our adaptive strategy in a rather abstract framework and for arbitrary partitions of elements without being specific on the choice of estimators and on important issues related to the implementation of the method in more than one space dimension. Let us note finally that, in view of our extensive computational experiments, it seems that a successful extension of our approach in more than one space dimensions is a challenging task directly related to issues such as anisotropic mesh refinement for conservation laws.

One of the main results of this paper is the rather surprising computational behavior of finite element schemes with very little or none artificial viscosity/relaxation. It turns out that the adaptive mesh selection suggested here is by itself a strong stabilization mechanism for shock computations. Even the direct discretization of the weak formulation of conservation laws by finite elements, that it is known to produce approximate solutions with spurious oscillations, when combined with the mesh adaptation yields stable solutions free of oscillations. Our main conclusions from the series present only few of the numerical experiments are:

- The same schemes with uniform mesh and with mesh redistribution have different *qualitative* behavior: In the cases of Limit Relaxation and of the direct Finite Element discretization to conservation laws,

the schemes on uniform mesh approximate solutions with expected oscillations. The same schemes on adaptive meshes are almost free of oscillations. In cases where the conservation law admits non classical shocks the same schemes while on uniform mesh seem to approximate the non classical shock, when mesh adaptation is used it approximates the entropic solution.

- Switched Relaxation Finite Element schemes is a very robust class of schemes. Like every Finite Element scheme they allow to use high order as well as lower order elements and can be applied on uniform or non-uniform grids without modifications. In addition the relaxation parameter can range all positive values independently from the time step. When this scheme is combined with adaptive spatial grid redistribution, it produces approximations of the entropic solution free of dispersive oscillations.
- In certain cases the oscillatory solutions of finite element schemes without adaptivity show a remarkable stability and seem to converge weakly in the mean sense.

The paper is organized as follows. In the next section we introduce the finite element schemes, on arbitrary partitions. Section 3 is devoted to the mesh redistribution process. In subsection 3.2.3 we describe what are the changes in finite element schemes when combined with our mesh redistribution algorithm.

2 Preliminaries

2.1 Relaxation Finite Element schemes

Relaxation models that approximate (1.1) are the basis of our schemes. In particular, the model suggested in [20]:

Find $u, v_1, \dots, v_d: \mathbb{R}^d \times [0, T] \rightarrow \mathbb{R}^M$ such that:

$$\begin{aligned} u(\cdot, 0) &= u_0 \quad \text{given, } v_i(\cdot, 0) = F_i(u_0), \quad i = 1, \dots, d, \\ \partial_t u + \sum_{i=1}^d \partial_{x_i} v_i &= 0 \\ \partial_t v_i + C_i \cdot \partial_{x_i} u &= -\frac{1}{\varepsilon} (v_i - F_i(u)), \quad i = 1, \dots, d, \end{aligned} \tag{2.1}$$

corresponds to the regularization of each component of the (1.1) by a wave operator of order ε , see [2]. Here the *relaxation characteristics* C_i are symmetric, positive definite matrices with constant coefficients that are selected to satisfy certain stability conditions, *the subcharacteristic conditions*, see [20, 35, 2, 3]. Thus the relaxation model induces a regularization mechanism with *finite speed of propagation* and results a pde with linear principal part. On the other hand the number of unknowns has been increased. Nevertheless, in schemes based on the discretization of (2.1) the extra cost is compensated by the simplicity and the natural implicit explicit discretization that this model admits.

We will assume that the solution u has compact support for all $t \in [0, T]$, i.e. it vanishes outside an (extended) compact set $\Omega_1 \subset \mathbb{R}^d$. Let Ω be a superset of Ω_1 suitable for discretization with $dist(\Omega_1, \Omega) = 1$ and $\mathcal{T} = \{K\}$ is a partition of Ω into elements with the usual properties, [9]. We will use the following notation:

- We will use the standard finite element space $S_{q,\mathcal{T}}$ of q order polynomials, which corresponds to a given admissible partition \mathcal{T} , i.e.

$$S_{q,\mathcal{T}} = \{\phi \in C(\Omega) : \phi|_K \in \mathbb{P}_q, K \in \mathcal{T}, \phi|_{\Omega^c} \equiv 0\},$$

for approximating scalar functions defined on the spatial domain. In the case of vector functions we will use the Cartesian power $S_{q,\mathcal{T}}^M$ of $S_{q,\mathcal{T}}$ spaces, thus, all the coordinates of the numerical solution are approximations given from the same FEM space.

- With $\Pi_{S_{q,\mathcal{T}}}$ we denote the interpolant projection on $S_{q,\mathcal{T}}$, i.e. for any function u defined on the domain, the $\Pi_{S_{q,\mathcal{T}}}(u)$ is an element of the $S_{q,\mathcal{T}}$ space which on some set of fixed points x of the domain interpolates function u (and/or its derivatives), that is $u(x) = \Pi_{S_{q,\mathcal{T}}}(u)(x)$. In the case of linear Finite Elements spaces the $\Pi_{S_{q,\mathcal{T}}^M}(u)$ element is defined by interpolating u on the node points $\{x_i\}_{i=1}^N$ of the partition. For some vector function $u = (u_1, \dots, u_M)^T : \Omega \rightarrow \mathbb{R}^M$ we denote by $\Pi_{S_{q,\mathcal{T}}^M}(u)$ the interpolant element from the Cartesian power space $S_{q,\mathcal{T}}^M$ i.e. $\Pi_{S_{q,\mathcal{T}}^M}(u) = (\Pi_{S_{q,\mathcal{T}}}(u_1), \dots, \Pi_{S_{q,\mathcal{T}}}(u_M))^T$.
- For any function u on Ω we shall notate with $u_{\mathcal{T}}$ the approximation element from the corresponding FEM space to the partition \mathcal{T} .

In this work we consider approximations from FEM spaces $S_{q,\mathcal{T}}$, obtained by Galerkin discretization of (1.1) and (2.1), that is:

- The Galerkin discretization of the Relaxation model of Conservation Law (2.1):

Seek $u_{\mathcal{T}}, v_{\mathcal{T},1}, \dots, v_{\mathcal{T},d}$ in $S_{q,\mathcal{T}}^M \times [0, T]$ such that, $u_{\mathcal{T}}(\cdot, 0) = \Pi_{S_{q,\mathcal{T}}^M}(u_0)$, $v_{\mathcal{T},i}(\cdot, 0) = \Pi_{S_{q,\mathcal{T}}^M}(F_i(u_0))$, $i = 1, \dots, d$ and for $t \in (0, T]$, for all $\phi \in S_{q,\mathcal{T}}^M$

$$\begin{aligned} (\partial_t u_{\mathcal{T}}, \phi) - \sum_{i=1}^d (v_{\mathcal{T},i}, \partial_{x_i} \phi) &= 0, \\ (\partial_t v_{\mathcal{T},i}, \phi) + (C_i \cdot \partial_{x_i} u_{\mathcal{T}}, \phi) &= -\frac{1}{\varepsilon} (v_{\mathcal{T},i} - F_i(u_{\mathcal{T}}), \phi), \quad i = 1, \dots, d. \end{aligned} \quad (2.2)$$

- The direct Galerkin discretization to the Conservation Law (1.1) :

Seek $u_{\mathcal{T}}$ in $S_{q,\mathcal{T}}^M \times [0, T]$ such that, $u_{\mathcal{T}}(\cdot, 0) = \Pi_{S_{q,\mathcal{T}}^M}(u_0)$ and for $t \in (0, T]$, for all $\phi \in S_{q,\mathcal{T}}^M$

$$(\partial_t u_{\mathcal{T}}, \phi) + \sum_{i=1}^d (-F_i(u_{\mathcal{T}}), \partial_{x_i} \phi) = 0. \quad (2.3)$$

2.2 Fully discrete schemes

Our finite element schemes are appropriate fully discrete versions of (2.2) and (2.3). We examine the case of system of conservation laws defined on one-dimensional ($d = 1$) domain $\Omega = [a, b]$. Let κ be the **uniform time step** which we intend to use in our calculations i.e., we interesting in approximating the instances $u(\cdot, t^n)$ of the solution u at moments $t^n = n \cdot \kappa$, $n = 0, 1, \dots$

2.2.1 The Relaxation Finite Element scheme (RFEM)

We start with the fully discrete Relaxation Finite Element (RFEM) scheme, presented in [2]. As it shown there, a choice that results to linear fully discrete scheme and it is well suited to the structure of (2.2), is a combination of an explicit and a diagonal implicit Runge Kutta discretization in time. In order to decouple the system (2.2), we discretize the first equation by using an Explicit RK (ERK) method while for the second we use an Diagonally Implicit RK (DIRK) method.

Assuming that the *Relaxation Galerkin* approximation $(u_{\mathcal{T}}^n, v_{\mathcal{T}}^n)$ of the solutions instance $(u(\cdot, t^n), v(\cdot, t^n))$ of (2.2) is been computed at moment t^n then, the approximation $(u_{\mathcal{T}}^{n+1}, v_{\mathcal{T}}^{n+1})$ of the solution at the next moment t^{n+1} is defined such that for every ϕ in $S_{q,\mathcal{T}}^M$:

$$\begin{aligned} (u_{\mathcal{T}}^{n+1}, \phi) &= (u_{\mathcal{T}}^n, \phi) + \kappa \sum_{i=1}^q b_i (-\partial_x v_{\mathcal{T}}^{n,i}, \phi), \\ (v_{\mathcal{T}}^{n+1}, \phi) &= (v_{\mathcal{T}}^n, \phi) + \kappa \sum_{i=1}^q \tilde{b}_i \left\{ (-C \cdot \partial_x u_{\mathcal{T}}^{n,i}, \phi) - \frac{1}{\varepsilon} (v_{\mathcal{T}}^{n,i} - F(u_{\mathcal{T}}^{n,i}), \phi) \right\}, \end{aligned} \quad (2.4)$$

where the intermediate stages $(u_{\tau}^{n,i}, v_{\tau}^{n,i})$, $i = 1, \dots, q$ are given by the following coupled systems of q -equations:

$$\begin{aligned} (u_{\tau}^{n,i}, \phi) &= (u_{\tau}^n, \phi) + \kappa \sum_{j=1}^{i-1} a_{ij} (-\partial_x v_{\tau}^{n,j}, \phi), \\ (v_{\tau}^{n,i}, \phi) &= (v_{\tau}^n, \phi) + \kappa \sum_{j=1}^i \tilde{a}_{ij} \left\{ (-C \cdot \partial_x u_{\tau}^{n,j}, \phi) - \frac{1}{\varepsilon} (v_{\tau}^{n,j} - F(u_{\tau}^{n,j}), \phi) \right\}, \end{aligned} \quad (2.5)$$

and the set of constants $A = (a_{ij})$, $b = (b_1, \dots, b_q)$, $\tilde{A} = (\tilde{a}_{ij})$, $\tilde{b} = (\tilde{b}_1, \dots, \tilde{b}_q)$, define the q -stage (ERK) and (DIRK) methods respectively (constant c denotes the relaxation characteristic).

In our experiments we use the following (ERK) methods proposed in [29], [28], which are of 2nd and 3rd order respectively

$$A = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}, \quad b = \begin{pmatrix} 1/2 \\ 1/2 \end{pmatrix}, \quad \tilde{A} = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1/4 & 1/4 & 0 \end{pmatrix}, \quad \tilde{b} = \begin{pmatrix} 1/6 \\ 1/6 \\ 2/3 \end{pmatrix}. \quad (2.6)$$

These explicit RK schemes are known to be appropriate for the discretization of conservation laws. The corresponding diagonally implicit methods (DIRK) that we use are

$$\tilde{A} = \begin{pmatrix} 0 & 0 \\ 1/2 & 1/2 \end{pmatrix}, \quad \tilde{b} = \begin{pmatrix} 1/2 \\ 1/2 \end{pmatrix}, \quad \tilde{\tilde{A}} = \begin{pmatrix} 0 & 0 & 0 \\ 1/2 & 1/2 & 0 \\ 1/4 & 0 & 1/4 \end{pmatrix}, \quad \tilde{\tilde{b}} = \begin{pmatrix} 1/6 \\ 1/6 \\ 2/3 \end{pmatrix}. \quad (2.7)$$

The intermediate stages $(u_{\tau}^{n,i}, v_{\tau}^{n,i})$, $i = 1, \dots, q$ are evaluated at the same time levels $\tau = \tilde{\tau} = (0, 1)$, and $\tau = \tilde{\tau} = (0, 1, 1/2)$ respectively. Note that evaluating the intermediate stages requires the solution of (2.5) which (since it can be decoupled) is a fully explicit finite element scheme. As it was observed in [2] the Relaxation Finite Element scheme (2.5)–(2.4) have the following main characteristics:

- The systems (2.4)–(2.5) are simple since they are explicit except the second system of (2.5) which is semi-implicit.
- They show a remarkable robustness, in terms of the polynomial degrees used in the finite element space.
- They converge as long as $\varepsilon, h \rightarrow 0$ and the discretization steps h, k respects the appropriate stability CFL condition.
- Like every FEM scheme they can be defined on non-uniform grids without modifications, so they can easily combined with grid redistribution in every discrete time-step.

On the other hand, the CFL condition depends also on the inserted relaxation variable v thus, in many cases in order to compute with very small ε the time step has to be severely reduced, cf. tables in the numerical experiments section.

2.2.2 The Switched Relaxation Finite Element scheme (SRFEM)

To overcome the difficulty and since we are interest to investigate the behavior of the RFEM scheme even in the limit area ($\varepsilon \rightarrow 0$), we introduce the *Switching Relaxation Finite Element* (SRFEM) scheme. This is a version of the RFEM scheme (2.4)–(2.5) in which, in each time step the relaxation variables $v_{\tau,i}$ are projected to the equilibrium manifold. The motivation behind the switching scheme is based on the observation:

The relaxation variables $v_{\tau,i}$, $i = 1, \dots, d$ of the model (2.2) initially as well as in the limit ($\varepsilon = 0$), take the values $v_{\tau,i} = F_i(u_{\tau})$, $i = 1, \dots, d$.

The proposed switching scheme apply also these settings at the start of every time step, but *not* at the intermediate stages of the RK method. The choice of the name is due to the fact that the relaxation model (2.1) reduces to the initial CL (1.1) if $v = f(u)$ and thus, the above process *switches* instantaneous the solving method from the relaxation to the direct FEM approximation. We proceed to the detailed definition of the SRFEM scheme. Using the same RK timestepping notation, let \mathcal{T} be a uniform partition of Ω , and $u_{\mathcal{T}}^0 = \Pi_{S_{q,\mathcal{T}}^M}(u_0)$. Assuming that the Switched Relaxation Finite Element approximation $u_{\mathcal{T}}^n$ of the solution instance $u(\cdot, t^n)$ of (2.2) is been computed at moment t^n then, the approximation $u_{\mathcal{T}}^{n+1}$ of the solution instance $u(\cdot, t^{n+1})$ at the next moment t^{n+1} is defined such that for every ϕ in $S_{q,\mathcal{T}}^M$, there holds:

$$(u_{\mathcal{T}}^{n+1}, \phi) = (u_{\mathcal{T}}^n, \phi) + \kappa \sum_{i=1}^q b_i(-\partial_x v_{\mathcal{T}}^{n,i}, \phi), \quad (2.8)$$

where the intermediate stages $(u_{\mathcal{T}}^{n,i}, v_{\mathcal{T}}^{n,i}), i = 1, \dots, q$ are given by the following coupled systems of q -equations:

$$\begin{aligned} (u_{\mathcal{T}}^{n,i}, \phi) &= (u_{\mathcal{T}}^n, \phi) + \kappa \sum_{j=1}^{i-1} a_{ij}(-\partial_x v_{\mathcal{T}}^{n,j}, \phi), \\ (v_{\mathcal{T}}^{n,i}, \phi) &= (\Pi_{S_{q,\mathcal{T}}^M}(F(u_{\mathcal{T}}^n)), \phi) + \kappa \sum_{j=1}^i \tilde{a}_{ij} \left\{ (-C \cdot \partial_x u_{\mathcal{T}}^{n,j}, \phi) - \frac{1}{\varepsilon} (v_{\mathcal{T}}^{n,j} - F(u_{\mathcal{T}}^{n,j}), \phi) \right\}. \end{aligned} \quad (2.9)$$

Note that at every time step, the computation of $v_{\mathcal{T}}^{n+1}$ (second equation of (2.4)) is not required any more since, the integration in time starts with $F(u_{\mathcal{T}}^n)$ instead of $v_{\mathcal{T}}^n$. Thus the computational cost of SRFEM scheme (2.8)–(2.9) and RFEM scheme (2.4)–(2.5) is about the same, for approximations using linear FEM spaces. In the case of RFEM schemes the solution of the second explicit-equation on (2.4) is required while, in the SRFEM case the calculation of the interpolant of $F(u_{\mathcal{T}}^n)$ must be done.

This scheme calculate approximations of some Relaxation type family where the parameter ε is time dependent, i.e. $\varepsilon(t) = 0$ on the moments $t^n, n = 0, 1, \dots$ otherwise $\varepsilon(t) = \varepsilon_0$ is constant. It is an interesting question if the family $\{u_{\mathcal{T}}^{\varepsilon(t)}\}$ is weakly convergent to the solution of the corresponding CL while ε_0 tends zero. In practice SRFEM scheme give bounded approximations even for negligible ε_0 and independently to the time step but, in the regime of the proper CFL stability condition of the solution u . Numerical experiments show that on uniform partitions:

- The parameter ε in SRFEM scheme can be taken arbitrarily small, without change on the time step κ , cf. tables in the numerical experiments section.
- For $\varepsilon > 10^{-5}$ the pointwise switching does not influence the relaxation mechanism so, in that range the given approximations from the SRFEM scheme coincide with those from the RFEM scheme. For negligible values of ε the relaxation mechanism becomes weak and thus in that range the scheme produce dispersive approximations.

2.2.3 The Limit Relaxation Finite Element scheme (LRFEM) and the Direct Finite Element scheme (DFEM)

Motivated by the computational behavior of the SRFEM scheme close to the limit, we introduce also the Limit Relaxation Finite Element scheme (LRFEM) which is derived by taking the limit of (2.9) as $\varepsilon \rightarrow 0$. In that case, the evolution step from the LRFEM approximation $u_{\mathcal{T}}^n$ of the solutions instance $u(\cdot, t^n)$ to the next approximated instance at moment t^{n+1} is defined such that, for every ϕ in $S_{q,\mathcal{T}}^M$:

$$(u_{\mathcal{T}}^{n+1}, \phi) = (u_{\mathcal{T}}^n, \phi) + \kappa \sum_{i=1}^q b_i(-\partial_x v_{\mathcal{T}}^{n,i}, \phi), \quad (2.10)$$

where the intermediate stages $(u_{\tau}^{n,i}, v_{\tau}^{n,i}), i = 1, \dots, q$ are given by the following system of q -equations:

$$\begin{aligned} (u_{\tau}^{n,i}, \phi) &= (u_{\tau}^n, \phi) + \kappa \sum_{j=1}^{i-1} a_{ij} (-\partial_x v_{\tau}^{n,j}, \phi), \\ (v_{\tau}^{n,i} - F(u_{\tau}^{n,i}), \phi) &= 0. \end{aligned} \tag{2.11}$$

Notice that this scheme is independent on the relaxation parameter ε and thus the relaxation mechanism is not present to guarantee the approximation to entropic solutions. Still, we will track the results of that scheme, in order to collect evidence for the behavior of the SRFEM schemes on the limit $\varepsilon = 0$.

One might observe that increasing the switching frequency on the LRFEM scheme i.e. setting $v_{\tau}^{n,i} = F(u_{\tau}^{n,i})$ even on the intermediate time levels (2.11) of the RK-method, the scheme is transformed to the fully discrete scheme of the *Direct Finite Element* (DFEM) approximation of the conservation law (2.3), with evolution step defined by:

$$(u_{\tau}^{n+1}, \phi) = (u_{\tau}^n, \phi) + \kappa \sum_{i=1}^q b_i (F(u_{\tau}^{n,i}), \partial_x \phi), \tag{2.12}$$

where the intermediate stages $u_{\tau}^{n,i}, i = 1, \dots, q$ are given by the following q -equations:

$$(u_{\tau}^{n,i}, \phi) = (u_{\tau}^n, \phi) + \kappa \sum_{j=1}^{i-1} a_{ij} (F(u_{\tau}^{n,j}), \partial_x \phi). \tag{2.13}$$

Numerical experiments show that on uniform partitions:

- While $\varepsilon \rightarrow 0$ the given approximations from the SRFEM scheme tends to the approximation from the LRFEM scheme.
- In certain cases of approximating nonclassical shocks, cf. the Burgers experiment, the given approximations from the LRFEM and DFEM schemes are different.

For the LFEM scheme (2.10)–(2.11) the reduction of the cost compared to the RFEM scheme is significant even for approximations from higher order FEM spaces. In that case the second vector-equation of (2.5) i.e. the semi-implicit term, is replaced with the calculation of an L_2 projection element which is a task of linear complexity, since the inverse mass matrix has been calculated for other needs of the RK method in RFEM. For that scheme the corresponding systems are completely explicit. Finally, in the DFEM scheme (2.12)–(2.13) all the systems are explicit without inserted unknown variables, thus the amount of calculations is the lowest possible.

In summary, the SRFEM scheme is a proper solving mechanism for CL problems, easy to implement in existing RFEM schemes and capable to produce either approximations of relaxation type or direct discretization, for testing RFEM schemes in all the range of the parameter ε . In the sequel these schemes will be combined with appropriate adaptive grid redistribution proposed in the next section.

3 Adaptive Grid Redistribution

Mesh adaptation is a main current stream towards to efficiently compute numerical solutions of complex systems by increasing the resolution of the underline numerical scheme. Several redistribution techniques have been introduced in the near past for solving the problem of proper mesh selection, starting with the "One shot method" of Babuska and the "Self-Adjusting method" of Harten, Hyman to the "Moving Mesh" methods of Leveque, Mackenzie, Petzold, Russell, Tao Tang and others (see bibliography [5, 6, 16, 4, 32, 16, 25, 18, 7, 26, 13, 27, 30, 33, 34]). 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.

The prospect in all of them is to increase the resolution of the final numerical solution when they combined with numerical schemes.

Our approach in this work is different and is based on a uniform distribution principle of appropriate measures defined through density functionals. This approach in some primitive form was introduced in [2] and is related although different to the "one shot" adaptive method of Babuska [5, 6]. We shall treat the problem of proper mesh selection in a general way. Our aim is to sketch the outline of a simple and of low complexity algorithm which can be inserted as an independent substep in numerical applications. In particular it is shown in one space dimension, how this strategy leads to the construction of the appropriate partition.

For some partition $\mathcal{T} = \{K\}$ of Ω into elementary sets, we shall use the following notation:

- For $N \in \mathbb{N}^+$ we denote with Part_N the family of partitions of Ω into N elementary sets K , i.e. $\text{Part}_N(\Omega) = \{\mathcal{T} \subset \wp(\Omega) : \mathcal{T} \text{ partition of } \Omega, \text{ card}(\mathcal{T}) = N\}$.

- $\sigma(\mathcal{T})$ is the σ -algebra generated by the partition \mathcal{T} which, for finite partition $\sigma(\mathcal{T})$ coincide with the set of unions from partition elements, i.e.

$$\sigma(\mathcal{T}) = \left\{ \bigcup_{i=1}^N K_i^{j_i} : j = (j_1, \dots, j_N) \in \{0, 1\}^N \right\}$$

where, $\{0, 1\}^N$ is the set of the N -lengthed sequences of 0's or 1's and the symbols K^1, K^0 denotes the set K and its complement set K^c respectively.

- With **resolution** $_{\mathcal{T}}$ we shall denote a measure which for any set A is defined by

$$\mathbf{resolution}_{\mathcal{T}}(A) = \text{card}\{K \in \mathcal{T} : K \subset A\},$$

i.e., the **resolution** $_{\mathcal{T}}$ of the partition \mathcal{T} over some set A represents the number of elements from \mathcal{T} that are contained in A .

3.1 Grid redistribution strategy

In general the following observation holds true in numerical applications:

*The local accuracy of any numerical approximation over some area of the domain, is an increasing function of the partitions **resolution** over that area, which in turn is bounded from above (e.g. in the case of HCL a CFL type condition imposes an upper bound to the local partitions **resolution**).*

This observation motivates our redistribution policy:

*Seek a new partition $\tilde{\mathcal{T}}$ with **resolution** $_{\tilde{\mathcal{T}}}$ that over any area of the domain is proportional to the local needs in partition elements, as they are measured by some given function.*

Analyzing the above we conclude that the redistribution process is described by the following two steps:

1. At start we have to estimate the "needs in partition elements" over the spatial domain, according to a given strictly positive and bounded functional g , called in sequel *estimator function*.
2. Finally the unknown partition is constructed using properly the estimator function g . In this work we shall take advantage of this function in the distribution sense. Let G be the measure of density g with respect to the Lebesgue measure μ i.e., the G measure of any Lebesgue measurable set $A \subset \Omega$ is defined by

$$G(A) = \int_A g \, d\mu. \quad (3.1)$$

By (3.1) the "local needs in partition elements" over some set A of the domain, are represented by the $G(A)$ measure so the redistribution strategy reads now as follows:

Seek a finite partition $\tilde{\mathcal{T}} = \{K\}$ of Ω such that, at least for sets A from $\sigma(\tilde{\mathcal{T}})$ the **resolution** $_{\tilde{\mathcal{T}}}(A)$ is proportional to the $G(A)$ measure, i.e. for some constant C the new partition is such that:

$$\forall A \in \sigma(\tilde{\mathcal{T}}) \quad G(A) = C \text{ resolution}_{\tilde{\mathcal{T}}}(A). \quad (3.2)$$

In particular if $N = \text{card}(\tilde{\mathcal{T}})$ is given then, by (3.2) for $A = \Omega$ we get $C = G(\Omega)/N$ and so, for the elementary sets $K \in \tilde{\mathcal{T}}$ (which are of unitary **resolution** $_{\tilde{\mathcal{T}}}$) we conclude from (3.2) that:

$$\forall K \in \tilde{\mathcal{T}} \quad G(K) = \frac{G(\Omega)}{N} \quad (3.3)$$

Using the additive property of measures one can conclude (3.2) from equation (3.3), thus these relations are equivalent.

Remark 1. *On one dimensional domains $\Omega = [a, b]$, relation (3.2) indicates immediately the nodes of the new partition $\tilde{\mathcal{T}}$. Indeed, due to the topology of real line the unknown nodes can be numbered in increasing order i.e. $a = \tilde{x}_0 < \tilde{x}_1 < \dots < \tilde{x}_{N-1} < \tilde{x}_N = b$ so, applying (3.2) with the sets $A = [a, \tilde{x}_i)$, $i = 1, 2, \dots, N-1$ we take:*

$$\int_a^{\tilde{x}_i} g(x) dx = \frac{i}{N} \int_a^b g(x) dx \quad (3.4)$$

Since we assume that g is strictly positive, relations (3.4) consult to equations with unique solutions the unknown nodes \tilde{x}_i , $i = 1, 2, \dots, N-1$.

Note that only for simplicity we examine strictly positive estimator functions. If the range of g contains the zero value, the set of the unknown nodes is not uniquely defined. In that case we can select one possible set of nodes by proper treatment of (3.4) (see more about the quantile function in [8], pp.189-191).

Definition: *In the sequel if g is a given estimator function, we shall call **G-uniform mesh**, every finite partition $\tilde{\mathcal{T}} \in \text{Part}_N(\Omega)$ of Ω for which, the corresponding G measure (3.1) has the property (3.3). The name is chosen to express the fact that, due to property (3.3), the G measure is distributed uniformly on the elements of those partitions.*

Lets note two properties of G-uniform meshes.

I) If $\mathcal{T} = \{K\} \in \text{Part}_N(\Omega)$ is a G-uniform partition then, by the definition of the corresponding G measure (3.1) and by (3.3), it follows:

$$\forall K \in \mathcal{T} \quad C \leq \max_{x \in K} \{g(x)\} \mu(K) \leq \max_{x \in K} \{g(x)\} \text{diam}(K)^d, \quad (3.5)$$

where $d = \text{dim}(\Omega)$, $C = G(\Omega)/N$.

II) For a given estimator function g let G be the corresponding measure (3.1). By the additive property of measures we have that, for any partition $\mathcal{T} = \{K\} \in \text{Part}_N(\Omega)$:

$$\frac{G(\Omega)}{N} = \frac{1}{N} \sum_{K \in \mathcal{T}} G(K) \leq \max_{K \in \mathcal{T}} \{G(K)\}.$$

Thus, in the case where \mathcal{T} is a G-uniform partition then, by (3.3) the above relation holds as strict equality, therefore the G-uniform partition solves the min-max problem:

For a given estimator function g seek a finite partition $\tilde{\mathcal{T}} \in \text{Part}_N(\Omega)$ such that:

$$\max_{K \in \tilde{\mathcal{T}}} \{G(K)\} = \min_{\mathcal{T} \in \text{Part}_N(\Omega)} \max_{K \in \mathcal{T}} \{G(K)\}. \quad (3.6)$$

Working with G-uniform meshes is a flexible way to produce partitions which satisfy certain restrictions, simply by selecting the proper estimator function g . The following remarks could also be thought as examples.

Remark 2. In the case of vector approximations $w = (w_1, \dots, w_M)^T$, the redistribution process can be used to define either a different partition for each component or a common partition for all the components, as follows:

Let g_k be the estimator function for the k -th component w_k , $k = 1, \dots, M$. Then we can use as common estimator function some weighted average, i.e. for positive weights a_1, \dots, a_M :

$$g(x) = \sum_{k=1}^M a_k g_k(x) \quad (3.7)$$

In our experiments we use the choice of weights that normalize the terms of the sum, i.e. $a_k = 1 / \int_{\Omega} g_k d\mu$, $k = 1, \dots, M$. In this way the **resolution** of the generated partition satisfy equally the local needs of each component.

Remark 3. As mentioned initially, in many cases the mesh density needs to be bounded above for stability reasons. Therefore, if we choose as estimator function some power g^p (for $p \in [0, 1]$) instead of the function g then, the corresponding G measure will be defined by:

$$G(A) = \int_A g^p d\mu. \quad (3.8)$$

For $p = 0$ or if the estimator is constant function, we observe from (3.8) that the G measure becomes proportional to the Lebesgue measure of the domain. Thus in these cases we conclude from (3.3) that the above redistribution process produce a uniform partition of the Ω . While p increases and if g is not constant function, the G measure (3.8) is distributed non uniformly in space, forcing this way the redistribution procedure to generate a non uniform partition with the property (3.5) thus, the minimum diameter of the partition elements respects the relation

$$\min_{K \in \mathcal{T}} \left\{ \left(\frac{C}{\max_{x \in K} \{g^p(x)\}} \right)^{1/d} \right\} \leq \min_{K \in \mathcal{T}} \{diam(K)\}. \quad (3.9)$$

So choosing p appropriately, the **resolution** of the generated G -uniform mesh follows the estimator function g while it respects a given upper bound.

Another reason for using some power of g as the actual estimator function is the "saturation" of the method. Since the above process take advantage of the estimator function in the distribution sense, the mean value of the estimator function should be close either to the minimum or to the maximum values in order to transfer productive informations over all the areas. But, in HCL problems it comes that the estimator function is a lot more sensitive over discontinuity areas and so, small changes over smooth areas are sweeping away. One solution in that problem is to rescale non-linear the estimators range, by using some power of the estimator function.

Computational issue: At computational level the function x^y is not well defined when x and y are both negligible positive numbers. For negligible y and while x tends zero, this function oscillate in the range $[0, 1]$ due to the finite cardinality of machine numbers (for implementations on ANSI C see ISO/IEC 9899 for proper range of the parameters x, y). To overcome the difficulty in our calculations we use the function $(\max\{10^{-20}, x\})^y$ as proper computational version of the power x^y . In this way the result behave always monotonically with respect to x and produce positive quantities even for the cases $x = 0$ and/or $y = 0$.

3.2 Application to dynamic PDE's on one dimensional domains

The numerical solution of dynamic PDE's can be constructed in several ways. In this work we examine the case where the numerical solution is constructed like a sequence of FEM spatial approximations of solution

instances, calculated by the evolution steps of the numerical schemes RFEM, SRFEM, LRFEM, DFEM. In that case the redistribution process could be applied at the start of every evolution step in order to produce partitions which improves the resolution of the approximated instance and it should be ended with the reconstruction of the current approximated instance to the new mesh.

Definition: We shall denote with $\mathbf{GMesh}(\mathcal{T}, w_{\mathcal{T}})$, any numerical process which produce a G -uniform partition $\tilde{\mathcal{T}}$ for some approximation function $w_{\mathcal{T}}$ using calculations on a given partition \mathcal{T} . The reconstruction step shall be denoted with $\mathbf{Rec}(\mathcal{T}, w_{\mathcal{T}}, \tilde{\mathcal{T}})$ since it depends on the given data and the new mesh.

Using these definitions, if $w_{\mathcal{T}}$ is a given approximation function defined on some partition \mathcal{T} then, the proposed Adaptive Grid Redistribution (**AGR**) returns a pair which contains a new partition $\tilde{\mathcal{T}}$ and a reconstructed function $\tilde{w}_{\tilde{\mathcal{T}}}$. The process can be represented by the equation:

$$(\tilde{\mathcal{T}}, \tilde{w}_{\tilde{\mathcal{T}}}) = \mathbf{AGR}(\mathcal{T}, w_{\mathcal{T}}),$$

where

$$\begin{aligned} \tilde{\mathcal{T}} &= \mathbf{GMesh}(\mathcal{T}, w_{\mathcal{T}}), \\ \tilde{w}_{\tilde{\mathcal{T}}} &= \mathbf{Rec}(\mathcal{T}, w_{\mathcal{T}}, \tilde{\mathcal{T}}). \end{aligned} \quad (3.10)$$

Although the above theory is valid in any space dimension, in the sequel we restrict the presentation on one dimensional domain $\Omega = [a, b] \subset \mathbb{R}$. Let \mathcal{T} be a finite partition of the domain $\Omega = [a, b]$ and $\{x_i\}_{i=0}^N$ are the corresponding nodes numbered in increasing order, i.e. $a = x_0 < x_1 < \dots < x_{N-1} < x_N = b$.

In this work we assume that the estimator is piecewise constant function defined on the intermediate grid, i.e. $g(x) = g(x_0)$, $x \in [x_0, x_{1/2})$, $g(x) = g(x_i)$, $x \in [x_{i-1/2}, x_{i+1/2})$, $i = 1, 2, \dots, N-1$ and $g(x) = g(x_N)$, $x \in [x_{N-1/2}, x_N]$ so, the corresponding distribution function $G([a, x]) = \int_a^x g(z) dz$ used in (3.4), is a piecewise linear function. Thus, having the values $G_i = G([a, x_i])$, $i = 0, 1, \dots, N$ on the node points $\{x_i\}_{i=0}^N$ of the old partition then, the equations (3.4) can immediately be solved so we conclude that the nodes $\{\tilde{x}_i\}_{i=0}^N$ of the new partition are $\tilde{x}_0 = a$ and for $\tilde{G}_0 = 0, k_0 := 0, i = 1, 2, \dots, N$ they given by:

$$\tilde{G}_i = \frac{i}{N} G_N, \quad k_i = \max_{k_{i-1} \leq \ell \leq N} \{\ell : G_{\ell} \leq \tilde{G}_i\}, \quad \tilde{x}_i = x_{k_i} + \frac{\tilde{G}_i - G_{k_i}}{G_{k_{i+1}} - G_{k_i}} (x_{k_{i+1}} - x_{k_i}). \quad (3.11)$$

In our experiments we use as reconstruction function the interpolant projection $\Pi_{S_{q,\tilde{\mathcal{T}}}}(w_{\mathcal{T}})$ on the corresponding FEM space $S_{q,\tilde{\mathcal{T}}}$ to the new partition $\tilde{\mathcal{T}}$, thus the reconstruction step is defining in general by the equation $\tilde{w}_{\tilde{\mathcal{T}}} = \Pi_{S_{q,\tilde{\mathcal{T}}}}(w_{\mathcal{T}})$. Especially for the case of linear FEM spaces, the values $\tilde{w}_{\tilde{\mathcal{T}}}(\tilde{x}_i)$ of the reconstructed function on the new nodes can explicitly be found, i.e. for the same sequences of distribution values $\{\tilde{G}_i\}_{i=0}^N$ and indexes $\{k_i\}_{i=0}^N$ as in (3.11), the values on the new partition are, $\tilde{w}_{\tilde{\mathcal{T}}}(\tilde{x}_0) = w_{\mathcal{T}}(x_0)$ and for $i = 1, 2, \dots, N$ they given by:

$$\begin{aligned} \tilde{w}_{\tilde{\mathcal{T}}}(\tilde{x}_i) &= w_{\mathcal{T}}(x_{k_i}) + \frac{\tilde{x}_i - x_{k_i}}{x_{k_{i+1}} - x_{k_i}} (w_{\mathcal{T}}(x_{k_{i+1}}) - w_{\mathcal{T}}(x_{k_i})) \\ &\stackrel{(3.11)}{=} w_{\mathcal{T}}(x_{k_i}) + \frac{\tilde{G}_i - G_{k_i}}{G_{k_{i+1}} - G_{k_i}} (w_{\mathcal{T}}(x_{k_{i+1}}) - w_{\mathcal{T}}(x_{k_i})). \end{aligned} \quad (3.12)$$

It comes out that the **AGR** process is an algorithm of low complexity (linear for linear FEM spaces) since in those relations the indexes k_i can be easily obtained.

3.2.1 Estimator functions for CL problems

Two estimator functions for the AGR process which experimentally have been proved good choices for applications on Conservation Laws problems are:

- **The a posteriori estimator** A natural choice of estimator function for CL problems comes from the a posteriori error analysis. For this choice the **GMesh** process will increase the resolution over areas

with big estimating error which in turn we hope to increase the accuracy over smooth areas due to Taylor expansion. Moreover, as it shown in [2, 15] the estimating error increases significant over the shock areas so dispersive terms which are dominant on these areas will be vanish on the corresponding G-uniform partition. The fundamental tool for the a posteriori error is the local variation of the approximate solution $u_{\mathcal{T}}$, therefore this estimator is defined as follows:

$$g(x) = \int_{\mathcal{D}} \text{Var}(u_{\mathcal{T}}(s)) ds \quad (3.13)$$

where \mathcal{D} is an interval contains the point x . At the discrete level, given a partition $\{x_i\}_{i=0}^N$, we approximate $g(x_i)$ on the intervals $[x_{i-1}, x_{i+1}]$ by g_i , given from the discretization of (3.13) using the trapezoidal integration rule

$$g_i = \frac{1}{2} (|u_{\mathcal{T}}(x_{i+1}) - u_{\mathcal{T}}(x_i)|(x_{i+1} - x_i) + |u_{\mathcal{T}}(x_i) - u_{\mathcal{T}}(x_{i-1})|(x_i - x_{i-1})), \quad i = 1, \dots, N \quad (3.14)$$

- **The curvature estimator** A useful family of estimator functions is based on geometrical characteristics of the solution. Since in CL problems the most interest areas are over discontinuities (either of the solution or its derivatives), one possible choice as estimator function is an appropriate approximation of the curvature of the already computed solution. The curvature of the function v is defined by

$$\frac{|v''(x)|}{(1 + v'(x)^2)^{\frac{3}{2}}}.$$

To this end, given a partition $\{x_i\}_{i=0}^N$, g is defined as the piecewise constant function with values g_i which are given by the inverse outer radius of the plane points $A_j = (x_j, u_{\mathcal{T}}(x_j))$, $j = i - 1, i, i + 1$. That is :

$$g_i = 2 \frac{\|(A_{i+1} - A_i) \times (A_i - A_{i-1})\|}{\|A_i - A_{i-1}\| \|A_{i+1} - A_i\| \|A_{i+1} - A_{i-1}\|}, \quad i = 1, \dots, N \quad (3.15)$$

where $\|\cdot\|$ denotes the Euclidean norm.

An elementary calculation shows that

$$g_i = \frac{\frac{2}{x_{i+1} - x_{i-1}} \left| \frac{u_{\mathcal{T}}(x_i) - u_{\mathcal{T}}(x_{i-1})}{x_i - x_{i-1}} - \frac{u_{\mathcal{T}}(x_{i+1}) - u_{\mathcal{T}}(x_i)}{x_{i+1} - x_i} \right|}{\left(1 + \left(\frac{u_{\mathcal{T}}(x_i) - u_{\mathcal{T}}(x_{i-1})}{x_i - x_{i-1}}\right)^2\right)^{1/2} \left(1 + \left(\frac{u_{\mathcal{T}}(x_{i+1}) - u_{\mathcal{T}}(x_i)}{x_{i+1} - x_i}\right)^2\right)^{1/2} \left(1 + \left(\frac{u_{\mathcal{T}}(x_{i+1}) - u_{\mathcal{T}}(x_{i-1})}{x_{i+1} - x_{i-1}}\right)^2\right)^{1/2}} \quad (3.16)$$

which approximates the curvature of $u_{\mathcal{T}}(x_i)$.

3.2.2 Graphical representation of the AGR process and the effect of the parameter p .

The **Gmesh** step of the **AGR** process can be represented graphically in Figure 1. Starting with some initial data (solid line) we calculate the estimator function $g(x)$ (dotted line) and the distribution function of the corresponding measure $G([a, x])$ (increasing dotted line). The proposed mesh (x position of the vertical lines under the distribution function) is given by inverting through the distribution function a uniform mesh of $N + 1$ points applied on the y -axis (y position of the horizontal lines, over the distribution function). Note that in these graphic representations the y -coordinates are valid only for the data while either the estimator or its distribution function were shifted and scaled vertically. In figure 1 we see the effect of the parameter p on the AGR procedure. While p increases the resulting mesh becomes more dense over discontinuity areas while it rarefies over smooth areas. The parameter p must be bounded from the above since, it may lead to grids with very poor resolution on the smooth areas. This bound depend on the estimator function and for the cases "a posteriori", "curvature" we experimentally know that it varies in the interval [0.12, 0.15].

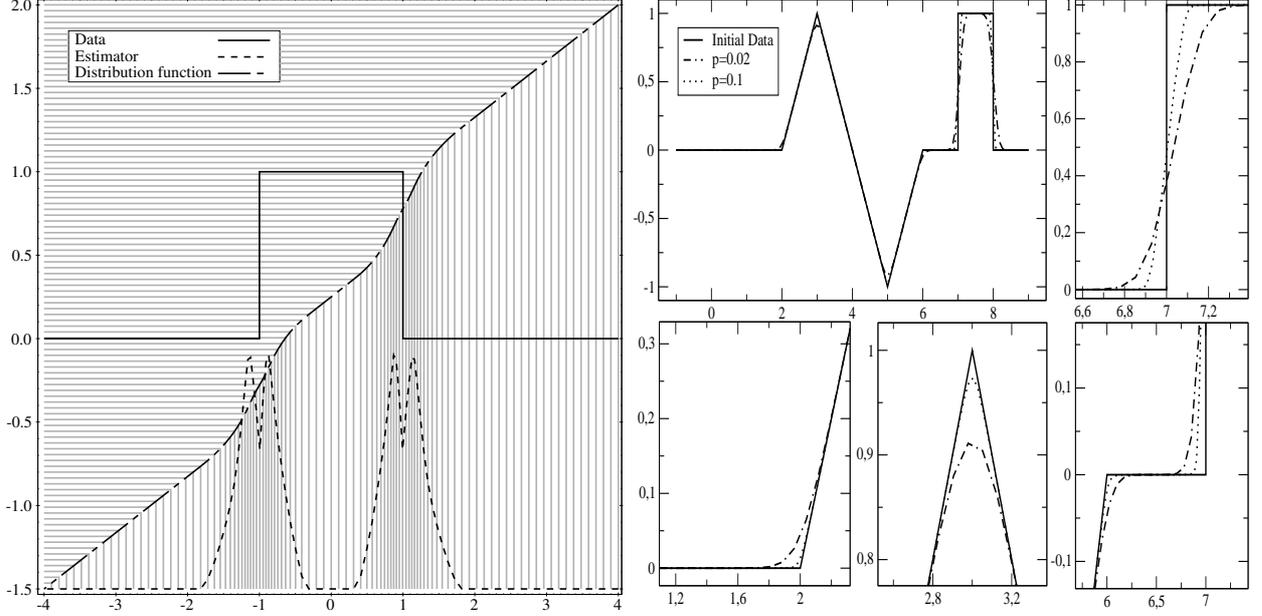


Figure 1: Graphical representation of the AGR procedure (left) and the effect of the parameter p on the reconstructed function to the new partition (right).

3.2.3 Finite Element schemes on adaptive G-uniform meshes

Let **Solver** denote in general one of the studied schemes RFEM, SRFEM, LRFEM, DFEM. We shall examine the evolution step for the pair which contains a partition and a solution instant on it, during the evolving steps of the **Solver** scheme. In the case where the **Solver** scheme apply only on a uniform partition \mathcal{T}_0 , the evolution step for the pair can be represented by the equation:

$$(\mathcal{T}_{n+1}, u_{\mathcal{T}_{n+1}}^{n+1}) = (\mathcal{T}_0, \mathbf{Solver}(\mathcal{T}_n, u_{\mathcal{T}_n}^n)), \quad n = 0, 1, \dots$$

while, in the adaptive G -uniform case, by the system of equations:

$$\begin{aligned} (\mathcal{T}_{n+1}, \tilde{u}_{\tilde{\mathcal{T}}_{n+1}}) &= \mathbf{AGR}(\mathcal{T}_n, u_{\mathcal{T}_n}^n) \\ (\mathcal{T}_{n+1}, u_{\mathcal{T}_{n+1}}^{n+1}) &= (\mathcal{T}_{n+1}, \mathbf{Solver}(\tilde{\mathcal{T}}_{n+1}, \tilde{u}_{\tilde{\mathcal{T}}_{n+1}})), \quad n = 0, 1, \dots \end{aligned}$$

We present the AGR adaptive version of the SRFEM scheme while the rest schemes are defined in similar way. Let \mathcal{T}_0 be a uniform partition of Ω with $\text{card}(\mathcal{T}_0) = N$, S_{q, \mathcal{T}_0} be the corresponding FEM space and $u_{\mathcal{T}_0}^0 = \Pi_{S_{q, \mathcal{T}_0}}^M(u_0)$. Assuming that at t^n , the pair $(\mathcal{T}_n, u_{\mathcal{T}_n}^n)$ is been computed then, the pair $(\mathcal{T}_{n+1}, u_{\mathcal{T}_{n+1}}^{n+1})$ at t^{n+1} moment of the Switched Relaxation approximation on G-uniform mesh is defined by:

$$\begin{aligned} (\mathcal{T}_{n+1}, \tilde{u}_{\tilde{\mathcal{T}}_{n+1}}) &= \mathbf{AGR}(\mathcal{T}_n, u_{\mathcal{T}_n}^n) \\ \text{and for every } \phi &\text{ in } S_{q, \mathcal{T}_{n+1}}^M \\ (u_{\mathcal{T}_{n+1}}^{n+1}, \phi) &= (\tilde{u}_{\tilde{\mathcal{T}}_{n+1}}, \phi) + \kappa \sum_{i=1}^q b_i (-\partial_x v_{\mathcal{T}_{n+1}}^{n,i}, \phi) \end{aligned} \tag{3.17}$$

where the intermediate stages $(u_{\tau_{n+1}}^{n,i}, v_{\tau_{n+1}}^{n,i}), i = 1, \dots, q$ are given by the following coupled systems of q -equations:

$$\begin{aligned} (u_{\tau_{n+1}}^{n,i}, \phi) &= (\tilde{u}_{\tau_{n+1}}, \phi) + \kappa \sum_{j=1}^{i-1} a_{i,j} (-\partial_x v_{\tau_{n+1}}^{n,j}, \phi) \\ (v_{\tau_{n+1}}^{n,i}, \phi) &= (\Pi_{S_{q,\tau_{n+1}}}^M(F(\tilde{u}_{\tau_{n+1}})), \phi) + \kappa \sum_{j=1}^i \tilde{a}_{i,j} \left\{ (-C \cdot \partial_x u_{\tau_{n+1}}^{n,j}, \phi) - \frac{1}{\varepsilon} (v_{\tau_{n+1}}^{n,j} - F(u_{\tau_{n+1}}^{n,j}), \phi) \right\} \end{aligned} \quad (3.18)$$

The above scheme inherits the characteristics of the scheme (2.8)–(2.9). Indeed (3.17)–(3.18) exploits the stabilizing mechanisms of mesh redistribution for improving the resolution around discontinuities and of switching for relaxing the dependence of the time step κ on the relaxation parameter ε . Furthermore schemes (2.8)–(2.9) and (3.17)–(3.18) share the same complexity.

Remark 4. *Numerical evidence show that switched relaxation scheme on G-uniform mesh (3.17)–(3.18) (and especially those with estimator function the curvature of the numerical solution (3.15)), preserve the high resolution of the method even for ε almost 0.*

4 Numerical Experiments

In this section we present the experiments we've done with the RFEM, SRFEM, LRFEM and DFEM schemes. In the first problem we study the behavior of the AGR procedure using the DRFEM scheme on partitions of 101 grid points. In the rest of the presented results, the integration in time was done with the 3rd order of the proposed ERK–DIRK methods and the spatial approximations were given by Periodic Linear Finite Elements on partitions of 201 node points. The schemes were tested on uniform and adaptive G-uniform partitions with estimator function the curvature of the numerical solution. For the RFEM scheme and for all the range $[10^{-5}, 10^{-2}]$ of the parameter ε the results were coincide with those from the SRFEM scheme. Therefore rather than presenting results from that scheme we give comparison tables of the maximum uniform time-step over of which these methods could integrate the calculations to the given time-end. From these tables one can observe uniformly for all the presented problems that:

- *In the case of SRFEM, LRFEM, DFEM schemes the maximum uniform time-step needed for integrating to the time-end it depends only on the corresponding CFL condition while for the case of the RFEM scheme it depends and on relaxation parameter ε .*

For all the other schemes we present the results in group of figures for comparison reasons. Each group contains two separate figures with the results on uniform and those on adaptive G-uniform partitions. The first group show the approximations from the SRFEM scheme for middle and small value of ε and also the approximation from the LRFEM scheme. The next group contains results from the LRFEM and the DFEM schemes. The last group contains only one figure where the results from the DFEM scheme on either partition cases, are compared with some reference solution. From these figures one can observe uniformly for all the presented problems that:

- *There exist always a proper range for the parameters ε, p for which the SRFEM scheme on G-uniform partitions, produce satisfactory approximations of the entropic solution and free of dispersive oscillations.*
- *In either uniform or G-uniform partitions, while ε tends zero the approximation from the SRFEM scheme becomes similar to the LRFEM's scheme approximation.*
- *The LRFEM and DFEM approximations even though can be different on uniform partitions, are always similar on adaptive G-uniform partitions.*

It seems that the DFEM scheme on adaptive G -uniform partitions can produce approximations of the entropic solution free of dispersive oscillations.

4.1 A Stationary equation

The first problem

$$\begin{aligned} u(x, 0) &= u_0(x) \\ \partial_t u &= 0 \end{aligned} \quad (4.1)$$

is for studying the behavior of the AGR process. The flux in that problem is $F(u) = 0$ so observe from (2.10)–(2.11) and (2.12)–(2.13) that all the calculations in the LRFEM, DFEM schemes are vanish. Thus the evolving step of the adaptive version of those schemes eliminate to the AGR substep.

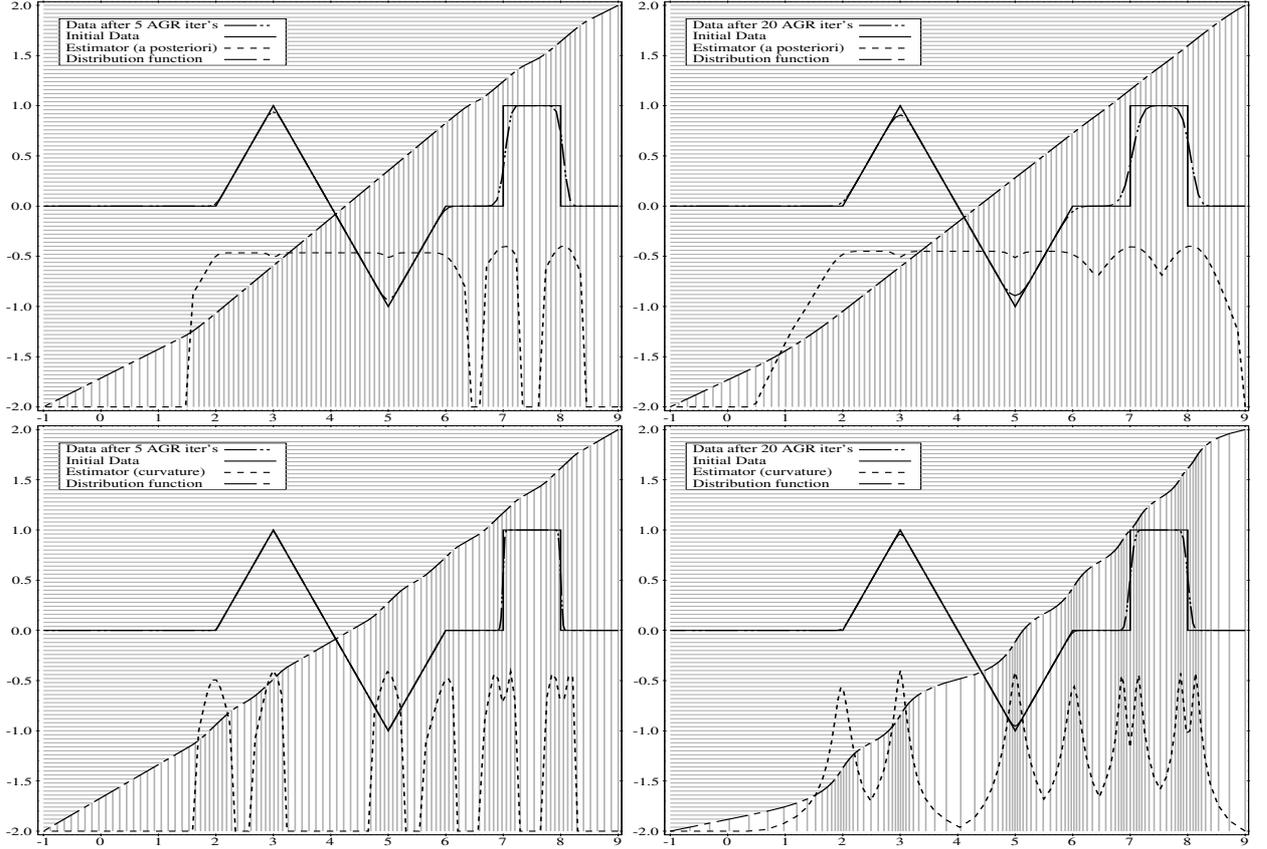


Figure 2: Iteratively application of the AGR process on Riemann Data ($N=101$ nodes, $p=0.012$), after 5 iter's (first column) and after 20 iter's (second column). The results in the first row were given by using the a-posteriori estimator function while in the second row by using the curvature estimator function.

Therefore, and since the solution of (4.1) is $u(x, t) = u_0(x)$, we expect that the sequence $\{(\mathcal{T}_n, u_{\mathcal{T}_n}^n)\}_{n \in \mathbb{N}}$ defined by:

$$\begin{aligned} \mathcal{T}_0 &\text{ is uniform partition of } \Omega, \quad u_{\mathcal{T}_0}^0 = \Pi_{S_{q, \mathcal{T}_0}^M}(u_0), \\ (\mathcal{T}_{n+1}, u_{\mathcal{T}_{n+1}}^{n+1}) &= \mathbf{AGR}(\mathcal{T}_n, u_{\mathcal{T}_n}^n), \quad n = 1, 2, \dots, \end{aligned} \quad (4.2)$$

must converge to some limit pair $(T^*, u_{T^*}^*)$ with the term $u_{T^*}^*$ be an approximation of the initial data $u_{T_0}^0$. Numerical evidence show that, indeed the resulting solution $(T^n, u_{T_n}^n)$ after a number of iterations which depend on the cardinality of the partition N , remains steady suggesting that the sequence (4.2) attains its limit. We present examples (figures 2) for partitions of 100 elements where the resulting solution $(T^n, u_{T_n}^n)$ remains steady after about 15 iterations. Observe that the final reconstructed data are indeed diffusive approximation of the initial data $u_{T_0}^0$. Therefore we must conclude that the AGR process induce diffusion by its own and in the right direction like the "upwinding". Also from figure 2 one can observe that, the approximation on G-uniform mesh with estimator function the variance is more diffusive than the one produced by using the curvature as estimator function.

Finally observe from figures 3 that the results from the AGR process depend on the global complexity of the data structure. For initial data of more complex structure (right figure) the final data after 20 iter's are more diffusive than those on the left figure. One way to reduce the dependence on the global complexity is to adapt in our algorithm the idea of Harten and Hyman [16] to keep prefixed a number of nodes of a coarser mesh. That is to divide the domain $[a, b]$ in to M uniform subintervals and apply the AGR process in each one of them by defining N/M subintervals where the corresponding G-measures shall be uniformly distributed.

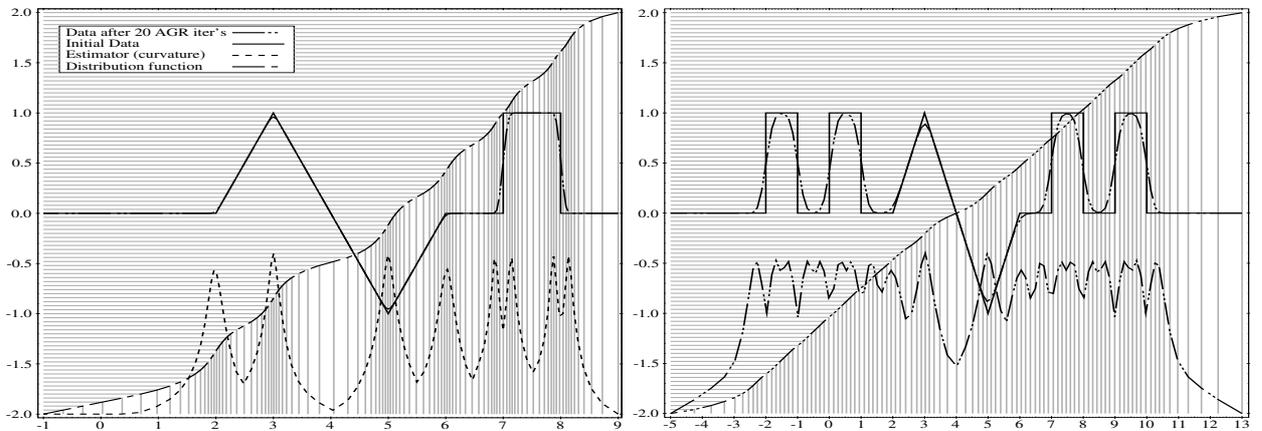


Figure 3: Iteratively application of the AGR process on Riemann Data ($N=101$ nodes, $p=0.012$) after 20 iter's. The diffusive behavior of the AGR process depend on the complexity of data structure.

4.2 A Transport equation

First we consider the solution of the following transport equation with Riemman initial data:

$$\begin{cases} u_0 = \mathbb{X}_{[0.3,0.6)} \\ \partial_t u + \partial_x(2u) = 0 \end{cases}, \quad x \in [0, 1], \quad t \in [0, 1]. \quad (4.3)$$

This is a linear problem of CL for testing the numerical diffusion inserted by the solving process. Due to the periodic boundary conditions every half time unit the corresponding instance of the exact solution should be coincide with the initial data. Numerically this problem is a difficult task for every FEM method which is using continuous elements. Between the evolution steps of the first period the shape of the initial data becomes diffusive and in the sequel it remains steady.

The following table presents the relaxation parameters and the minimum intermediate uniform time steps needed for the RFEM, SRFEM, LRFEM, DFEM schemes in order to integrate the calculations on uniform

partition to the time moment $t = 1$.

<i>Scheme</i>	ε	C_1	<i>Time steps</i>
<i>RFEM</i>	$\varepsilon = 5 \cdot 10^{-4}$	4.5	400
<i>RFEM</i>	$\varepsilon = 1.25 \cdot 10^{-4}$	4.5	1600
<i>RFEM</i>	$\varepsilon = 5 \cdot 10^{-5}$	4.5	3800
<i>SRFEM</i>	$\varepsilon = 5 \cdot 10^{-4}$	4.5	400
<i>SRFEM</i>	$\varepsilon = 1.25 \cdot 10^{-4}$	4.5	400
<i>SRFEM</i>	$\varepsilon = 5 \cdot 10^{-5}$	4.5	400
<i>LRFEM</i>	—	—	400
<i>DFEM</i>	—	—	400

Figures 4, 5, 6 present results for the moment $t = 1$.

4.3 A Burgers equation

The second problem we consider is a Burgers equation with Riemman initial data:

$$\begin{cases} u_0 = \mathbb{X}_{[0,5)} - \mathbb{X}_{[-5,0) \cup [5,6]} \\ \partial_t u + \partial_x(u^2/2) = 0 \end{cases}, \quad x \in [-5, 6], t \in [0, 2]. \quad (4.4)$$

The exact solution of that problem contains a centered rarefaction wave emanating from the point $x = 0$ and a steady sock at the point $x = 5$. Note that the finite difference scheme of Roe as well the high resolution TVD schemes of MUSCL type, need proper modifications (entropy fix) in order to overcome some entropy violation in the rarefaction area, see [23]. Observe from the figures that the AGR finite element schemes produce satisfactory approximations without further modifications.

The following table presents the relaxation parameters and the minimum intermediate uniform time steps needed for the RFEM, SRFEM, LRFEM, DFEM schemes in order to integrate the calculations on uniform partition to the time moment $t = 2$.

<i>Scheme</i>	ε	C_1	<i>Time steps</i>
<i>RFEM</i>	$\varepsilon = 10^{-4}$	100	3800
<i>RFEM</i>	$\varepsilon = 10^{-5}$	100
<i>RFEM</i>	$\varepsilon = 10^{-6}$	100
<i>SRFEM</i>	$\varepsilon = 10^{-4}$	100	200
<i>SRFEM</i>	$\varepsilon = 10^{-5}$	100	200
<i>SRFEM</i>	$\varepsilon = 10^{-6}$	100	200
<i>LRFEM</i>	—	—	200
<i>DFEM</i>	—	—	200

Figures 7, 8, 9 present results for the moment $t = 2$. The exact solution was calculated using the corresponding characteristic lines and is given by:

$$u(x, 2) = \begin{cases} -1, & x \in [-5, -2) \\ 0.5x, & x \in [-2, 2) \\ 1, & x \in [2, 5) \\ -1, & x \in [5, 6] \end{cases}$$

Observe from figure 8 that in this problem the LRFEM and DFEM approximations are different on uniform partitions while on adaptive G-uniform partitions they coincide.

4.4 A Buckley-Leverett equation

The next problem we consider is a Buckley-Leverett equation with Riemman initial data:

$$\begin{cases} u_0 = \mathbb{X}_{[0,0.1] \cup [0.5,1]} \\ \partial_t u + \partial_x \left(\frac{u^2}{u^2 + 0.5(1-u)^2} \right) = 0 \end{cases}, \quad x \in [0, 1], t \in [0, 0.4]. \quad (4.5)$$

This problem is chosen in order to test the behavior of these schemes on non convex fluxes. While time pass, two rarefaction waves are constructed emanating from the points $x = 0.1$, $x = 0.5$. The left side of these waves, is kept steady while the right side terminates on moving shocks to the right.

The following table presents the relaxation parameters and the minimum intermediate uniform time steps needed for the RFEM, SRFEM, LRFEM, DFEM schemes in order to integrate the calculations on uniform partition to the time moment $t = 0.4$.

<i>Scheme</i>	ε	C_1	<i>Time steps</i>
<i>RFEM</i>	$\varepsilon = 5 \cdot 10^{-4}$	20	400
<i>RFEM</i>	$\varepsilon = 1.25 \cdot 10^{-4}$	35	800
<i>RFEM</i>	$\varepsilon = 5 \cdot 10^{-5}$	100	3400
<i>SRFEM</i>	$\varepsilon = 5 \cdot 10^{-4}$	100	400
<i>SRFEM</i>	$\varepsilon = 5 \cdot 10^{-5}$	100	400
<i>SRFEM</i>	$\varepsilon = 5 \cdot 10^{-6}$	100	400
<i>LRFEM</i>	–	–	400
<i>DFEM</i>	–	–	400

Figures 10, 11, 12, present results for the the moment $t = 0.28$ when the left shock almost meet the steady point of the right rarefaction wave. Observe from Figures 11-right and 12 that the adaptive LRFEM, DFEM schemes produce entropic approximation but some oscillations of small amplitude are still remaining around the socks. The approximation of SRFEM with $\varepsilon = 5 \cdot 10^{-6}$ is free of oscillations while the same scheme with $\varepsilon = 5 \cdot 10^{-8}$ is similar to the approximation obtained by LRFEM. The exact solution was calculated by first solving the non-linear equation of the characteristic lines, in terms of x :

$$u(x + 0.28 \cdot \partial_x \left(\frac{u_0^2(x)}{u_0^2(x) + 0.5(1-u_0(x))^2} \right), 0.28) = u_0(x)$$

using the bisection method and for 501 points uniformly distributed on the spatial domain, and then eliminate the multi-valued areas by a correct shock using the equal-area rule, see [23].

4.5 A Shallow water system

The last problem we consider is a CL system of two equations modelling shallow water flows:

$$\begin{cases} \begin{pmatrix} u_{0,1} \\ u_{0,2} \end{pmatrix} = \begin{pmatrix} 2 \mathbb{X}_{[0.3,0.4]} + 1.2 \cdot \mathbb{X}_{[0.6,0.7]} \\ 0 \end{pmatrix} \\ \partial_t \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} + \partial_x \left(\frac{u_2}{u_1} + \frac{u_1^2}{2} \right) = 0 \end{cases}, \quad x \in [0, 1], t \in [0, 1]. \quad (4.6)$$

This problem models the gravitational collapse of two liquid towers on a liquid surface (for gravitational constant $g = 1$). The hight of the liquid surface is represented by the first component of the solution while the second component represents the discharge of the liquid, see [23]. While time pass, four sock and rarefaction waves are formed. These socks are moving to the exterior at opposite directions and and approximately at $t = 0.085$ the two in the middle interact with each other yielding a rather complicated structure.

The following table presents the relaxation parameters and the minimum intermediate uniform time steps needed for the RFEM, SRFEM, LRFEM, DFEM schemes in order to integrate the calculations on uniform

partition to the time moment $t = 1.0$.

<i>Scheme</i>	ε	C_1, C_2	<i>Time steps</i>
<i>RFEM</i>	$\varepsilon = 5 \cdot 10^{-4}$	4, 4	600
<i>RFEM</i>	$\varepsilon = 1.25 \cdot 10^{-4}$	4, 4	1600
<i>RFEM</i>	$\varepsilon = 5 \cdot 10^{-5}$	4, 4	3800
<i>SRFEM</i>	$\varepsilon = 5 \cdot 10^{-4}$	4, 4	400
<i>SRFEM</i>	$\varepsilon = 1.25 \cdot 10^{-4}$	4, 4	400
<i>SRFEM</i>	$\varepsilon = 5 \cdot 10^{-5}$	4, 4	400
<i>LFEM</i>	—	—	400
<i>DFEM</i>	—	—	400

Figures 13, 14, 15, 16, 17, 18, present results for the moment $t = 0.14$ where the solution instance have a very complicate structure.

5 Conclusions

In this work we introduce and study finite element schemes based on relaxation models and the *Adaptive Grid Redistribution* process.

Among them, probably the most appropriate scheme for shock computations is the *Switched Relaxation Finite Element* scheme. It is constructed by a modification of the *Relaxation Finite Element* schemes in order to approximate solutions of the Conservation Law problems with timestep independent from the relaxation parameter. The improvement of the RFEM scheme was made by setting the relaxation variables to the equilibrium state at the start of every evolution step.

The AGR process was designed to reconstruct the numerical solution on a new partition of the spatial domain. The generated partition was defined to have resolution controlled by some estimator function g which actually represents some (pre-)selective characteristics of the numerical solution. From the uniform distribution of the corresponding measure it was shown on one dimensional domains, how this policy leads to the new partition—the *G-uniform* partition. For the choice g^p as estimator function with p left as a free parameter, the AGR became flexible for producing *G*-uniform partitions which satisfy certain restrictions on the variation of the mesh.

We then experimentally tested the adaptive FEM schemes with linear elements on a number of scalar and/or system of Conservation Law problems. The problems have been chosen for their well known, in the literature of Conservation Laws, "bad" characteristics (shocks, rarefaction areas, steady shocks). The tests were made on either uniform or *G*-uniform partitions with estimator function the curvature of the numerical solution. From the corresponding results we have concluded to:

- The SRFEM scheme on adaptive *G*-uniform meshes is an efficient solving mechanism for computing solutions for Conservation Laws. Even it contains three stabilization mechanisms (relaxation and spatial redistribution for regularization and resolution purposes, and switching for less dependence on the relaxation parameters) it can be easy implemented in existing RFEM schemes improving this way their behavior in time and in space and relaxing the dependence on the relaxation parameters, with negligible extra computational cost. On every test problem, this scheme has been proved robust and capable to produce high resolution approximations of the entropic solution even in the limit case and without further modifications.
- The generic AGR process has been proved a good machinery for generating partitions of high resolution with low computational cost. From the experiments we are convinced that this process conclude by its own stabilization properties of "upwinding" type. Still, in order to become a stand allown regularization mechanism it needs further improvement and theoretical support.
- It seems that the pure DFEM scheme when applies on adaptive meshes of *G*-uniform type, can be transformed to a robust non-oscillatory entropic scheme.

The development of the SRFEM scheme on adaptive G -uniform meshes is the main goal of this work but, the last two unexpected conclusions are even more interesting for the research on the field of computational Conservation Laws.

Acknowledgments

The author wish to thank Professors Ch. Makridakis and Th. Katsaounis for their helpful discussions. This work 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, *Finite Elements for Hyperbolic Systems of Conservation Laws: New Methods and Computational Techniques*, Ph.D. Thesis, University of Crete, (2002).
- [2] Arvanitis Ch, Katsaounis Th and Makridakis Ch, *Adaptive finite element relaxation schemes for Hyperbolic Conservation Laws*, Math. Model. Numer. Anal., **35**, (2001), 17-33.
- [3] 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., **42**, (2004), 1357–1393.
- [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] Babuška I, *The adaptive finite element method*, TICAM Forum Notes no 7, (1997), University of Texas at Austin
- [6] Babuška I and Gui W *Basic principles of feedback and adaptive approaches in the finite element method*, Comput. Methods Appl. Mech. Engrg. Volume 55, (1986), 27–42
- [7] 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.
- [8] Billingsley P, *Probability and Measure*, JOHN WILEY & SONS Press, 2nd edition (1992).
- [9] Brenner SC and Scott LR, *The Mathematical Theory of Finite Element Methods*, Springer -Verlag, New York, 1994.
- [10] Cockburn B, Johnson C, Shu CW and Tadmor E, *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations*, A. Quarteroni (Ed.), Lecture notes in Mathematics, Volume 1697, Springer-Verlag, (1998).
- [11] Cockburn B, Hou S and Shu CW *The Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. IV. The multidimensional case*, Math. Comp. **54** (1990) 545–581.
- [12] Dafermos CM, *Hyperbolic conservation laws in continuum physics*. Grundlehren der Mathematischen Wissenschaften 325, Springer-Verlag, Berlin, (2000).
- [13] Fazio R, LeVêque RJ, *Moving-Mesh methods for One-Dimensional Hyperbolic Problems Using CLAW-PACK*, Computers and Mathematics with Applications, (2003), **45**:273–298.
- [14] Godlewski E and Raviart PA, *Numerical Approximation of Hyperbolic Systems of Conservation Laws.*, Appl. Math. Series 118, Springer-Verlag, Berlin, (1996).

- [15] Gosse L and Makridakis Ch, *Two a posteriori error estimates for one dimensional scalar conservation laws*, SIAM J. Numer. Anal. **38** , (2000), 964-988 .
- [16] Harten S, Hyman JM, *Self Adjusting Grid Methods for One-Dimensional Hyperbolic Conservation Laws*, Journal of Computational Physics, (1983), **50**:17–33.
- [17] Hou TY and Lax PD *Dispersive approximations in fluid dynamics*, Comm. Pure Appl. Math. **44** (1991), no. 1, 1–40.
- [18] Hyman JM, Li S, and Petzold LR, *An Adaptive Moving Mesh Method with Static Rezoning for Partial Differential Equations*, Computers and Mathematics with Applications, (2003), **46**:1511–1524.
- [19] Jaffré J, Johnson C and Szepessy A, *Convergence of the discontinuous Galerkin finite element method for hyperbolic conservation laws*, Math. Models Methods Appl. Sci. **5** , (1995), 367–386 .
- [20] Jin S and Xin Z *The relaxing schemes for systems of conservation laws in arbitrary space dimensions*, Comm. Pure Appl. Math. **48** , (1995), 235-277 .
- [21] Johnson C and Szepessy A, *On the convergence of a finite element method for a nonlinear hyperbolic conservation law*, Math. Comp. **49**, (1987), 427–444.
- [22] Kurganov A and Tadmor E, *New high resolution central schemes for nonlinear conservation laws and convection-diffusion equations*, J. Comput. Phys. **160** , (2000), 241-282.
- [23] LeVêque RJ, *Finite Volume Methods for Hyperbolic Problems*, Cambridge University Press, 2002.
- [24] LeVêque RJ and Yee HC, *A study of numerical methods for hyperbolic conservation laws with stiff terms* , Comp. Phys. Volume 86, (1990), 187-210.
- [25] Li R, Tang T, and Zhang P, *Moving Mesh Methods in Multiple Dimensions Based on Harmonic Maps*, Journal of Computational Physics, (2001), **170**:562–588.
- [26] Li S, Petzold L, *Moving Mesh methods with upwind schemes for time dependent PDEs*, J. Comp. Phys., (1997), **131**:368–377.
- [27] Li S, Petzold L, and Ren Y, *Stability of Moving Mesh systems of Partial Differential Equations*, SIAM J. Sci. Comput., (1998); **20**:719-738.
- [28] Shu CW, *Total-variation-diminishing time discretizations*, SIAM J. Sc. Comp. **9** , (1988), 1073–1084.
- [29] Shu CW and Osher S, *Efficient implementation of essentially nonoscillatory shock-capturing schemes*, J. Comput. Phys. **77** , (1988), 439–471.
- [30] Stockie JM, Mackenzie JA, Russell RD, *A Moving mesh method for one-dimensional hyperbolic conservation laws*, SIAM J. Sci. Comput., (2001); **22**:1791-1813.
- [31] Süli E, *A-posteriori error analysis and adaptivity for finite element approximations of hyperbolic problems*, Lecture notes in Computational Science and Engineering. Volume 5, pages 123-194 Springer-Verlag, (1998).
- [32] Tan Z, Zhang Z, Huang Y, Tang T. *Moving mesh methods with locally varying time steps*, Journal of Computational Physics, (2004), **35**:17–33.
- [33] Tang H. *Solution of the shallow-water equations using an adaptive moving mesh method*, Int. J. Numer. Meth. Fluids, (2004), **44**:789-810.
- [34] Tang H, Tang T. *Adaptive mesh methods for one- and two-dimensional hyperbolic conservation laws*, SIAM J. Numer. Anal., (2003), **41**: 487–515.
- [35] Tzavaras A, *Viscosity and relaxation approximation for hyperbolic systems of conservation laws*, Lecture notes in Computational Science and Engineering. Volume 5, pages 73-122 Springer-Verlag, (1998).

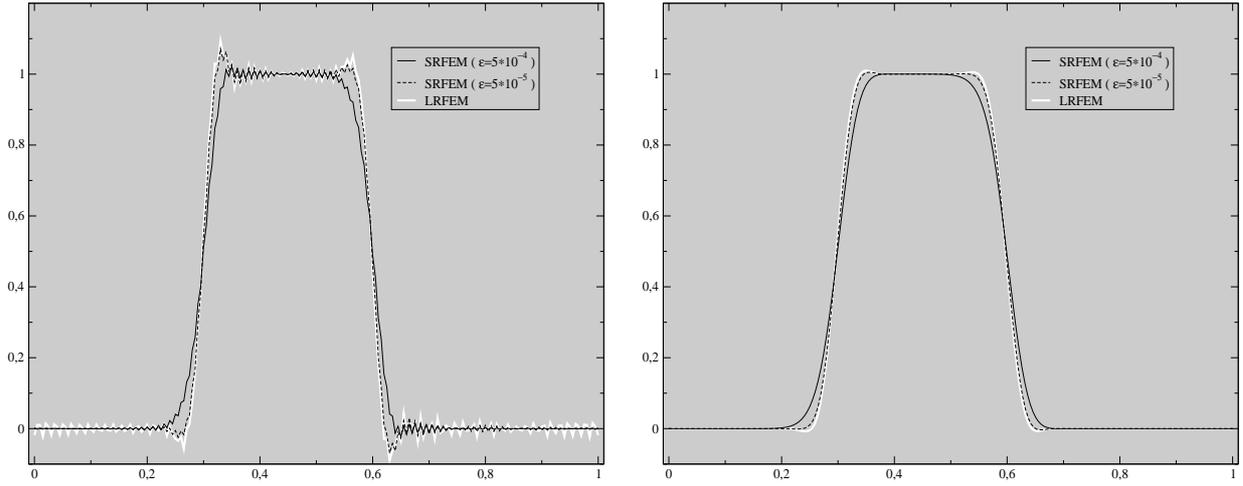


Figure 4: Advection problem ($t = 2$), SRFEM scheme as ε shrinks and LRFEM scheme on uniform partition (left) and on G -uniform partition for $p = 0.0043$ (right).

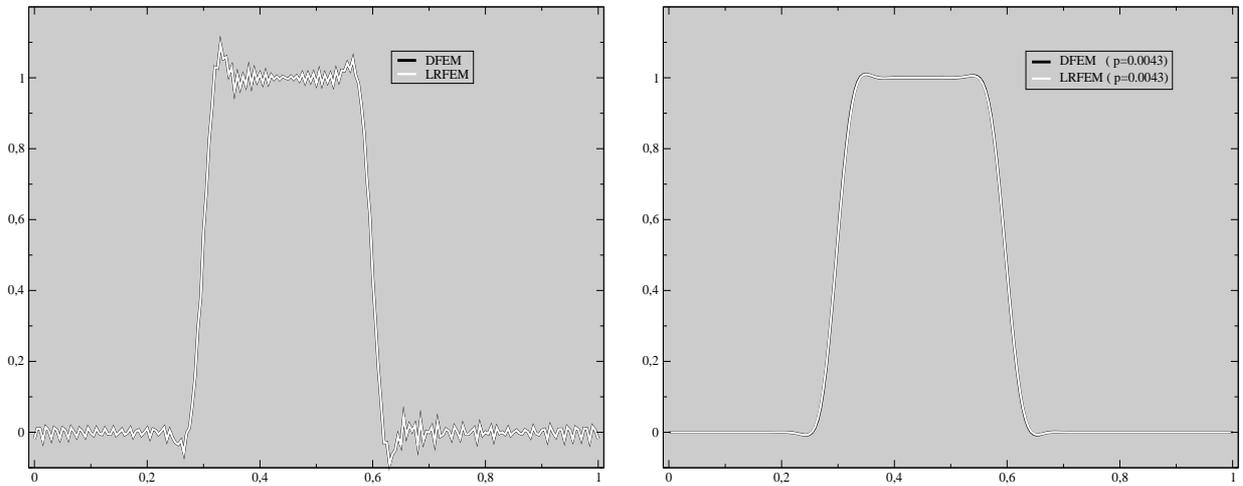


Figure 5: Advection problem ($t = 2$), LRFEM and DFEM scheme on uniform partition (left) and on G -uniform partition (right).

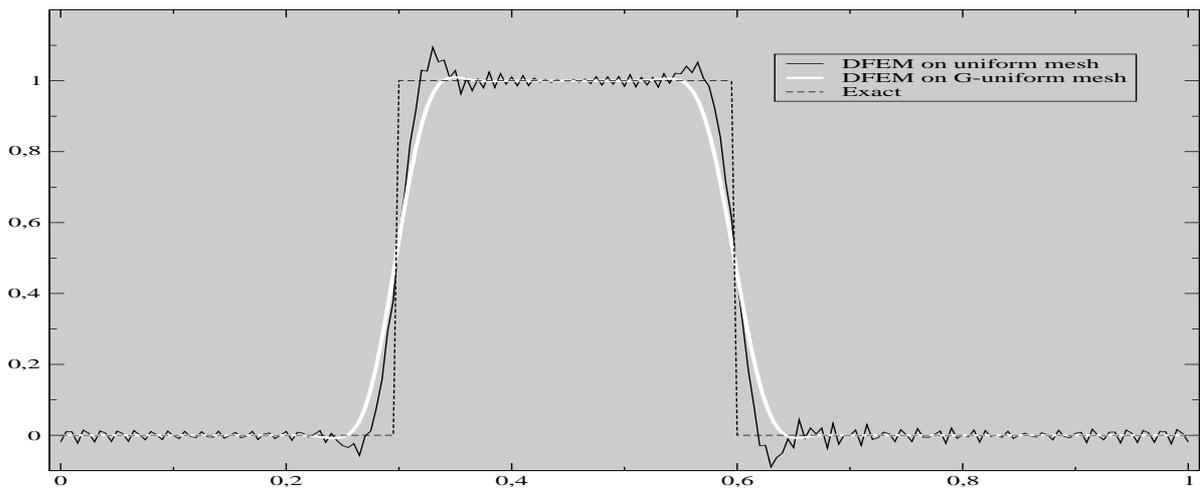


Figure 6: Advection problem ($t = 2$), DFEM scheme on uniform partition and on G -uniform partition ($p = 0.0043$) and the exact solution.

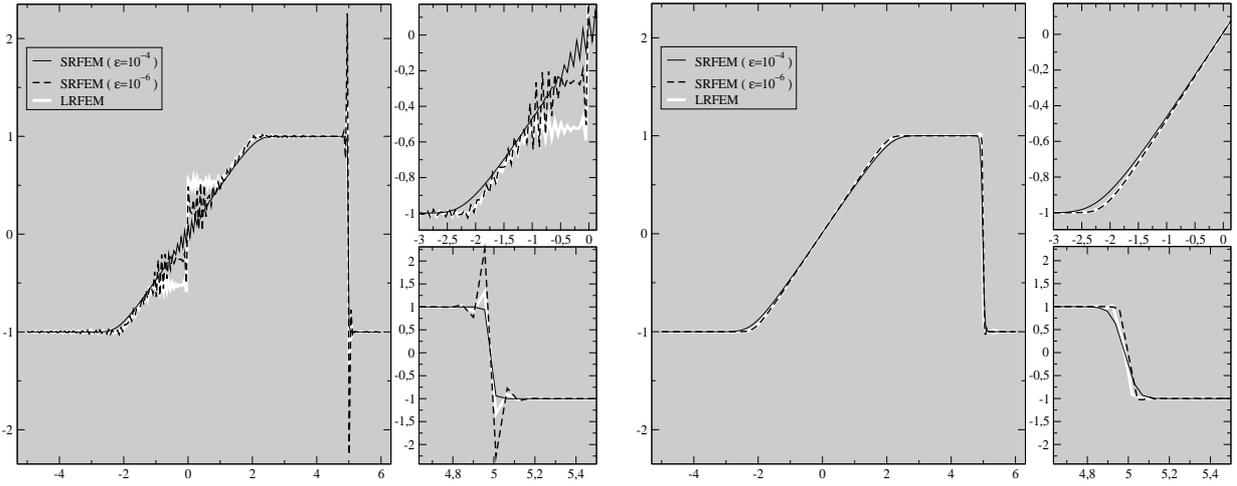


Figure 7: Burgers problem ($t = 0.28$), SRFEM scheme as ε shrinks and LRFEM scheme on, uniform partition (left) and G-uniform with $p = 0.035$ (right).

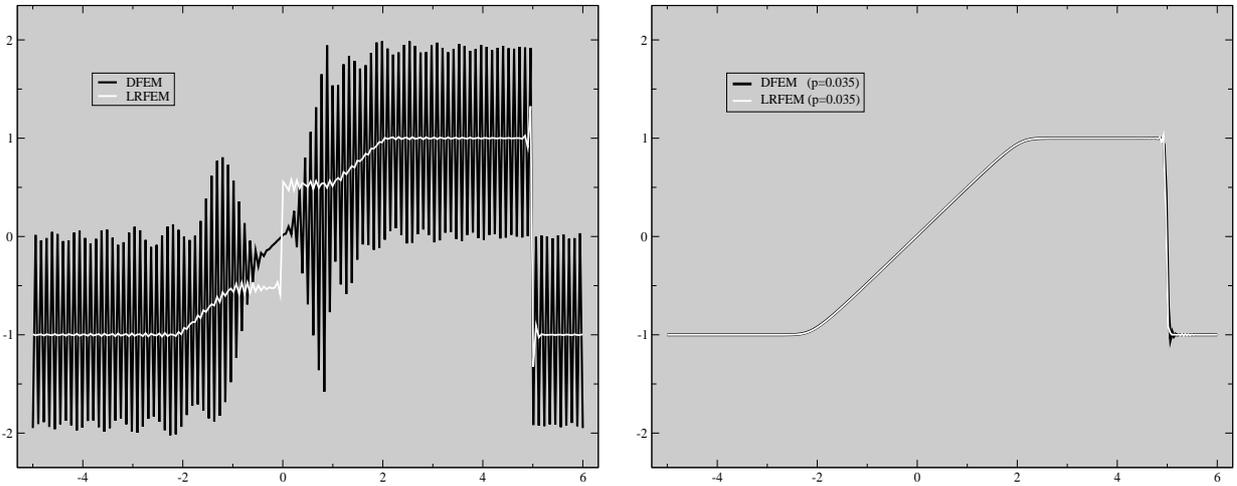


Figure 8: Burgers problem ($t = 0.28$), LRFEM and DFEM scheme on uniform partition (left) and on G-uniform partition (right).

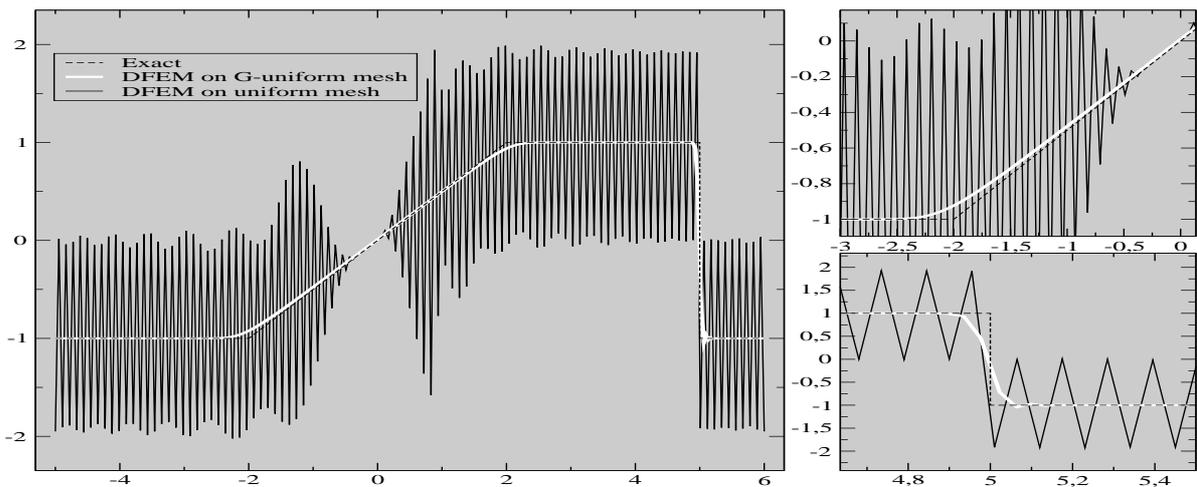


Figure 9: Burgers problem ($t = 0.28$), DFEM scheme on uniform partition and on G-uniform partition ($p = 0.035$) and the exact solution.

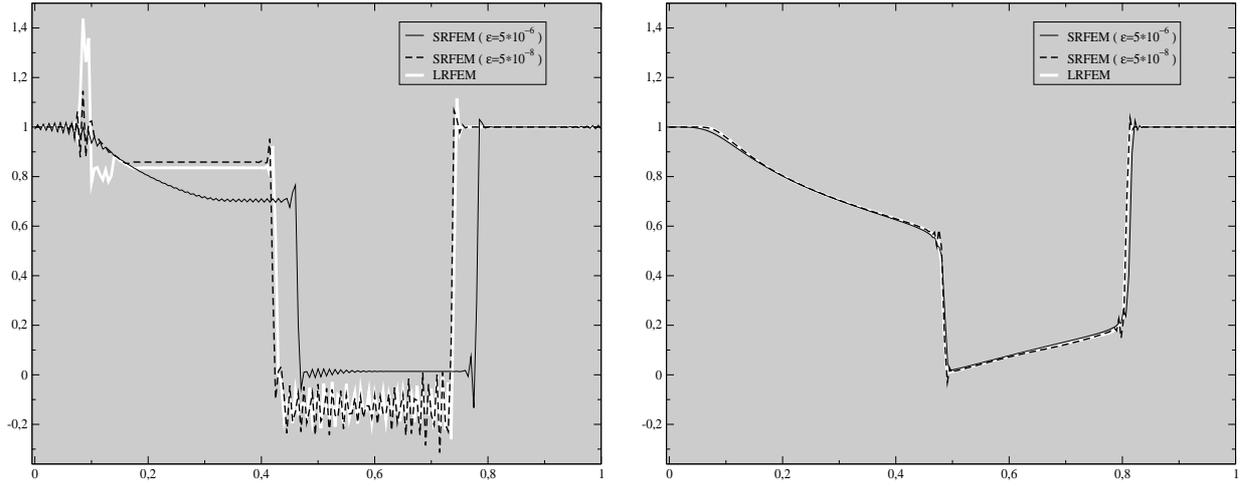


Figure 10: Buckley Leverett problem ($t = 0.28$), LRFEM and DFEM scheme on uniform partition (left) and on G-uniform partition for $p = 0.035$ (right).

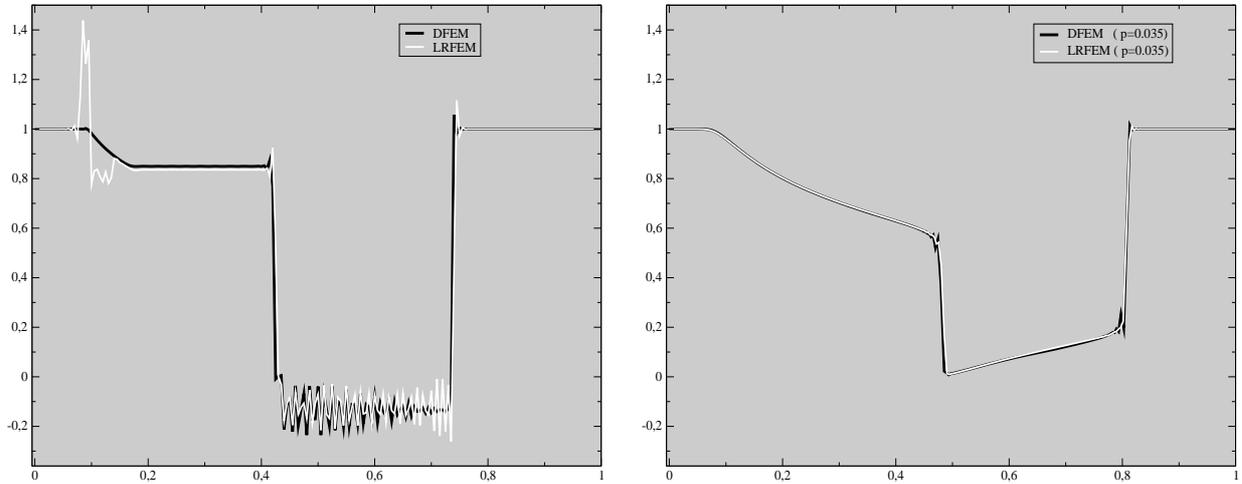


Figure 11: Buckley Leverett problem ($t = 0.28$), LRFEM and DFEM scheme on uniform partition (left) and on G-uniform partition (right).

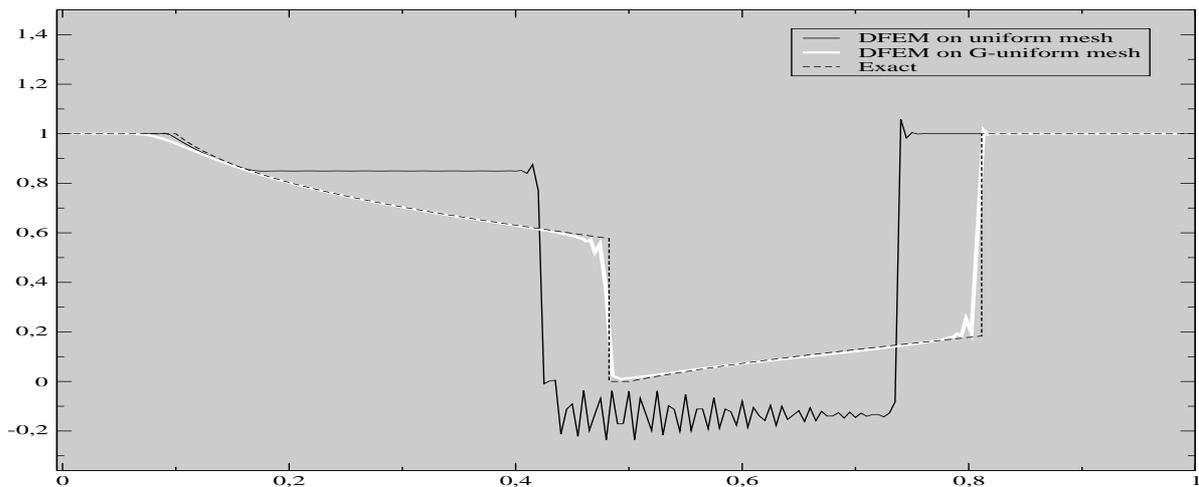


Figure 12: Buckley Leverett problem ($t = 0.28$), DFEM scheme on uniform and G-uniform ($p = 0.035$) partitions and the exact solution.

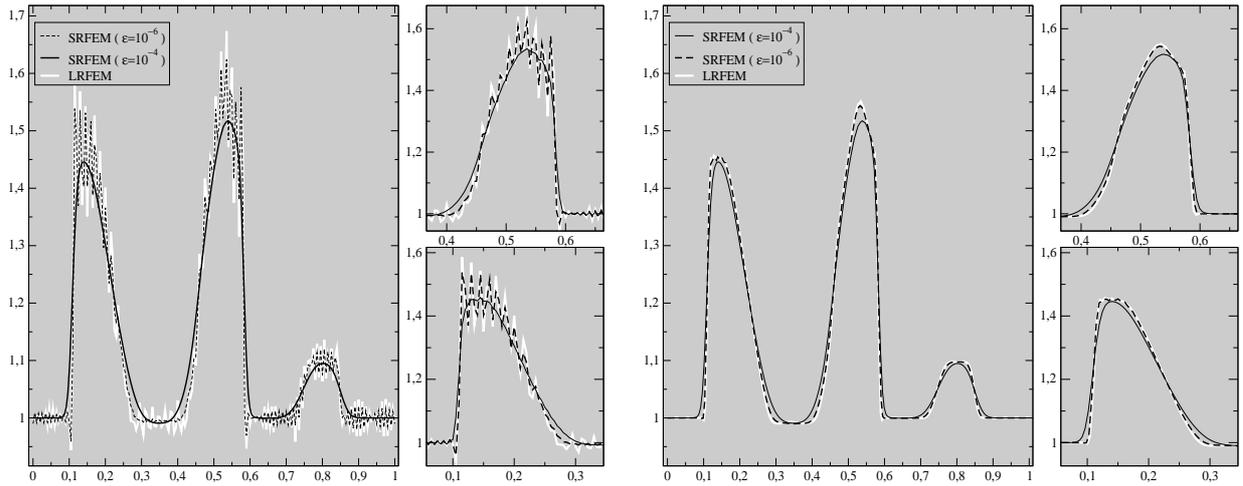


Figure 13: 1th component of the Shallow water problem ($t = 0.055$), SRFEM scheme as ε shrinks and LRFEM scheme on uniform partition (left) and on G-uniform partition with $p = 0.035$ (right).

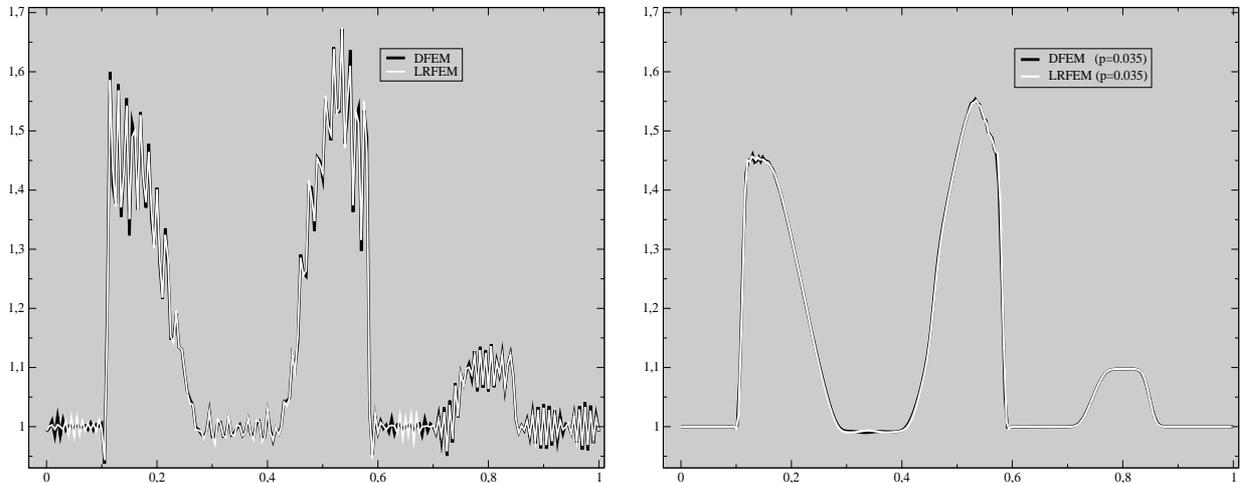


Figure 14: 1th component of the Shallow water problem ($t = 0.055$), LRFEM and DFEM scheme on uniform partition (left) and on G-uniform partition (right).

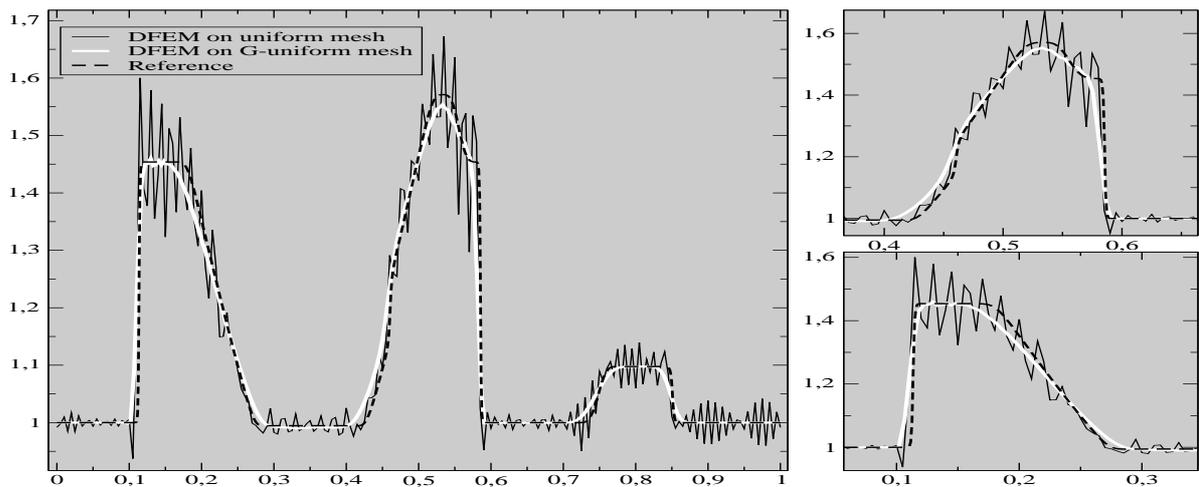


Figure 15: 1th component of the Shallow water problem ($t = 0.055$), DFEM scheme on uniform and G-uniform ($p = 0.035$) partitions and the reference solution.

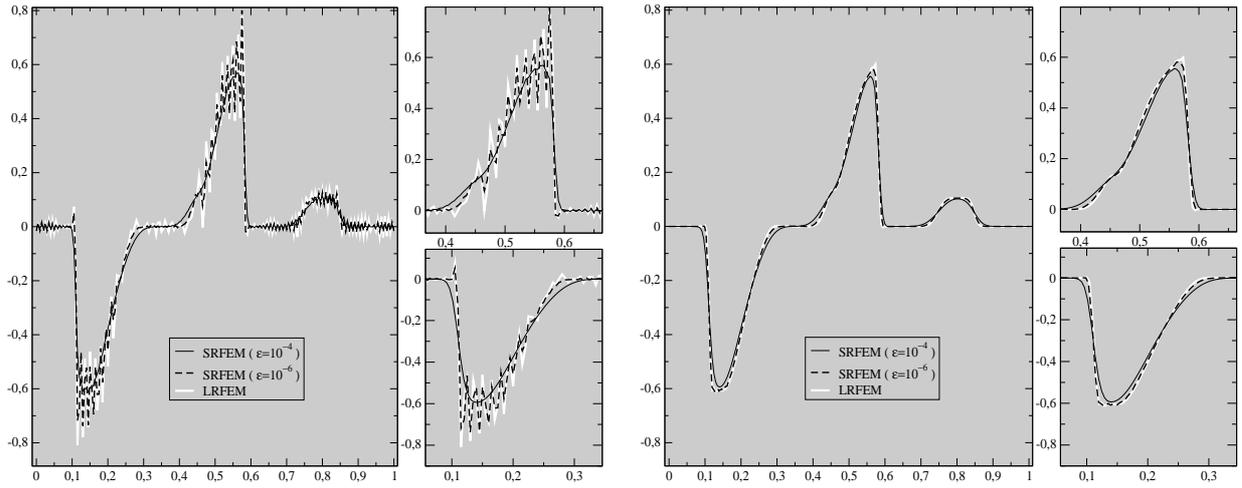


Figure 16: 2 th component of the Shallow water problem ($t = 0.055$), SRFEM scheme as ε shrinks and LRFEM scheme on uniform partition (left) and on G-uniform with $p = 0.035$ (right).

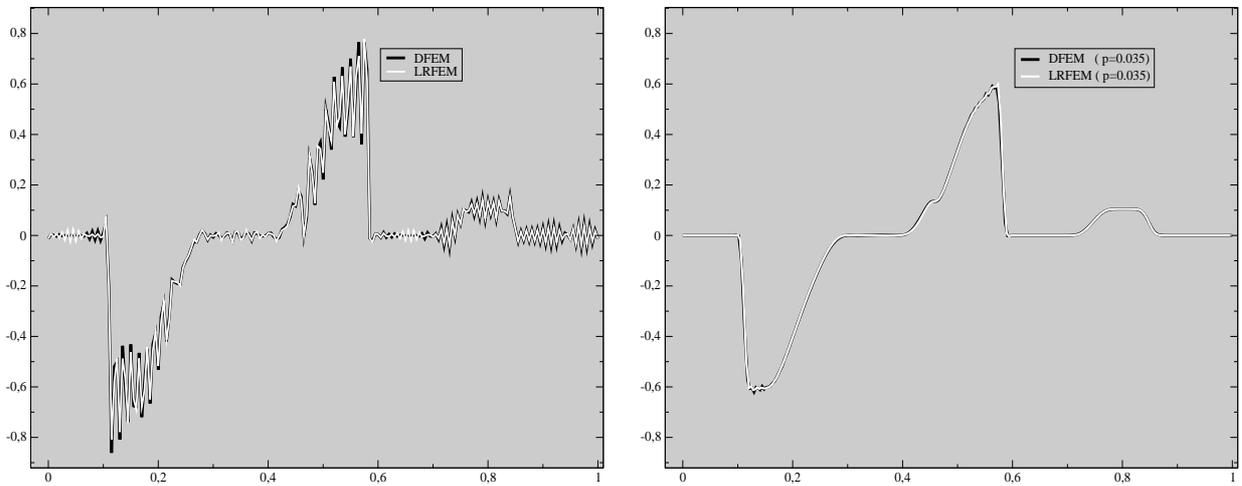


Figure 17: 2th component of the Shallow water problem ($t = 0.055$), LRFEM and DFEM scheme on uniform partition (left) and on G-uniform partition (right).

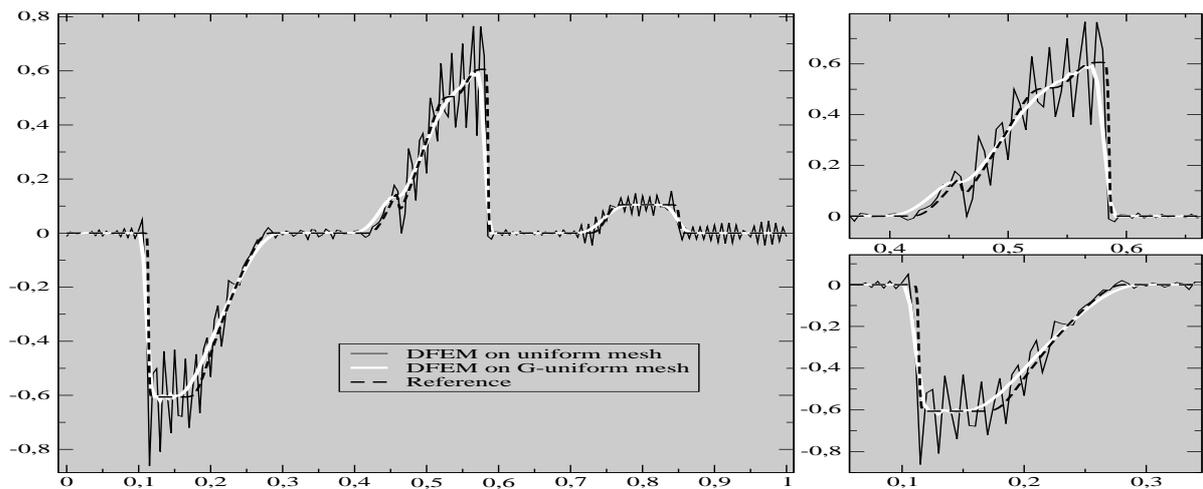


Figure 18: 2th component of the Shallow water problem ($t = 0.055$), DFEM scheme on uniform partition and G-uniform partition ($p = 0.035$) and the reference solution.