J Sci Comput (2008) 34: 1–25
DOI 10.1007/s10915-007-9155-7

Mesh Redistribution Strategies and Finite Element Schemes for Hyperbolic Conservation Laws

Christos Arvanitis

Received: 1 December 2006 / Revised: 17 June 2007 / Accepted: 28 August 2007 /
Published online: 21 September 2007

Abstract In this work we consider a new class of Relaxation Finite Element schemes for hyperbolic conservation laws, with more stable behavior on the limit area of the relaxation parameter. Combining 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 parameter where the scheme lacks of the relaxation mechanism. Thus we experimentally conclude that the proposed spatial redistribution process, has by its own interesting stabilization properties for computational solutions of conservation law problems.

Keywords Finite element methods · Relaxation model · Adaptive mesh 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 (HSCL); *find* $u : \mathbb{R}^d \times [0, T] \to \mathbb{R}^M$ *such that*

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

C. Arvanitis (✉)
Department of Mathematics and Statistics, University of Cyprus, 1678 Nicosia-Cyprus, Greece
e-mail: arvas@ucy.ac.cy

C. Arvanitis
Institute of Applied and Computational Mathematics, FORTH, 71110 Heraklion, Crete, Greece

Finite Element (FE) methods was not a very popular choice for computing solutions of (1.1). When applied directly to this system, the method will result in 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 HSCL was an important issue since the works of von Neumann, see the very interesting work by Lax and Hou [14] 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 [9, 10, 16, 18]. 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 was introduced in [1, 2]. These methods are based on relaxation models, they do not include additional stabilization terms other than relaxation, but alternatively are designed to be used in conjunction with appropriate mesh refinement.

Mesh adaptation is a main current stream towards the efficient numerical solution of complex systems. In the 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, etc.) in adaptive finite element literature and software implementation is at a remarkable level. In this paper we focus on the behavior of finite element schemes for HSCL when combined with adaptive meshes. In particular, our aim in this work is

- To propose and experimentally study new finite element schemes for HSCL.
- 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 adaptively evolving mesh.

New Finite Element Schemes. Our schemes are based on the idea of Relaxation Finite Element schemes of [1] but are further developed. The new schemes consist an extremely robust class of finite element methods for HSCL. The idea of the schemes in [1, 2] was to use the simple and very appropriate for computational purposes model of Jin and Xin [17] to construct finite element schemes without additional stabilization mechanisms. In this work we suggest a modification of these schemes that computationally decouples the action of relaxation parameter ε and the time step size. The resulting scheme, called *Switched Relaxation Finite Element*, shows remarkable stability even for extremely small values of ε. As a natural extension then we consider the limiting case $\varepsilon = 0$. The resulting scheme is called *Limit Relaxation Finite Element*. This scheme although connected to the *Relaxed* difference schemes of Jin and Xin [17] has a major difference. Namely, in the case of Relaxed schemes the main stabilization mechanism is of "upwinding" type (inherited by the upwind discretization of the linear part of the relaxation system), while in our case the Limit Relaxation Finite Element scheme *is 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.

Adaptive Mesh Redistribution. The adaptive strategy studied in this work is based on 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 FE space corresponding 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. A crucial point is the choice of appropriate estimator functions that are used to redefine the mesh of the space domain. The corresponding partitions are called *G-uniform* since, the Riemann-Stieltjes measure G, induced from the estimator g, is uniformly distributed on them. We use two different estimator functions that we consider appropriate for approximating HSCL. One is motivated by available a posteriori estimates in the scalar case, [1, 12]. 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 dimensions. 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.

The adaptive mesh redistribution stabilizes Finite Element Dispersive Schemes. A main result in this paper is the rather surprising computational behavior of finite element schemes with very little or no 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 finite elements discretization of conservation laws, that is known to produce approximate solutions with spurious oscillations, when combined with the mesh adaptation yields stable solutions free of oscillations.

We present here a few numerical results from a series of numerical experiments we have performed; our main conclusions are:

- Switched Relaxation Finite Element schemes consist a robust class of schemes. Like every class of Finite Element schemes they allow to use high order as well as lower order elements and can be applied on uniform or non-uniform meshes without modifications. Because of the relaxation mechanism, it can be applied immediately on HSCL problems without the need of additional stabilization terms. Moreover the relaxation parameter ε can take even negligible positive values without affecting the time step size.
- 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, 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 schemes on uniform mesh seem to approximate the non classical shock, while, when mesh adaptation is used, they approximate the entropic solution.

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 Sect. 4 we present the numerical tests using the above schemes and finally, we highlight some conclusions in Sect. 5.

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 [17]:

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

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

corresponds to the regularization of each component of (1.1) by a wave operator of order ε, see [1]. Here the *relaxation characteristics* C_i are symmetric, positive definite matrices of $\mathbb{R}^{M,M}$ that are selected to satisfy certain stability conditions, *the subcharacteristic conditions*, see [1, 2, 17, 30]. Thus the relaxation model induces a regularization mechanism with *finite speed of propagation* and results in 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 and $\mathcal{T} = \{K_i\}_{i=1}^N$ is a partition of Ω into N elements, with the usual properties [8] related to the spatial size $h = \min_{K\in\mathcal{T}} \operatorname{diam}(K)$. We will use the following notation:

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

$$S_{r,\mathcal{T}} = \{\phi \in C(\Omega) : \phi\,|_K \in \mathbb{P}_r(K),\, K \in \mathcal{T}\},$$

  for approximating scalar functions defined on the spatial domain. In the case of vector functions, we will use the Cartesian power $S_{r,\mathcal{T}}^M$ of $S_{r,\mathcal{T}}$ space, thus, all the coordinates of the numerical solution are approximations given from the same space.
- With $\Pi_{S_{r,\mathcal{T}}}$ we denote the interpolant operator on $S_{r,\mathcal{T}}$, i.e., for any function u defined on the domain, $\Pi_{S_{r,\mathcal{T}}}(u)$ is an element of the space $S_{r,\mathcal{T}}$ which interpolates u (and/or its derivatives), on some set of spatial points defined through the elements of the partition $\mathcal{T}$. In the case of linear Finite Elements spaces, the element $\Pi_{S_{1,\mathcal{T}}}(u)$ is defined by interpolating u on the node points $\{x_i\}_{i=0}^N$ of the partition. For some vector function $u = (u_1,\ldots,u_M)^{\mathrm{T}} : \Omega \to \mathbb{R}^M$ we denote by $\Pi_{S_{r,\mathcal{T}}^M}(u)$ the interpolant element from the Cartesian power space $S_{r,\mathcal{T}}^M$, i.e., $\Pi_{S_{r,\mathcal{T}}^M}(u) = (\Pi_{S_{r,\mathcal{T}}}(u_1),\ldots,\Pi_{S_{r,\mathcal{T}}}(u_M))^{\mathrm{T}}$.
- For any function u on Ω we shall denote by $u_{\mathcal{T}}$ the approximation element from the corresponding FE space to the partition $\mathcal{T}$.

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

- The Galerkin discretization directly to the Conservation Law (1.1): Seek $u_{\mathcal{T}} \in S^M_{r,\mathcal{T}} \times [0,T]$ such that $u_{\mathcal{T}}(\cdot,0) = \Pi_{S^M_{r,\mathcal{T}}}(u_0)$, and for $t \in (0,T]$,

$$(\partial_t u_{\mathcal{T}}, \phi) + \sum_{i=1}^{d} (-F_i(u_{\mathcal{T}}), \partial_{x_i}\phi) = 0, \quad \forall \phi \in S^M_{r,\mathcal{T}}. \tag{2.2}$$

- The Galerkin discretization of the Relaxation model of Conservation Law (2.1): Seek $u_{\mathcal{T}}, v_{\mathcal{T},1}, \ldots, v_{\mathcal{T},d} \in S^M_{r,\mathcal{T}} \times [0,T]$ such that $u_{\mathcal{T}}(\cdot,0) = \Pi_{S^M_{r,\mathcal{T}}}(u_0)$, $v_{\mathcal{T},i}(\cdot,0) = \Pi_{S^M_{r,\mathcal{T}}}(F_i(u_0))$, $i = 1,\ldots,d$, and for $t \in (0,T]$, *for all* $\phi \in S^M_{r,\mathcal{T}}$,

$$\begin{aligned} &(\partial_t u_{\mathcal{T}}, \phi) + \sum_{i=1}^{d} (-v_{\mathcal{T},i}, \partial_{x_i}\phi) = 0, \\ &(\partial_t v_{\mathcal{T},i}, \phi) + (C_i \cdot \partial_{x_i} u_{\mathcal{T}}, \phi) = -\frac{1}{\varepsilon}(v_{\mathcal{T},i} - F_i(u_{\mathcal{T}}), \phi), \quad i = 1,\ldots,d. \end{aligned} \tag{2.3}$$

2.2 Fully Discrete Schemes

Our finite element schemes are appropriate fully discrete versions of (2.3) and (2.2). 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 are interested in approximating the instances $u(\cdot,t^n)$ of the solution u at moments $t^n = n \cdot \kappa$, $n = 0,1,\ldots$.

2.2.1 The Relaxation Finite Element Scheme

We start with the fully discrete Relaxation Finite Element (RFE) scheme, presented in [1]. As it is shown there, a choice that results in to a linear fully discrete scheme well suited to the structure of (2.3), is a combination of an explicit and a diagonally implicit Runge Kutta discretization in time. In order to decouple the system (2.3), we discretize the first equation by using an Explicit RK (ERK) method while for the second we use a Diagonally Implicit RK (DIRK) method.

Assuming that the *Relaxation Finite Element* approximation $(u^n_{\mathcal{T}}, v^n_{\mathcal{T}})$ of the solutions instance $(u(\cdot,t^n),\ v(\cdot,t^n))$ of (2.3) has been computed at moment t^n, then the approximation $(u^{n+1}_{\mathcal{T}}, v^{n+1}_{\mathcal{T}})$ of the solution at the next moment t^{n+1} is defined such that *for every* ϕ *in* $S^M_{r,\mathcal{T}}$:

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

where the intermediate stages $(u^{n,i}_{\mathcal{T}}, v^{n,i}_{\mathcal{T}})$, $i = 1,\ldots,q$, *are given by the following coupled systems of* q*-equations*:

$$\begin{aligned} (u^{n,i}_{\mathcal{T}}, \phi) &= (u^n_{\mathcal{T}}, \phi) + \kappa \sum_{j=1}^{i-1} a_{ij} (-\partial_x v^{n,j}_{\mathcal{T}}, \phi), \\ (v^{n,i}_{\mathcal{T}}, \phi) &= (v^n_{\mathcal{T}}, \phi) + \kappa \sum_{j=1}^{i} \tilde{a}_{ij} \left\{ (-C \cdot \partial_x u^{n,j}_{\mathcal{T}}, \phi) - \frac{1}{\varepsilon}(v^{n,j}_{\mathcal{T}} - F(u^{n,j}_{\mathcal{T}}), \phi) \right\}, \end{aligned} \tag{2.5}$$

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

In our experiments we use the following ERK methods proposed in [25], [24], which are of second and third order, respectively,

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

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

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

The intermediate stages $(u_{\mathcal{T}}^{n,i}, v_{\mathcal{T}}^{n,i}), i = 1, \ldots, q$, 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 [1] the RFE schemes have the following main characteristics:

- Like every FE scheme they can be defined on non-uniform mesh without modifications, so they can be combined with mesh redistribution.
- They show a remarkable robustness, in terms of the order of polynomials used in the finite element space.
- The systems (2.4, 2.5) are easy to solve, since they are explicit except the second system of (2.5) which is semi-implicit, and the corresponding mass matrix to finite elements methods is of band type. In our experiments we solve the systems using the band LU decomposition of the mass matrix which, related to the cardinality N of the partition $\mathcal{T}$, is a task of linear complexity. In particular for one dimensional domains, the computational cost of the evolution step is $(K_1 \cdot r^2 + K_2 \cdot M \cdot q) \cdot N$ flops, where K_1, K_2 are some constants, and r is the band width which is equal to the order of the used polynomials.
- They converge to the entropic solution without the need of additional stabilization terms, as long as $\varepsilon, h \to 0$, and the discretization steps h, κ satisfying the modified CFL stability condition

$$\kappa < C_1 \varepsilon < C_2 h, \tag{2.8}$$

where the constants C_1, C_2 depends on the flux F of the HSCL (see also [2]).

2.2.2 *The Switched Relaxation Finite Element Scheme*

Numerical experiments confirm the stability condition (2.8). Indeed, in many cases in order to get the RFE approximation on a fixed partition for smaller ε the time step has to be severely reduced (see tables in the numerical experiments section).

To overcome the difficulty and since we are interested in investigating the behavior of the RFE scheme for very small ε, we introduce the *Switched Relaxation Finite Element* (SRFE) scheme; a proper version of the RFE scheme with the difference that every evolution step begins with the projection of the relaxation variables $v_{\mathcal{T},i}$, $i = 1, \ldots, d$ to the equilibrium manifold. The motivation behind the switched scheme is based on the observation:

The relaxation variables v_i, $i = 1, \ldots, d$ of the model (2.1), which account for the modification of CFL *condition (see [2]), initially as well as in the limit ($\varepsilon = 0$), take the values $v_i = F_i(u)$, $i = 1, \ldots, d$.*

The proposed switched scheme applies also these settings at the beginning of every evolution step, but not at the intermediate stages of the RK method. The choice of the name is due to the fact that for the setting $v_i = F(u_i)$, the relaxation model (2.1) reduces to the HSCL (1.1), thus, the above process *switches* instantaneous the solving method from the RFE (2.3) to the direct FE (2.2) approximation. We proceed to the detailed definition of the SRFE scheme. Using the same RK timestep notation, let $\mathcal{T}$ be a uniform partition of Ω, and $u^0_{\mathcal{T}} = \Pi_{S^M_{r,\mathcal{T}}}(u_0)$. Assuming that the *Switched Relaxation Finite Element* approximation $u^n_{\mathcal{T}}$ of the solutions instance $u(\cdot, t^n)$ of (2.3) has been computed at moment t^n, then, the approximation $u^{n+1}_{\mathcal{T}}$ at the next moment t^{n+1} is defined such that, *for every ϕ in $S^M_{r,\mathcal{T}}$, there holds*:

$$(u^{n+1}_{\mathcal{T}}, \phi) = (u^n_{\mathcal{T}}, \phi) + \kappa \sum_{i=1}^{q} b_i(-\partial_x v^{n,i}_{\mathcal{T}}, \phi), \tag{2.9}$$

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

$$\begin{aligned}
(u^{n,i}_{\mathcal{T}}, \phi) &= (u^n_{\mathcal{T}}, \phi) + \kappa \sum_{j=1}^{i-1} a_{ij}(-\partial_x v^{n,j}_{\mathcal{T}}, \phi), \\
(v^{n,i}_{\mathcal{T}}, \phi) &= (\Pi_{S^M_{r,\mathcal{T}}}(F(u^n_{\mathcal{T}})), \phi) \\
&\quad + \kappa \sum_{j=1}^{i} \tilde{a}_{ij} \left\{ (-C \cdot \partial_x u^{n,j}_{\mathcal{T}}, \phi) - \frac{1}{\varepsilon}(v^{n,j}_{\mathcal{T}} - F(u^{n,j}_{\mathcal{T}}), \phi) \right\}.
\end{aligned} \tag{2.10}$$

Note that at every time step, in the case of RFE schemes the solution $v^{n+1}_{\mathcal{T}}$ of the second explicit-equation in (2.4) is required, while in the SRFE case the interpolant of $F(u^n_{\mathcal{T}})$ must be calculated, since the integration in time starts with $F(u^n_{\mathcal{T}})$ instead of $v^n_{\mathcal{T}}$. Thus the computational cost of the SRFE and RFE schemes is of the same complexity.

These schemes approximate the weak solution of a Relaxation type model where the relaxation parameter ε is temporally variable, i.e., $\varepsilon(t) = 0$ on the moments t^n, $n = 0, 1, \ldots$, otherwise $\varepsilon(t) = \varepsilon_0$ some constant. It is an interesting question whether the family $\{u^{\varepsilon(t)}_{\mathcal{T}}\}$ is weakly convergent to the solution of the corresponding HSCL when ε_0 tends zero. In practice, the SRFE scheme gives bounded approximations even for negligible ε_0 without affecting the time step, but in the regime of the proper CFL stability condition for the HSCL, i.e., $\kappa < C_2 h$. Numerical experiments show that on uniform partitions:

- The parameter ε in the SRFE scheme can be taken arbitrarily small, with no further change to the time step κ (see tables in the numerical experiments section).
- For $\varepsilon > 10^{-5}$ the time switch does not influence the relaxation mechanism and so in that range the given approximations from the SRFE schemes coincide with those from the RFE scheme.

2.2.3 The Limit Relaxation Finite Element and the Direct Finite Element Schemes

Motivated by the computational behavior of the SRFE scheme close to the limit, we introduce also the *Limit Relaxation Finite Element* (LRFE) scheme which is derived by taking

the limit of (2.10) as $\varepsilon \to 0$. Thus the evolution step from the LRFE 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* $S_{r,\mathcal{T}}^M$:

$$(u_{\mathcal{T}}^{n+1}, \phi) = (u_{\mathcal{T}}^n, \phi) + \kappa \sum_{i=1}^{q} b_i(-\partial_x v_{\mathcal{T}}^{n,i}, \phi), \tag{2.11}$$

with the intermediate stages $(u_{\mathcal{T}}^{n,i}, v_{\mathcal{T}}^{n,i})$, $i = 1, \ldots, q$, *given by the following system of* q*-equations*:

$$\begin{aligned} &(u_{\mathcal{T}}^{n,i}, \phi) = (u_{\mathcal{T}}^n, \phi) + \kappa \sum_{j=1}^{i-1} a_{ij}(-\partial_x v_{\mathcal{T}}^{n,j}, \phi), \\ &(v_{\mathcal{T}}^{n,i} - F(u_{\mathcal{T}}^{n,i}), \phi) = 0. \end{aligned} \tag{2.12}$$

Notice that this scheme is independent from the relaxation parameter ε, so 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 evidences for the behavior of the SRFE scheme on the limit $\varepsilon = 0$.

One may observe that if we apply the switched technique on the LRFE scheme at the intermediate time levels (2.12) of the RK-method, i.e. setting $v_{\mathcal{T}}^{n,i} = F(u_{\mathcal{T}}^{n,i})$, $i = 1, 2, \ldots, q$, the LRFE scheme is transformed to the fully discrete scheme of the *Direct Finite Element* (DFE) approximation (2.2) for the HSCL, defined by:

$$(u_{\mathcal{T}}^{n+1}, \phi) = (u_{\mathcal{T}}^n, \phi) + \kappa \sum_{i=1}^{q} b_i(F(u_{\mathcal{T}}^{n,i}), \partial_x \phi), \tag{2.13}$$

where the intermediate stages $u_{\mathcal{T}}^{n,i}$, $i = 1, \ldots, q$, *are given by the following* q*-equations*:

$$(u_{\mathcal{T}}^{n,i}, \phi) = (u_{\mathcal{T}}^n, \phi) + \kappa \sum_{j=1}^{i-1} a_{ij}(F(u_{\mathcal{T}}^{n,j}), \partial_x \phi). \tag{2.14}$$

Numerical experiments on uniform partitions, show that:

- While $\varepsilon \to 0$ the given approximations from the SRFE scheme tend to the approximation from the LRFE scheme.
- In certain cases the given approximations from the LRFE and DFE schemes are different (see the Burgers experiment).

For the LFE scheme (2.11, 2.12) the reduction of the cost compared to the RFE scheme is significant. 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 also is a task of linear complexity but with smaller multiplying constant. Finally, in the DFE scheme (2.13, 2.14) all the systems are explicit without the inserted relaxation variables $v_{\mathcal{T}}^n$, thus the amount of calculations is the lowest possible.

3 Adaptive Mesh Redistribution

Adaptive mesh redistribution is a main current stream towards efficiently computing numerical solutions of high resolution. Several redistribution techniques have been recently

introduced for proper mesh selection, starting with the "One shot method" of Babuška and the "Self-Adjusting method" of Harten, Hyman to the "Moving Mesh" methods of LeVêque, Russell, Tao Tang and others (see bibliography [3–6, 11, 13, 15, 20–22, 26–29, 32], and [31] for a complete survey on the "spatial rezoning" and its contribution to moving mesh methods). These methods calculate the spatial positions of the nodes of the new mesh, some of them by solving an Euler-Lagrange equation, while others by optimizing proper energy metrics. An interesting version based on the Lagrangian form of HSCL can be found in [23].

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 [1] and is related although different to the "one shot method" of Babuška [4, 5]. Here 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.

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

- For some positive integer N, we denote with $\mathrm{Part}_N(\Omega)$ the family of partitions of Ω into N elementary sets K, i.e.

$$\mathrm{Part}_N(\Omega) = \{\mathcal{T} \subset \wp(\Omega) : \mathcal{T} \text{ partition of } \Omega, \mathrm{card}(\mathcal{T}) = N\},$$

  where $\wp(\Omega)$ denotes the powerset of Ω, i.e., the set of all subsets of Ω.
- $\sigma(\mathcal{T})$ denotes the σ-algebra generated by the partition $\mathcal{T}$; for finite partitions, the corresponding σ-algebra coincide with the set containing unions of partition elements, i.e., if $\mathcal{T} = \{K_i\}_{i=1}^N$ then

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

  where by K_i^1, K_i^0 we denote the set K_i and the empty set $\emptyset$ respectively.
- With $\mathrm{resolution}_{\mathcal{T}}$ we shall denote a measure which for any proper subset $A \subset \Omega$ is defined by

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

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

3.1 Mesh Redistribution Policy

In general the following observation holds true for numerical calculations based on some partition T of the domain:

> *The accuracy of any numerical approximation over some subset A of the domain is an increasing function of the* $\mathrm{resolution}_{\mathcal{T}}(A)$, *which in turn must be bounded from above* (*e.g. in the case of* HSCL *the* CFL *condition imposes an upper bound on* $\mathrm{resolution}_{\mathcal{T}}(A)$ *for stability reasons*).

This observation motivates our redistribution policy:

> *Seek a partition* $\tilde{\mathcal{T}}$ *with* $\mathrm{resolution}_{\tilde{\mathcal{T}}}$ *so that, over any area of the domain, it must be 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. 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. Construct the new partition by properly using the estimator function g. Let G be the measure of density g with respect to the Lebesgue measure μ, i.e., the G measure of a Lebesgue measurable set $A \subset \Omega$ is

$$G(A) = \int_A g \, d\mu. \tag{3.1}$$

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 policy 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$_{\tilde{\mathcal{T}}}(A)$ *must be proportional to the* $G(A)$ *measure, i.e., for some constant* C *it holds*

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

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 $K \in \tilde{\mathcal{T}}$ (for which resolution$_{\tilde{\mathcal{T}}}(K) = 1$) we conclude from (3.2) that:

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

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

Definition 1 *In sequel, a finite partition* $\mathcal{T} = \{K\} \in \text{Part}_N(\Omega)$ *shall be called* G*-uniform related to a given estimator function* g*, if the corresponding measure* $G : \sigma(\mathcal{T}) \to \mathbb{R}$ *s.t.* $G(A) = \int_A g \, d\mu$ *is uniformly distributed on* $\sigma(\mathcal{T})$*, i.e.,* $\forall K \in \mathcal{T} \, G(K) = G(\Omega)/N$*. In such a case the set* $\bigcup_{K \in \mathcal{T}} \partial K$ *shall be called a* G*-uniform mesh related to the estimator function* g*.*

Let us note some properties of G*-uniform partitions.*

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

$$\forall K \in \mathcal{T} \quad C \le \max_{x \in K}\{g(x)\}\mu(K) \le \max_{x \in K}\{g(x)\}\text{diam}(K)^d, \tag{3.4}$$

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

(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* $\mathcal{T} = \{K\} \in \text{Part}_N(\Omega)$:

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

Thus, in the case where $\mathcal{T}$ *is a* G*-uniform partition then by* (3.3) *the above relation holds as equality, so this class of partitions solves the min-max problem*:

Seek a finite partition $\tilde{\mathcal{T}} \in \text{Part}_N(\Omega)$ *such that for the corresponding measure* G *it holds*

$$\max_{K \in \tilde{\mathcal{T}}}\{G(K)\} = \min_{\mathcal{T} \in \text{Part}_N(\Omega)} \max_{K \in \mathcal{T}}\{G(K)\}. \tag{3.5}$$

Property (3.5) links G-uniform partitions with most of the moving mesh methods of the bibliography where the node positions of the new mesh constitute the minimizer of a proper metric.

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

$$\int_a^{\tilde{x}_i} g(x)\,dx = \frac{i}{N}\int_a^b g(x)\,dx. \tag{3.6}$$

Since we assume that g is strictly positive, (3.6) has as unique solution the set of the unknown nodes $\tilde{x}_i$, $i = 1, 2, \ldots, N-1$. Note that only for simplicity we examine strictly positive estimator functions. If the range of g contains the zero value, then the unknown nodes is not a uniquely defined set. In that case, we can select one of the possible sets by proper treatment of (3.6) (see more about the quantile function in [7], pp. 189–191).

For more than one dimensions, the domain partitioning is a very complex task. In these cases the domain Ω_1 can be approximated by a union of hypertetrahedra of $\mathbb{R}^d$, so the corresponding computational mesh can be described by the topological structure "*simplicial complex*"; a union of chains containing the neighbourhoods of simplices, starting from vertices (0-simplices), edges (1-simplices) and faces (2-simplices), up to facets ($(d-1)$-simplices). Our aim in future work is to construct G-uniform meshes for higher dimensional domains, by sequentially equidistributing proper measures of $\mathbb{R}^d, \mathbb{R}^{d-1}, \ldots, \mathbb{R}^1$ in order to create the corresponding simplicial complex structure with low computational cost.

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

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

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

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

In our experiments we use weights that normalize the terms of the sum, i.e., $a_k = 1/\int_\Omega g_k\,d\mu$, $k = 1, \ldots, M$.

Remark 3 As initially mentioned, 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:

$$G(A) = \int_A g^p\,d\mu. \tag{3.8}$$

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 produces a uniform partition of 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 (3.4), thus the minimum diameter of the partition elements satisfies the relation

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

So choosing p appropriately, the resolution$_{\mathcal{T}}$ of the generated partition T follows the estimator function g while it respects a given upper bound.

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 to zero, this function oscillates 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). In our calculations with double precision on ANSI C, we use the function $(\max\{10^{-20}, x\})^y$ as proper computational version of the power x^y. In this way the result behaves always monotonically with respect to x and produces positive quantities even for the cases $x = 0$ and/or $y = 0$.

3.2 Application to Evolution PDE's

The numerical solution of evolution 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 FE spatial approximations of solution instances, calculated by the evolution steps of the numerical schemes RFE, SRFE, LRFE, DFE. In this case, the redistribution process can be applied at the start of every evolution step in order to produce partitions that improve the resolution of the approximated instance, coming to an end with the reconstruction of the above approximation to the new mesh.

Definition 2 *We shall denote with* $\mathrm{GMesh}(\mathcal{T}, w_{\mathcal{T}})$, *any numerical process which produces a G-uniform partition* $\tilde{\mathcal{T}}$ *for some approximation* $w_{\mathcal{T}}$ *using calculations on a given partition* $\mathcal{T}$. *The reconstruction step shall be denoted by* $\mathrm{Rec}(\mathcal{T}, w_{\mathcal{T}}, \tilde{\mathcal{T}})$ *since it depends on the given data and the new partition.*

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

$$(\tilde{\mathcal{T}}, \tilde{w}_{\tilde{\mathcal{T}}}) = \mathrm{AGMR}(\mathcal{T}, w_{\mathcal{T}}),$$

where

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

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 $\{x_i\}_{i=0}^N$ be the corresponding nodes numbered in an increasing order, i.e., $a = x_0 < x_1 < \cdots < x_{N-1} < x_N = b$. In this work we assume that the estimator is a piecewise constant function defined on the partition $\mathcal{T}$, i.e., $g(x) = g(x_i)$, $x \in [x_i, x_{i+1})$, $i = 0, 1, \ldots, N-1$, so the corresponding distribution $G([a, x)) = \int_a^x g(z)\,dz$ used in (3.6), is piecewise linear. Thus, having the values $G_i = G([a, x_i))$, $i = 0, 1, \ldots, N$, on the node points $\{x_i\}_{i=0}^N$ of the old partition then, (3.6) can explicitly be solved and produce the nodes $\{\tilde{x}_i\}_{i=0}^N$ of the new partition, i.e., $\tilde{x}_0 = a$ and for $\tilde{G}_0 = 0$, $k_0 := 0$, $i = 1, 2, \ldots, N$ they given by:

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

In most of the moving mesh methods in bibliography, the reconstruction on the new mesh, is based on a mass conservative interpolation. In the recent work [31] the L_2 projection was used as a reconstruction function, and in such a way so that the reconstructed approximation preserves in addition the momentum. In our experiments we use as reconstruction function the interpolant operator $\Pi_{S_{r,\tilde{\mathcal{T}}}}(w_{\mathcal{T}})$ of the corresponding FE space $S_{r,\tilde{\mathcal{T}}}$ to the new partition $\tilde{\mathcal{T}}$, thus the reconstruction step here is defined by the equation $\tilde{w}_{\tilde{\mathcal{T}}} = \Pi_{S_{r,\tilde{\mathcal{T}}}}(w_{\mathcal{T}})$. Especially for the case of linear FE spaces, the values $\tilde{w}_{\mathcal{T}}(\tilde{x}_i)$ 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 indices $\{k_i\}_{i=0}^N$ 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 $i = 1, 2, \ldots, N$ they given by:

$$\tilde{w}_{\tilde{\mathcal{T}}}(\tilde{x}_i) = w_{\mathcal{T}}(x_{k_i}) + \frac{\tilde{x}_i - x_{k_i}}{x_{k_i+1} - x_{k_i}} (w_{\mathcal{T}}(x_{k_i+1}) - w_{\mathcal{T}}(x_{k_i})). \tag{3.12}$$

It comes out that the AGMR is an algorithm of linear complexity, since in relations (3.11), (3.12) the indices k_i can be easily obtained.

3.2.1 Estimator Functions for Conservation Laws Problems

Two estimator functions for the AGMR 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 HSCL problems comes from the a posteriori error analysis presented in [1, 12]. 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. Moreover, as it is shown in [1, 12] the estimated error increases significantly over the shock areas, so dispersive terms which are dominant on these areas will be vanished 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:

$$g(x) = \int_{\mathcal{D}} \mathrm{Var}(u_{\mathcal{T}}(s))\,ds, \tag{3.13}$$

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

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

- *The curvature estimator* A useful family of estimator functions is based on geometrical characteristics of the solution. Since in HSCL problems the most interesting areas are over discontinuities (either of the solution or of its derivatives), one possible choice as estimator function is an appropriate approximation of the *curvature* of the already computed solution. The *curvature* of some smooth function v, is the Euclidean norm of the vector curvature of the corresponding graph $\{(x, v(x)) : x \in \text{Domain}(v)\}$, and thus it is defined by

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

At discrete level, given the graph points $\{(x_i, u_{\mathcal{T}}(x_i))\}_{i=0}^{N}$ of the numerical solution, our choice for the corresponding estimator to the *curvature*, is the piecewise constant function defined from the values g_i; the inverse radius of the occluding circle on the plane points $A_j = (x_j, u_{\mathcal{T}}(x_j))$, $j = i-1, i, i+1$, that is:

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

where $\|\cdot\|$ denotes the Euclidean norm. An elementary calculation shows that

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

which, for smooth function $u_{\mathcal{T}}$, is an approximation of the curvature$(u_{\mathcal{T}}(x_i))$.

3.2.2 Graphical Representation of the AGMR Process and the Effect of the Parameter p

The GMesh step of the AGMR process can be represented graphically like in Fig. 1 (left). Starting with some initial data (solid line) we calculate the estimator function $g(x)$

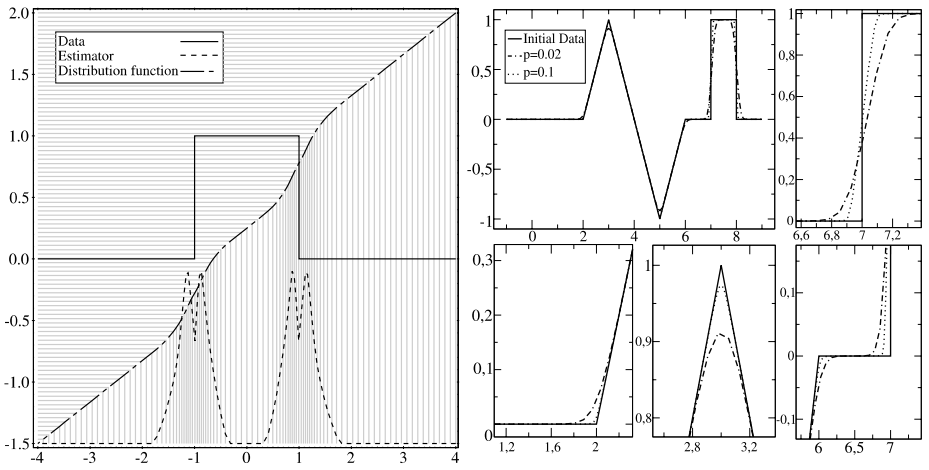

Fig. 1 Graphical representation of the AGMR procedure (*left*) and the effect of the parameter p on the reconstructed function to the new mesh (*right*)

(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 Fig. 1 (right) we see the effect of the parameter p on the AGMR 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 above, since it may lead to a mesh with poor resolution over the smooth areas. For the cases "a posteriori", "*curvature*" we experimentally know that p varies successfully in the interval $[0.005, 0.05]$ (when the AGMR process cooperates with finite element methods).

3.2.3 Finite Element Schemes on Adaptive G-uniform Meshes

Let *Solver* be one of the studied schemes RFE, SRFE, LRFE, DFE. We shall examine the pair which contains a partition and a corresponding approximation on it, during the evolution steps of the Solver. In the case where the Solver applies always on uniform spatial partition $\mathcal{T}_0$, the evolution step for that pair can be represented by the equation:

$$(\mathcal{T}_{n+1}, u^{n+1}_{\mathcal{T}_{n+1}}) = (\mathcal{T}_0, \text{Solver}(\mathcal{T}_n, u^n_{\mathcal{T}_n})), \quad n = 0, 1, \ldots,$$

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

$$\begin{aligned}
&(\mathcal{T}_{n+1}, \tilde{u}_{\mathcal{T}_{n+1}}) = \text{AGMR}(\mathcal{T}_n, u^n_{\mathcal{T}_n})\\
&(\mathcal{T}_{n+1}, u^{n+1}_{\mathcal{T}_{n+1}}) = (\mathcal{T}_{n+1}, \text{Solver}(\mathcal{T}_{n+1}, \tilde{u}_{\mathcal{T}_{n+1}})), \quad n = 0, 1, \ldots.
\end{aligned}$$

We present the AGMR version of the SRFE schemes while the other schemes are defined in similar way. Let $\mathcal{T}_0$ be a uniform partition of Ω with $\text{card}(\mathcal{T}_0) = N$, $S_{r,\mathcal{T}_0}$ be the corresponding FE space and $u^0_{\mathcal{T}_0} = \Pi_{S^M_{r,\mathcal{T}_0}}(u_0)$. Assuming that at t^n, the pair $(\mathcal{T}_n, u^n_{\mathcal{T}_n})$ has been computed, then the pair $(\mathcal{T}_{n+1}, u^{n+1}_{\mathcal{T}_{n+1}})$ at the moment t^{n+1} of the *Switched Relaxation approximation* on *G-uniform mesh* is defined by:

$$\begin{aligned}
&(\mathcal{T}_{n+1}, \tilde{u}_{\mathcal{T}_{n+1}}) = \text{AGMR}(\mathcal{T}_n, u^n_{\mathcal{T}_n}) \quad \textit{and for every } \phi \textit{ in } S^M_{r,\mathcal{T}_{n+1}}\\
&(u^{n+1}_{\mathcal{T}_{n+1}}, \phi) = (\tilde{u}_{\mathcal{T}_{n+1}}, \phi) + \kappa \sum_{i=1}^{q} b_i (-\partial_x v^{n,i}_{\mathcal{T}_{n+1}}, \phi)
\end{aligned} \tag{3.17}$$

with intermediate stages $(u^{n,i}_{\mathcal{T}_{n+1}}, v^{n,i}_{\mathcal{T}_{n+1}})$, $i = 1, \ldots, q$, *given by the following coupled systems of* q*-equations*:

$$\begin{aligned}
(u^{n,i}_{\mathcal{T}_{n+1}}, \phi) &= (\tilde{u}_{\mathcal{T}_{n+1}}, \phi) + \kappa \sum_{j=1}^{i-1} a_{ij} (-\partial_x v^{n,j}_{\mathcal{T}_{n+1}}, \phi),\\
(v^{n,i}_{\mathcal{T}_{n+1}}, \phi) &= (\Pi_{S^M_{r,\mathcal{T}_{n+1}}}(F(\tilde{u}_{\mathcal{T}_{n+1}})), \phi)\\
&\quad + \kappa \sum_{j=1}^{i} \tilde{a}_{ij} \left\{ (-C \cdot \partial_x u^{n,j}_{\mathcal{T}_{n+1}}, \phi) - \frac{1}{\varepsilon} (v^{n,j}_{\mathcal{T}_{n+1}} - F(u^{n,j}_{\mathcal{T}_{n+1}}), \phi) \right\}.
\end{aligned} \tag{3.18}$$

4 Numerical Experiments

In this section we present the experiments we have performed with the RFE, SRFE, LRFE and DFE schemes. In the first problem we study the behavior of the AGMR procedure using the DFE scheme on mesh of 101 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.

The results from the RFE scheme were always coincided with those from the SRFE scheme, for every tested value of parameter ε in the range $[10^{-5}, 10^{-2}]$. Therefore instead of presenting results from this scheme, we give some comparison tables with the minimum number of uniform time steps, needed by the schemes, on uniform spatial mesh, for approximating the solution at a given time end. From these tables one can observe that:

- *The minimum number of uniform time steps needed by some scheme in order to produce, on uniform spatial mesh, the approximation of the solution at a given time end, in the cases of SRFE, LRFE and DFE schemes depends only on the CFL condition while in the case of the RFE scheme it depends also on the relaxation parameter ε.*

For the other schemes we present the results in groups of figures for comparison reasons. Each group contains two separate figures presenting the results on uniform and on adaptive G-uniform partitions. The first group shows the approximations from the SRFE scheme for middle and small value of ε and also the approximation from the LRFE scheme. The next group contains results from the LRFE and the DFE schemes. The last group contains only one figure presenting the results from the DFE scheme on either partition cases, in comparison with some reference solution. From these figures one can observe that:

- *There exist proper ranges for the parameters ε, p for which the SRFE schemes on G-uniform partitions, produce approximations of the entropic solution almost free of dispersive oscillations.*
- *In either uniform or G-uniform partitions, as ε tends to zero the approximation from the SRFE scheme becomes similar to the LRFE's scheme approximation.*
- *The LRFE and DFE approximations even though can be different on uniform partitions, are always similar on adaptive G-uniform partitions (see Burgers example).*
- *It seems that the DFE scheme on adaptive G-uniform partitions can produce approximations of the entropic solution free of dispersive oscillations.*

4.1 The Stationary Equation

In the first problem we study the diffusive behavior of the AGMR process. We consider the stationary equation $\partial_t u = 0$ with initial data $u(x,0) = u_0(x)$, over some spatial interval. This problem can be thought as a conservation law problem with flux $F(u) = 0$, so in this case all the adaptive schemes reduce to the AGMR substep. Therefore, since the solution is $u(x,t) = u_0(x)$, $\forall t \geq 0$, we expect that the sequence $\{(\mathcal{T}_n, u^n_{\mathcal{T}_n})\}_{n\in\mathbb{N}}$, defined by:

$$\begin{aligned} &\mathcal{T}_0 \text{ is uniform partition of } \Omega,\ u^0_{\mathcal{T}_0} = \Pi_{S^M_{r,\mathcal{T}_0}}(u_0), \\ &(\mathcal{T}_{n+1}, u^{n+1}_{\mathcal{T}_{n+1}}) = \mathrm{AGMR}(\mathcal{T}_n, u^n_{\mathcal{T}_n}), \quad n = 0, 1, \ldots, \end{aligned} \tag{4.1}$$

should be convergent to some limit pair $(\mathcal{T}_\star, u^\star_{\mathcal{T}_\star})$, with the term $u^\star_{\mathcal{T}_\star}$ being an approximation of the initial data $u^0_{\mathcal{T}_0}$. Numerical evidences show that indeed the resulting pair $(\mathcal{T}_n, u^n_{\mathcal{T}_n})$

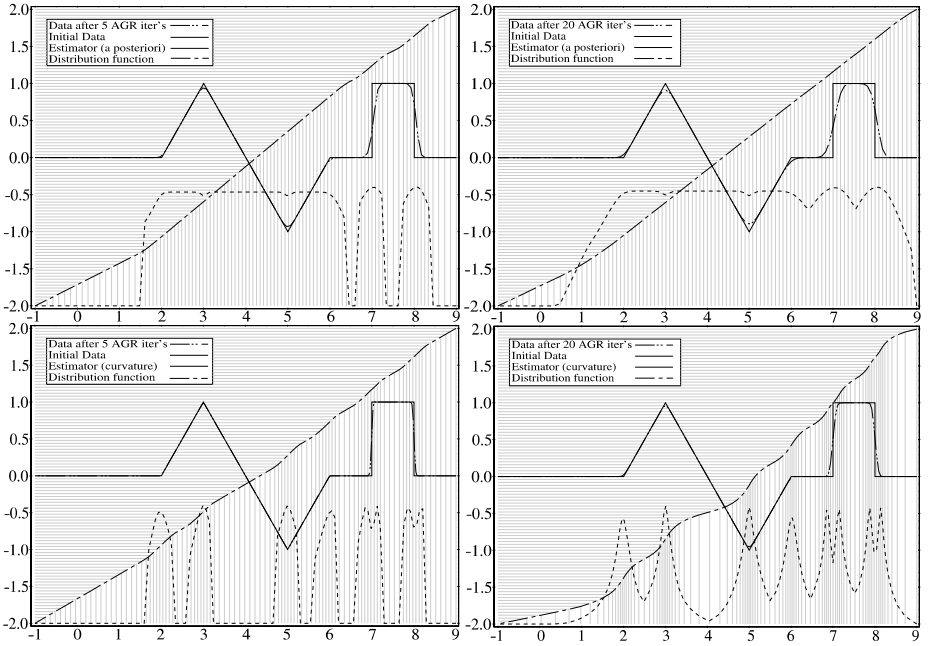

Fig. 2 Iteratively application of the AGMR process on discontinuous data ($N = 101$ nodes, $p = 0.03$), after 5 iterations (*first column*) and after 20 iterations (*second column*). The results in the *first row* were given by using the a-posteriori estimator function while in the *second row* by using the curvature estimator function

after a number of iterations remains steady suggesting that the sequence (4.1) attains its limit. This number depends on the cardinality N of the partition, the complexity of the data structure as well as on the estimator function.

In Fig. 2 we present examples for partitions of 100 elements where the resulting solution $(\mathcal{T}_n, u^n_{\mathcal{T}_n})$ remains steady after about 15 iterations. Observe that the final reconstructed data are indeed diffusive approximation of the initial data $u^0_{\mathcal{T}_0}$. Therefore we must conclude that the AGMR process admits diffusion in the solution. From these figures one can also observe, that the approximation on G-uniform mesh related to the variance is more diffusive than the one on G-uniform mesh related to curvature.

4.2 A Burgers Equation

The next problem we consider, is a Burgers equation with Riemann initial data:

$$\begin{cases} u_0 = \mathbb{X}_{[0,5)} - \mathbb{X}_{[-5,0)\cup[5,6]}, \\ \partial_t u + \partial_x(\frac{u^2}{2}) = 0, \end{cases} \qquad x \in [-5, 6], t \in [0, 2]. \tag{4.2}$$

The exact solution of this problem contains a centered rarefaction wave emanating from the point $x = 0$ and a steady shock at $x = 5$.

The following table contains the minimum number of uniform time steps needed by the RFE, SRFE, LRFE and DFE schemes, in order to produce on uniform spatial mesh the approximation of the solution at the time $t = 2.0$. Observe that for $\varepsilon \leq 10^{-5}$ the RFE scheme needs an unacceptable number of uniform time steps.

Scheme	ε	C	Time steps
RFE	$\varepsilon = 10^{-4}$	10	3800
RFE	$\varepsilon = 10^{-5}$	10	>10000
SRFE	$\varepsilon = 10^{-4}$	10	200
SRFE	$\varepsilon = 10^{-6}$	10	200
LRFE	–	–	200
DFE	–	–	200

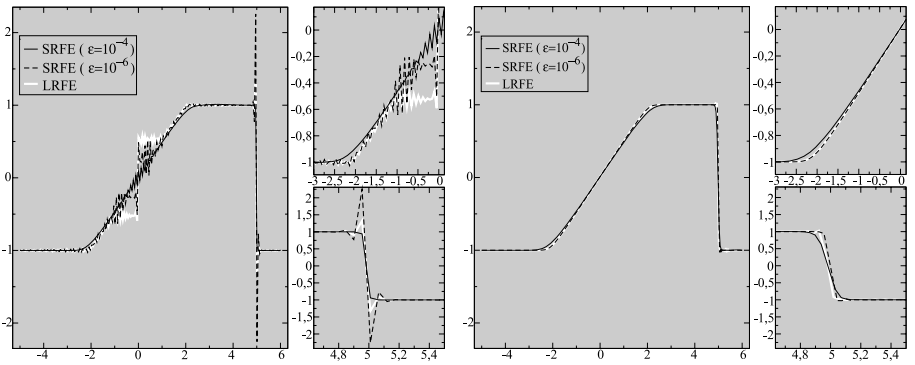

Fig. 3 Burgers problem ($t = 2.0$), SRFE (while ε drops down) and LRFE schemes, on uniform mesh (*left*) and on G-uniform mesh for $p = 0.035$ (*right*)

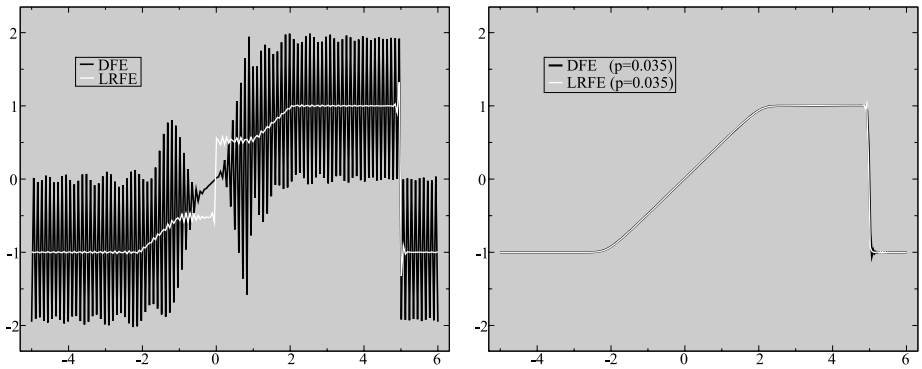

Fig. 4 Burgers problem ($t = 2.0$), LRFE and DFE scheme on uniform mesh (*left*) and on G-uniform mesh for $p = 0.035$ (*right*)

In Figs. 3, 4, 5 we present results for the moment $t = 2.0$. The exact solution was calculated using the corresponding characteristic lines and is given by $u(x, 2) = -\mathbb{X}_{[-5,-2)} + 0.5x\mathbb{X}_{[-2,2)} + \mathbb{X}_{[2,5)} - \mathbb{X}_{[5,6]}$. Observe from Fig. 4 that in this problem the LRFE and DFE approximations are different on uniform partitions while they coincide on adaptive G-uniform partitions. Note that the finite differences scheme of Roe as well as the high resolution TVD schemes of MUSCL type, need a proper modification (entropy fix) in order to overcome an

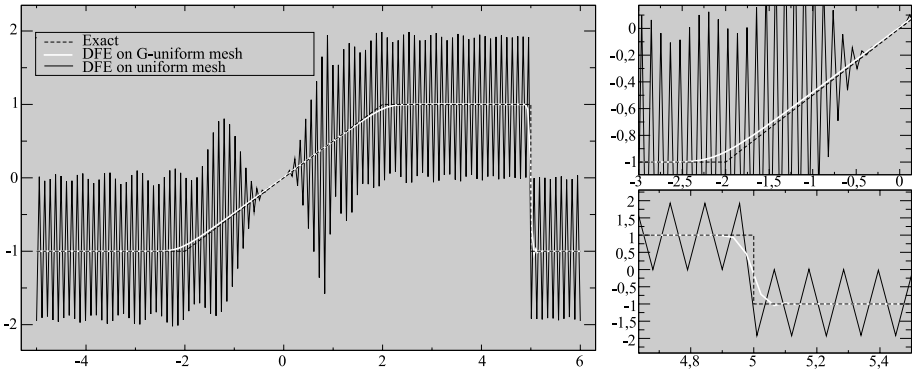

Fig. 5 Burgers problem ($t = 2.0$), DFE scheme on uniform mesh and on G-uniform mesh for $p = 0.035$ and the exact solution

entropy violation at the point $x = 0$ (see [19]), while, the adaptive finite element schemes produce satisfactory approximations without further modifications (figures).

4.3 A Buckley–Leverett Equation

The next problem we consider, is a Buckley–Leverett equation with Riemann initial data:

$$\begin{cases} u_0 = \mathbb{X}_{[0,0.1)\cup[0.5,1]}, \\ \partial_t u + \partial_x \left(\frac{u^2}{u^2 + 0.5(1-u)^2} \right) = 0, \end{cases} \qquad x \in [0,1],\ t \in [0,0.4], \tag{4.3}$$

and has been chosen for testing the proposed FE schemes on conservation laws with non convex flux. In this problem, two rarefaction waves are evolving emanating from steady points, at $x = 0.1$ and $x = 0.5$, and terminating to moving shocks.

The following table contains the minimum number of uniform time steps, needed by the RFE, SRFE, LRFE and DFE schemes in order to produce on uniform spatial mesh the approximation of the solution at the time $t = 0.4$.

Scheme	ε	C	Time steps
RFE	$\varepsilon = 5 \cdot 10^{-4}$	4	400
RFE	$\varepsilon = 1.25 \cdot 10^{-4}$	6	800
RFE	$\varepsilon = 5 \cdot 10^{-5}$	10	3400
SRFE	$\varepsilon = 5 \cdot 10^{-4}$	10	400
SRFE	$\varepsilon = 5 \cdot 10^{-6}$	10	400
LRFE	–	–	400
DFE	–	–	400

In Figs. 6, 7, 8 we present results for the moment $t = 0.28$, when the left shock almost meets the steady point of the right rarefaction wave. Observe from Figs. 7-right and 8 that the adaptive LRFE, DFE schemes produce entropic approximations but some oscillations of small amplitude are still remaining on the shock areas. The approximation from the adaptive

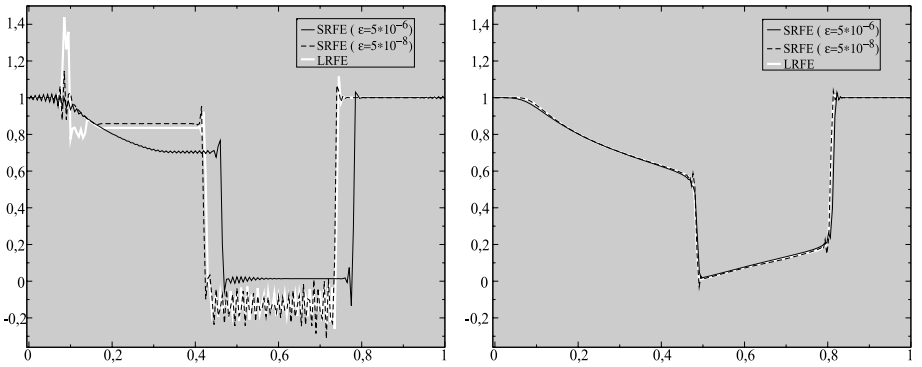

Fig. 6 Buckley Leverett problem ($t = 0.28$), SRFE (while ε drops down) and LRFE schemes, on uniform mesh (*left*) and on G-uniform mesh for $p = 0.035$ (*right*)

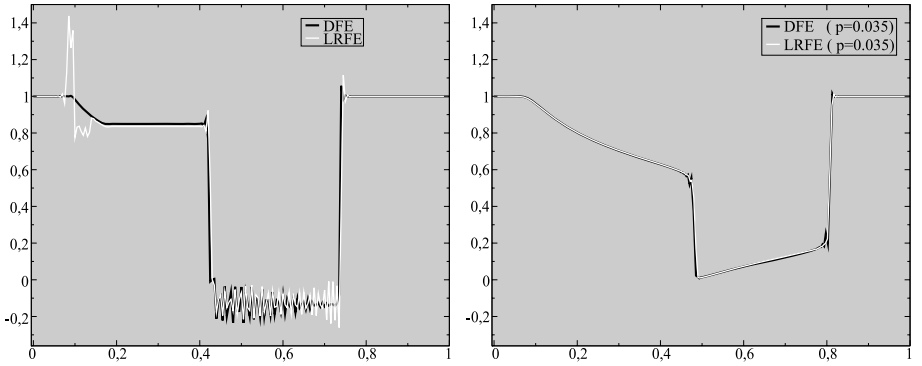

Fig. 7 Buckley Leverett problem ($t = 0.28$), LRFE and DFE scheme on uniform mesh (*left*) and on G-uniform mesh for $p = 0.035$ (*right*)

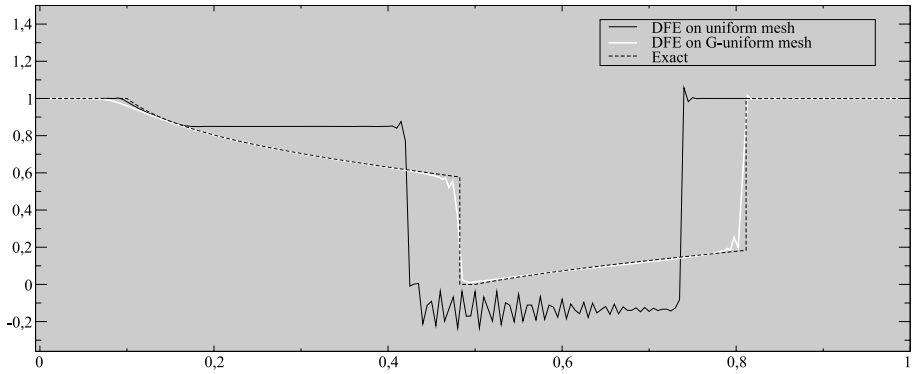

Fig. 8 Buckley Leverett problem ($t = 0.28$), DFE scheme on uniform and on G-uniform mesh for $p = 0.035$ and the exact solution

SRFE 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 the adaptive LRFE scheme. The exact solution was calculated by first solving the non-linear equation of the characteristic lines:

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

in terms of x, using the bisection method 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 [19].

4.4 A Shallow Water System

Finally we consider with the following system of Conservation Law equations, arising in shallow water mechanics:

$$\begin{cases} \begin{pmatrix} u_{0,1} \\ u_{0,2} \end{pmatrix} = \begin{pmatrix} 1 + \mathbb{X}_{[0.3,0.4]} + 0.2\mathbb{X}_{[0.6,0.7]} \\ 0 \end{pmatrix}, \\ \partial_t \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} + \partial_x \begin{pmatrix} u_2 \\ \frac{u_2^2}{u_1} + \frac{u_1^2}{2} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \end{cases} \quad x \in [0, 1],\ t \in [0, 1]. \tag{4.4}$$

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 [19]. As time passes, four shocks and rarefaction waves are formed which are moving to the exterior at opposite directions and approximately at $t = 0.085$, the middle two interacts with each other, yielding this way to a rather complicated structure.

The following table contains the minimum number of uniform time steps, needed by the RFE, SRFE, LRFE and DFE schemes in order to produce on uniform spatial mesh the approximation of the solution at the time $t = 1.0$.

Scheme	ε	C_1, C_2	Time steps
RFE	$\varepsilon = 5 \cdot 10^{-4}$	2, 2	600
RFE	$\varepsilon = 1.25 \cdot 10^{-4}$	2, 2	1600
RFE	$\varepsilon = 5 \cdot 10^{-5}$	2, 2	3800
SRFE	$\varepsilon = 5 \cdot 10^{-4}$	2, 2	400
SRFE	$\varepsilon = 5 \cdot 10^{-5}$	2, 2	400
LRFE	–	–	400
DFE	–	–	400

In Figs. 9–14 we present results for the moment $t = 0.14$. The reference solution was calculated with a TVD finite difference scheme (with the MinMod Limiter) on a uniform spatial mesh of 950 nodes.

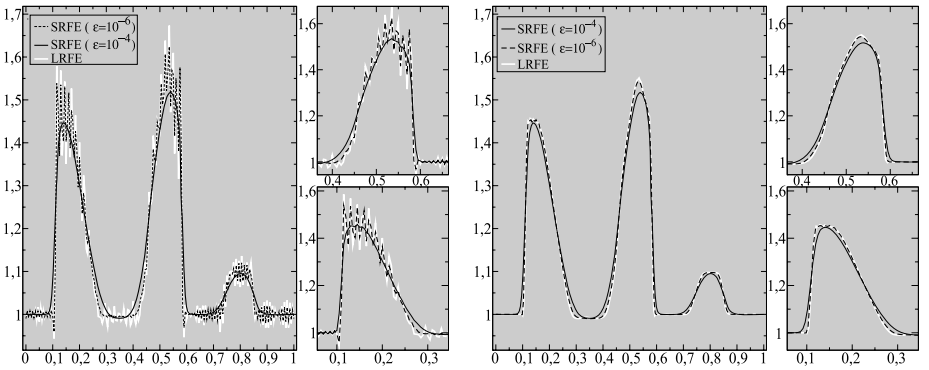

Fig. 9 First component of the Shallow water problem ($t = 0.14$), SRFE (while ε drops down) and LRFE schemes, on uniform mesh (*left*) and on G-uniform mesh for $p = 0.035$ (*right*)

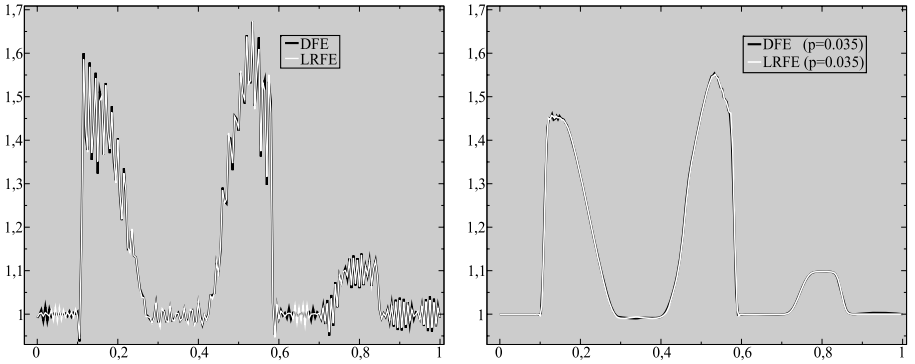

Fig. 10 First component of the Shallow water problem ($t = 0.14$), LRFE and DFE schemes on uniform mesh (*left*) and on G-uniform mesh for $p = 0.035$ (*right*)

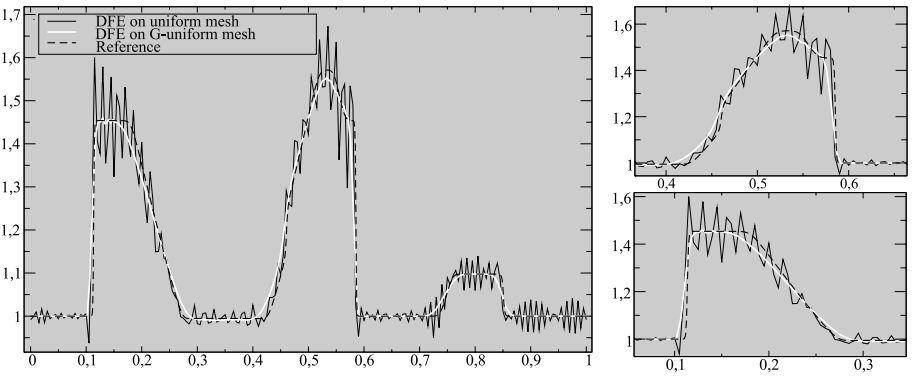

Fig. 11 First component of the Shallow water problem ($t = 0.14$), DFE scheme on uniform and on G-uniform mesh for $p = 0.035$ and the reference solution

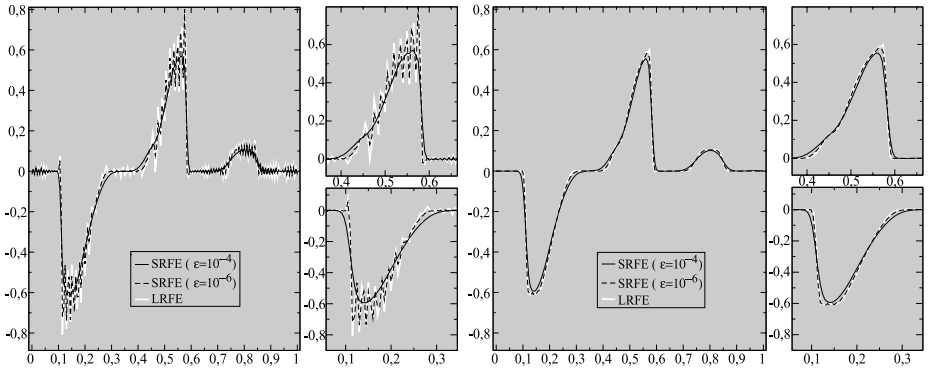

Fig. 12 Second component of the Shallow water problem ($t = 0.14$), SRFE (while ε drops down) and LRFE schemes, on uniform mesh (*left*) and on G-uniform mesh for $p = 0.035$ (*right*)

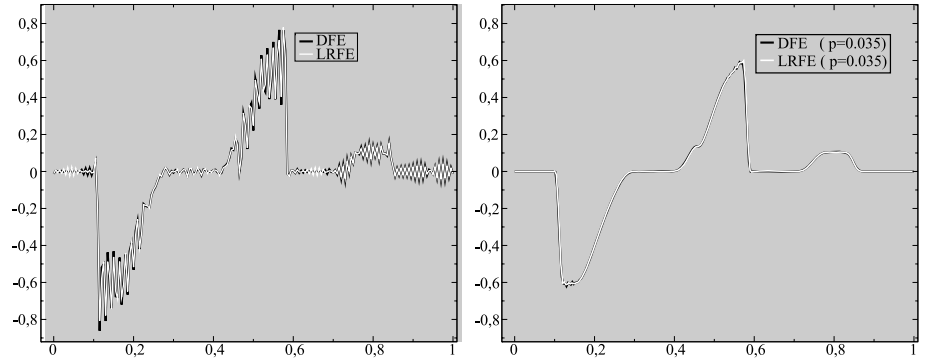

Fig. 13 Second component of the Shallow water problem ($t = 0.14$), LRFE and DFE schemes on uniform mesh (*left*) and on G-uniform mesh for $p = 0.035$ (*right*)

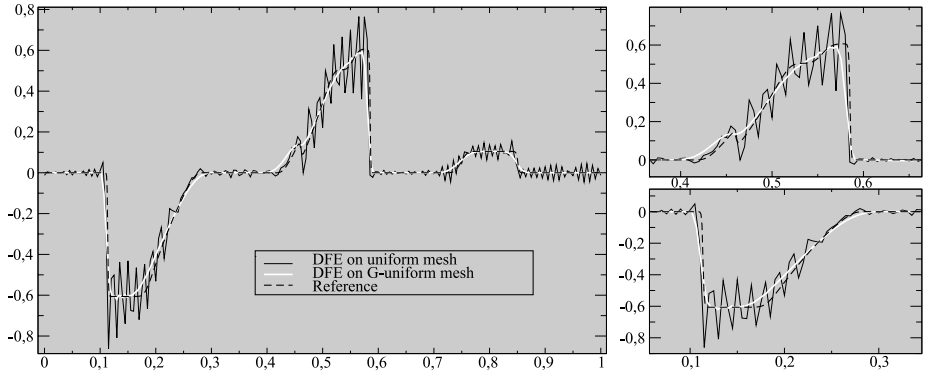

Fig. 14 Second component of the Shallow water problem ($t = 0.14$), DFE scheme on uniform and on G-uniform mesh for $p = 0.035$ and the reference solution

5 Conclusions

In this work we introduced and studied finite element schemes based on relaxation models as well as the *Adaptive G-Mesh Redistribution* process (AGMR).

Among all the presented schemes, probably the most appropriate for shock computations is the Switched Relaxation Finite Element schemes (SRFE); a proper modification of the Relaxation Finite Element schemes (RFE) for approximating the solution of the Relaxation model with the time step being decoupled from the relaxation parameter ε. The RFE schemes were improved by setting, at the start of every evolution step, the relaxation variables to the equilibrium state.

The AGMR process was designed to reconstruct the numerical solution on a new partition of the spatial domain. The generated partition was determined so its resolution must follows a given estimator g, which represents some pre selected characteristics of the numerical solution. From the uniform distribution of the corresponding measure, it was shown in one dimensional domains, how this policy leads to the new partition and for this reason we called it *G-uniform* partition. Choosing g^p as estimator function, the AGMR became flexible for producing a G-uniform mesh with density controllable also from the free parameter p in order to satisfy given bounds.

Then we experimentally tested the adaptive FE 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 performed either on uniform partition or on G-uniform partition related to a functional analogous to the *curvature* of smooth functions. From the corresponding results we made the following conclusions:

- The SRFE schemes on adaptive G-uniform meshes constitute an efficient class for computing solutions for HSCL. 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 area of the relaxation parameters and without further modifications.
- The AGMR process has been proved a good algorithm of low computational cost, for generating partitions leading to high resolution approximations. From the experiments we are convinced that, in addition, this process has strong stabilization properties, i.e., it surpasses the oscillations and produces entropic approximations. It will be very interesting to further develop this algorithm (especially in two and three dimensions) and to provide theoretical support for its behaviour.
- It seems that the pure DFE scheme when applied on adaptive G-uniform meshes becomes a robust non-oscillatory entropic scheme.

The development of the SRFE schemes on adaptive G-uniform meshes is the main goal of this work, but the last two unexpected conclusions might be even more interesting for future research on the field of computational Conservation Laws.

Acknowledgements The author wishes to thank Professors Ch. Makridakis and Th. Katsaounis for 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, C., Katsaounis, T., Makridakis, C.: Adaptive finite element relaxation schemes for hyperbolic conservation laws. Math. Model. Numer. Anal. **35**, 17–33 (2001)

2. Arvanitis, C., Makridakis, C., Tzavaras, A.: Stability and convergence of a class of finite element schemes for hyperbolic systems of conservation laws. SIAM J. Numer. Anal. **42**, 1357–1393 (2004)
3. Azarenok, B.N., Ivanenko, S.A., Tang, T.: Adaptive mesh redistribution method based on Godunov's scheme. Commun. Math. Sci. **1**, 152–179 (2003)
4. Babuška, I.: The adaptive finite element method. TICAM Forum Notes no. 7, University of Texas, Austin (1997)
5. Babuška, I., Gui, W.: Basic principles of feedback and adaptive approaches in the finite element method. Comput. Methods Appl. Mech. Eng. **55**, 27–42 (1986)
6. Beckett, G., Mackenzie, J.A.: Convergence analysis of finite difference approximations on equidistributed grids to a singularly perturbed boundary value problem. Appl. Numer. Math. **35**, 87–109 (2000)
7. Billingsley, P.: Probability and Measure, 2nd edn. Wiley, New York (1992)
8. Brenner, S.C., Scott, L.R.: The Mathematical Theory of Finite Element Methods, 2nd edn. Springer, New York (2002)
9. Cockburn, B., Hou, S., Shu, C.W.: The Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. IV. The multidimensional case. Math. Comput. **54**, 545–581 (1990)
10. Cockburn, B., Johnson, C., Shu, C.W., Tadmor, E.: In: Quarteroni, A. (ed.) Advanced Numerical Approximation of Nonlinear Hyperbolic Equations. Lecture Notes in Mathematics, vol. 1697. Springer, Berlin (1998)
11. Fazio, R., LeVêque, R.J.: Moving-Mesh methods for one-dimensional hyperbolic problems using CLAWPACK. Comput. Math. Appl. **45**, 273–298 (2003)
12. Gosse, L., Makridakis, C.: Two a posteriori error estimates for one dimensional scalar conservation laws. SIAM J. Numer. Anal. **38**, 964–988 (2000)
13. Harten, S., Hyman, J.M.: Self adjusting grid methods for one-dimensional hyperbolic conservation laws. J. Comput. Phys. **50**, 17–33 (1983)
14. Hou, T.Y., Lax, P.D.: Dispersive approximations in fluid dynamics. Commun. Pure Appl. Math. **44**, 1–40 (1991)
15. Hyman, J.M., Li, S., Petzold, L.R.: An adaptive moving mesh method with static rezoning for partial differential equations. Comput. Math. Appl. **46**, 1511–1524 (2003)
16. Jaffré, J., Johnson, C., Szepessy, A.: Convergence of the discontinuous Galerkin finite element method for hyperbolic conservation laws. Math. Models Methods Appl. Sci. **5**, 367–386 (1995)
17. Jin, S., Xin, Z.: The relaxing schemes for systems of conservation laws in arbitrary space dimensions. Commun. Pure Appl. Math. **48**, 235–277 (1995)
18. Johnson, C., Szepessy, A.: On the convergence of a finite element method for a nonlinear hyperbolic conservation law. Math. Comput. **49**, 427–444 (1987)
19. LeVêque, R.J.: Finite Volume Methods for Hyperbolic Problems. Cambridge University Press, Cambridge (2002)
20. Li, R., Tang, T., Zhang, P.: Moving Mesh methods in multiple dimensions based on harmonic maps. J. Comput. Phys. **170**, 562–588 (2001)
21. Li, S., Petzold, L.: Moving Mesh methods with upwind schemes for time dependent PDEs. J. Comput. Phys. **131**, 368–377 (1997)
22. Li, S., Petzold, L., Ren, Y.: Stability of moving Mesh systems of partial differential equations. SIAM J. Sci. Comput. **20**, 719–738 (1998)
23. Lipnikov, K., Shashkov, M.: The error-minimization-based strategy for moving Mesh methods. Commun. Comput. Phys. **1**, 53–81 (2006)
24. Shu, C.W.: Total-variation-diminishing time discretizations. SIAM J. Sci. Comput. **9**, 1073–1084 (1988)
25. Shu, C.W., Osher, S.: Efficient implementation of essentially nonoscillatory shock-capturing schemes. J. Comput. Phys. **77**, 439–471 (1988)
26. Stockie, J.M., Mackenzie, J.A., Russell, R.D.: A Moving mesh method for one-dimensional hyperbolic conservation laws. SIAM J. Sci. Comput. **22**, 1791–1813 (2001)
27. Tan, Z., Zhang, Z., Huang, Y., Tang, T.: Moving mesh methods with locally varying time steps. J. Comput. Phys. **35**, 17–33 (2004)
28. Tang, H.: Solution of the shallow-water equations using an adaptive moving mesh method. Int. J. Numer. Methods Fluids **44**, 789–810 (2004)
29. Tang, H., Tang, T.: Adaptive mesh methods for one- and two-dimensional hyperbolic conservation laws. SIAM J. Numer. Anal. **41**, 487–515 (2003)
30. Tzavaras, A.: Viscosity and relaxation approximation for hyperbolic systems of conservation laws. In: Lecture notes in Computational Science and Engineering, vol. 5, pp. 73–122. Springer, New York (1998)
31. Zhang, Z.: Moving mesh methods for convection-dominated equations and nonlinear conservation problems. Ph.D. Thesis, Hong Kong Baptist University (2003)
32. Zhang, Z.: Moving mesh method with conservative interpolation based on L^2-projection. Commun. Comput. Phys. **1**, 930–944 (2006)