HAL
open science

Numerical simulation of conservation laws with moving grid nodes: Application to tsunami wave modelling

Gayaz Khakimzyanov, Denys Dutykh, Dimitrios Mitsotakis, Nina Shokina

To cite this version:

Gayaz Khakimzyanov, Denys Dutykh, Dimitrios Mitsotakis, Nina Shokina. Numerical simulation of conservation laws with moving grid nodes: Application to tsunami wave modelling. Geosciences, 2019, 9 (5), pp.197. 10.3390/geosciences9050197 . hal-01223510v6

HAL Id: hal-01223510

https://hal.science/hal-01223510v6

Submitted on 25 Apr 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution - NonCommercial - NoDerivatives 4.0 International License

Gayaz Khakimzyanov
Institute of Computational Technologies, Novosibirsk, Russia

Denys Dutykh
CNRS, Université Savoie Mont Blanc, France

Dimitrios Mitsotakis
Victoria University of Wellington, New Zealand

Nina Shokina
Albert-Ludwigs-Universität Freiburg, Freiburg im Breisgau, Germany

Numerical simulation of conservation laws with moving grid nodes: Application to tsunami wave modelling

arXiv.org / HAL

Numerical simulation of conservation laws with moving grid nodes: Application to tsunami wave modelling

Gayaz Khakimzyanov, Denys Dutykh*, Dimitrios Mitsotakis, and Nina Yu. Shokina

Abstract. In the present article we describe a few simple and efficient finite volume type schemes on moving grids in one spatial dimension combined with appropriate predictor–corrector method to achieve higher resolution. The underlying finite volume scheme is conservative and it is accurate up to the second order in space. The main novelty consists in the motion of the grid. This new dynamic aspect can be used to resolve better the areas with large solution gradients or any other special features. No interpolation procedure is employed, thus unnecessary solution smearing is avoided, and therefore, our method enjoys excellent conservation properties. The resulting grid is completely redistributed according the choice of the so-called *monitor function*. Several more or less universal choices of the monitor function are provided. Finally, the performance of the proposed algorithm is illustrated on several examples stemming from the simple linear advection to the simulation of complex shallow water waves. The exact well-balanced property is proven. We believe that the techniques described in our paper can be beneficially used to model tsunami wave propagation and run-up.

Key words and phrases: Conservation laws; finite volumes; conservative finite differences; moving grids; adaptivity; advection; shallow water equations; wave run-up

MSC: [2010]74S10 (primary), 74J15, 74J30 (secondary)

* Corresponding author.

Contents

1 Introduction **4**

2 Linear scalar equation **7**

2.1 A predictor-corrector scheme on a fixed uniform mesh 7

2.2 Predictor-corrector schemes on a moving grid 9

2.3 Construction and motion of the grid 11

Initial grid generation 12
Unsteady computations 13
A lyrical digression 13
Smoothing step 14
On the choice of the monitor function 14

2.4 Numerical results 15

3 Nonlinear scalar equation **17**

3.1 Numerical results 19

4 Nonlinear system of equations **20**

4.1 Finite volume discretization 23

Well-balanced property 25
Conservation property on moving grids 26

4.2 Numerical results 28

Exact solution derivation 28
Validation 29
Solitary wave run-up 32

5 Some open problems **33**

6 Discussion **35**

6.1 Conclusions 35

6.2 Perspectives 36

Acknowledgments 36

References **37**

A Illustrative examples **42**

A.1 Example 1 42

A.2 Example 2 43

B Abbreviations 44

1. Introduction

The main goal of the present manuscript consists in introducing to tsunami modelling community some state-of-the-art numerical methods for solving nonlinear hyperbolic equations, which allow to use the available degrees of freedom in the (nearly-) optimal way. Until the last 15 years, tsunami wave modelling has been done in the framework of Nonlinear Shallow Water Equations (NSWE). The numerical simulations have been performed by practitioners essentially with several well-established community codes. To give a few examples, let us mention the code TUNAMI [50] based on a conservative finite difference leap-frog scheme on staggered grids. This scheme approximates NSWE on real bathymetries. Another widely used code is MOST, which is based on the dimensional splitting method* [86, 87]. We would like to mention also the software COMCOT, which is based on a modified leap-frog scheme discretizing NSWE in Cartesian and spherical coordinates [91]. Finally, in the Institute of Computational Technologies (SB RAS) an in-house software MGC based on the finite difference MacCormack scheme in spherical coordinates was developed as well [76].

A big boost in tsunami research happened after Tsunami Boxing day [83] due to the availability of real world data. After this Sumatra 2004 event, several studies questioned the importance of dispersive effects for trans-oceanic tsunami propagation [22, 43, 69]. Their conclusions were supported by comparisons of numerical results with field and DART buoy data. We can only mention that the importance of dispersive effects has been already highlighted in 1982 by Mirchina and Pelinovsky [68]:

> *[. . .] the considerations and estimates for actual tsunamis indicate that nonlinearity and dispersion can appreciable affect the tsunami wave propagation at large distances. [. . .]*

One of the first tsunami numerical models including spherical effects *and* linear dispersive terms was TUNAMI-N2. A sensitivity study was performed in [23] on the example of Sumatra 2004 event. The reported results were in favor of including dispersive effects. Another Weakly Nonlinear Weakly Dispersive (WNWD) model was presented in [64, 65]. It was basically a spherical counterpart of the classical Peregrine system [70]. Let us mention also the study by Glimsdal *et al.* (2013) [38] where the importance of dispersion has been investigated numerically and the use of Fully Nonlinear Weakly Dispersive (FNWD) model was recommended. Such FNWD spherical equations (with the horizontal velocity defined on a certain surface inside the fluid bulk) have been derived in [57], however, only WNWD numerical results were presented. In a recent work [53] a systematic derivation of spherical rotating FNWD (and WNWD) models have been performed and the so-called *base model* was derived, which incorporates many other models as particular cases. The reduction of spherical FNWD to WNWD models was illustrated and several known models were recovered in this way. Finally, the fully nonlinear numerical results have been presented in [54]. In particular, the importance of sphericity, Coriolis and dispersion effects was thoroughly

*This technique is also known as the alternating directions.

discussed. The numerical algorithm employed in [54] is based on the splitting approach: an elliptic equation is solved to determine the non-hydrostatic pressure component and an evolution system of hyperbolic equations with source terms is solved to advance in time the velocity and water height variables. The algorithm was implemented as an explicit two-step predictor–corrector scheme. The volume of publication [54] did not allow the authors to describe all details of the employed numerical method. Henceforth, if we adopt dispersive models for tsunami simulations*, there is a clear need in robust numerical methods to solve hyperbolic balance equations. The present manuscript should be considered as a tutorial-style article to introduce the moving grid algorithms to the developers of future generation tsunami propagation codes.

The numerical simulation of conservation laws is one of the most dynamic and central parts of modern numerical analysis. The systems of conservation laws appear in many fields ranging from traffic modeling [67] to shallow water modelling [32] and compressible fluid mechanics [92]. The finite volume method was proposed in a pioneering work of S. Godunov (1959) [39, 40] and developed later by Ph. Roe [73] and many other researchers (see [35] for the current state of the art).

Nowadays the complexity of the problems which arise in practice is such that we have to think about the optimal usage of available computational resources. In this way the community came with the idea of developing numerical methods on adaptive grids in order to have higher resolution only where it is needed [6]. Historically, *hp*-adaptive methods were first developed for the Finite Element-type (FEM) discretizations [7] (including the newest discontinuous Galerkin discretizations as well [46]). The main advantage of FEM is that this method is quite flexible in respect of local approximation spaces, thus allowing for relatively easy p-adaptivity. FEM was applied to hyperbolic problems under the guise of relaxation schemes [3]. Later h-adaptive methods have been adapted for finite volume methods as well [11]. However, the widely used approach nowadays consists in performing local grid refinement (or coarsening) with locally nested grids [10]. For instance, it has been successfully applied to tsunami propagation and run-up problems [37] as well as to more complex two-phase incompressible flows [71].

Adaptive mesh refinement techniques applied to the spatial variable after the computation of an approximate solution of the model equations at each time step usually consist of the following steps:

- Generation of a new spatial mesh according to prescribed adaptivity criteria;
- Reconstruction or interpolation of the numerical solution on the new mesh;
- Integration in time of the numerical solution on the new mesh.

The main disadvantages of this methodology is that the reconstruction of the numerical solution on the new mesh increases the complexity of the method and also reduces the accuracy of the method by adding more dissipation or dispersion to the solution. Moreover, the variable number of nodes represents some difficulties while implementing the algorithm

*It seems now that future generation tsunami modelling codes will include at least some dispersive effects.

on a computer. The underlying data structures have to be flexible enough to insert or to suppress some elements.

The approach we propose in this study is different and does not require interpolation or reconstruction of the solution to the new mesh (a conservative interpolation is employed in [84]). Although the proposed method is closer to the redistribution method proposed in [2, 84], it is simpler and more elegant at the level of implementation details. Namely, it is based on the transformation of the model equations to some new equations whose solutions are independent of the new mesh. Namely, the method uses the same number of discretization points of the spatial mesh during the simulation and the adaptivity is achieved by moving the grid nodes to the places indicated by the so-called *monitor function*. The adapted grid is obtained as a solution of an elliptic (or parabolic) problem, which ensures the smoothness of the obtained grid. The main idea behind the equidistribution principle is to distribute the nodes over the computational domain so that the measure of a cell times the value of the monitor function on it be approximatively constant [27, 28]. The heart of the matter in the equidistribution method is the construction of a suitable monitor function. This method was described independently in the seminal paper [78], and later in [47]. Numerous subsequent developments were published in recent references [36, 49, 63]. The review of earlier works on the adaptive grid generation can be found in [41, 62]. The 'Holy Grail' in choosing the monitor function is to achieve ideally the reduction of error in orders of magnitude when the discretization step $\Delta x \to 0$, as illustrated in [51]. The main advantages of the proposed approach include:

- Conservation in space;
- Second order accurate (in smooth regions);
- TVD property for scalar problems;
- Well-balanced character;
- Preservation of the discrete conservation law in cell centers and cell interfaces.

The present manuscript sheds also some light on the hyperobolic part of the splitting approach we used earlier to simulate fully nonlinear and weakly dispersive wave propagation [55].

For simplicity, we focus on one-dimensional cases only, and even in this simple case some open problems are outlined throughout the manuscript. According to our knowledge the pioneers in 1D case were [82] (for nonlinear shallow water equations) and [1] (for gas dynamics). The generation of 2D curvilinear adaptive grids is a relatively classical topic [75]. A rigorous definition of a curvilinear grid was given in [79]. The generalization of the equidistribution method to two spatial dimensions was performed for the first time in [26]. There are a few studies which report recent modern implementations on *moving* redistributed two-dimensional grids (see *e.g.* [5, 9, 17, 84]) with generalized monitor functions formalized in [49]. Some approaches for the inter-comparison of generated grids are described in [72]. We refer also to [14] for a recent review of moving grid techniques. The moving grid methods seem to be unavoidable in the modern numerical analysis. There are some classes of problems, which cannot be solved, in principle, with a fixed grid. To

give an example, we can mention the class of problems involving moving boundaries such as free surface flows [12, 52, 58].

The discretization of the model equations is based on explicit, fully discrete predictor-corrector Finite Volume schemes that are accurate up to the second order in space. Predictor–corrector schemes have been introduced by MacCormack in [66] and can be considered as a natural generalization of splitting schemes in higher spatial dimensions.

We have to mention that non-uniform grids can be used also to preserve some Lie symmetry group of the equation at the discrete level (see [18, Section 3.2.2] for a brief survey and the Burgers–Hopf equation example). Originally this approach was proposed by Dorodnitsyn [30] and developed later in [48]. In our study the mesh motion is directed solely by the equidistribution principle as in [84] without taking into account the symmetry considerations for the moment.

The present article is organized as follows. In Section 2 we discuss in all details the implementation of the algorithm for the linear advection equation. Then, we generalize it to the nonlinear case in Section 3 and systems of conservation laws in Section 4. Some open problems are outlined in Section 5. Finally, the main conclusions and perspectives of our study are given in Section 6.

2. Linear scalar equation

In order to present the moving grid method we shall start with the simplest scalar linear advection equation in this section and, then, we shall increase gradually the complexity by adding first the nonlinearities in the following section and moving to the systems by the end of our manuscript.

We pay special attention to the linear case for the following reasons:

- The main properties of the scheme are the most transparent in the linear case;
- The generalization to the nonlinear and vectorial cases will be easier once the linear case is fully understood. So, it will allow us to go faster in the subsequent sections;
- An exhaustive error analysis is possible in the linear (and presumably only in the linear) case.

In order to construct an efficient finite volume scheme on a moving mesh, we have to choose first a robust and an accurate scheme on a fixed grid, which will be generalized later to incorporate the motion of mesh points. Such a scheme retained for our study is described in the next Section.

2.1. A predictor-corrector scheme on a fixed uniform mesh

Consider the following linear advection equation with a constant propagation velocity

$$u_t + a\, u_x = 0, \qquad a \in \mathbb{R}. \tag{2.1}$$

The subscripts in this study denote the partial derivatives, *i.e.* $u_t = \frac{\partial u}{\partial t}$, $u_x = \frac{\partial u}{\partial x}$, *etc.*

We introduce a uniform discretization of the real space line $\mathbb{R}$ with nodes $x_j = j\,\Delta x$, where $\Delta x > 0$, $j \in \mathbb{Z}$ is the spatial discretization step, and for simplicity we do not pay attention to the boundary conditions here. The interval $\mathbf{C}_j = [x_j, x_{j+1}]$ will be referred as the cell $\mathbf{C}_j$. The time step is denoted by τ and the discrete solution is computed at $t^n = n\tau$, $n \in \mathbb{Z}^+$. Equation (2.1) is discretized in space and time using an explicit fully discrete predictor-corrector scheme

$$\frac{u^*_{j+1/2} - u^n_{j+1/2}}{\tau^*_{j+1/2}} + a\, u^n_{x,j+1/2} = 0, \qquad \frac{u^{n+1}_j - u^n_j}{\tau} + a\,\frac{u^*_{j+1/2} - u^*_{j-1/2}}{\Delta x} = 0, \tag{2.2}$$

where τ is the time step. The intermediate quantities $u^*_{j\pm1/2}$ are evaluated in the middle of cells at $x_{j\pm1/2} = \dfrac{x_j + x_{j\pm1}}{2} = x_j \pm \frac{1}{2}\Delta x$ and at time instances $t = t^n + \tau^*_{j+1/2}$. Moreover, the following notations have been introduced in (2.2):

$$u^n_{j+1/2} = \frac{u^n_{j+1} + u^n_j}{2}, \quad u^n_{x,j+1/2} = \frac{u^n_{j+1} - u^n_j}{\Delta x}, \quad \tau^*_{j+1/2} = \frac{\tau}{2}\left(1 + \theta^n_{j+1/2}\right).$$

We underline the fact that the parameter θ has not been fixed for the moment and it may vary from one cell and one time layer to another one.

The predictor-corrector scheme (2.2) can be recast as a one-step scheme by combining two equations together:

$$\frac{u^{n+1}_j - u^n_j}{\tau} + a\,\frac{u^n_{j+1} - u^n_{j-1}}{2\Delta x} - \frac{\tau a^2}{2\Delta x}\left\{\left((1+\theta)u_x\right)^n_{j+1/2} - \left((1+\theta)u_x\right)^n_{j-1/2}\right\} = 0. \tag{2.3}$$

The last fully discrete scheme is a canonical form of all two steps explicit schemes for equation (2.1), since many well-known schemes can be obtained from (2.3) by choosing carefully the parameter θ. For example,

Lax–Wendroff scheme [60] ($\theta = 0$):

$$\frac{u^{n+1}_j - u^n_j}{\tau} + a\,\frac{u^n_{j+1} - u^n_{j-1}}{2\Delta x} - \frac{\tau a^2}{2\Delta x}\left\{u^n_{x,j+1/2} - u^n_{x,j-1/2}\right\} = 0.$$

First order upwind [20] ($\theta = \frac{1}{C} - 1 > 0$): The number $C \coloneqq |a|\,\dfrac{\tau}{\Delta x} < 1$ is the famous Courant–Friedrichs–Lewy number [19], which controls the stability of the upwind scheme

$$\frac{u^{n+1}_j - u^n_j}{\tau} + \frac{a + |a|}{2}\,u^n_{x,j-1/2} + \frac{a - |a|}{2}\,u^n_{x,j+1/2} = 0.$$

Lax–Friedrichs scheme [13] ($\theta = \frac{1}{C^2} - 1 > 0$):

$$\frac{u^{n+1}_j - \frac{1}{2}\left(u^n_{j+1} + u^n_{j-1}\right)}{\tau} + a\,\frac{u^n_{j+1} - u^n_{j-1}}{2\Delta x} = 0.$$

However, the schemes listed above are not suitable for accurate simulations, since the first order upwind and Lax–Friedrichs schemes are too diffusive in practice and Lax–Wendroff scheme is not monotonicity preserving. A good solution consists in choosing the parameter θ adaptively. In a previous work [77] the following strategy was proposed:

$$\theta^n_{j+1/2} = \begin{cases} 0, & \left|u^n_{x,j+1/2}\right| \leqslant \left|u^n_{x,j+1/2-s}\right| \quad \text{and} \quad u^n_{x,j+1/2}\cdot u^n_{x,j+1/2-s} \geqslant 0, \\ \theta_0(1-\xi^n_{j+1/2}), & \left|u^n_{x,j+1/2}\right| > \left|u^n_{x,j+1/2-s}\right| \quad \text{and} \quad u^n_{x,j+1/2}\cdot u^n_{x,j+1/2-s} \geqslant 0, \\ \theta_0, & u^n_{x,j+1/2}\cdot u^n_{x,j+1/2-s} < 0, \end{cases}$$

with $\theta_0 \coloneqq \dfrac{1}{C}-1>0$, $s \coloneqq \operatorname{sign}(a)$, $\xi^n_{j+1/2} \coloneqq \dfrac{u^n_{x,j+1/2-s}}{u^n_{x,j+1/2}}$. The first case corresponds to the classical Lax–Wendroff scheme, the third case is the classical first order upwind scheme and the second case is detailed in the following Remark 1. However, the proposed combination of schemes produces a robust TVD-type scheme, which ensures a very good trade-off between the solution accuracy and monotonicity properties. The scheme parameter θ plays somehow the rôle of the traditional `minmod` slope limiter in more conventional MUSCL-type finite volume methods [61]. Its performance on fixed and mobile meshes will be illustrated in numerical sections below. The complete justification and validation of this scheme can be found in [77].

Remark 1. *We note that the canonical form (2.3) contains not only three points schemes on a symmetric stencil. For instance, if we assume for simplicity $a>0$, and set*

$$\theta^n_{j+1/2} \coloneqq \left(\frac{1}{C} - 1\right)\left(1 - \frac{u^n_{x,j-1/2}}{u^n_{x,j+1/2}}\right),$$

then we shall obtain the following second order upwind scheme on an asymmetric stencil:

$$\frac{u^{n+1}_j - u^n_j}{\tau} + a\,\frac{3u^n_{x,j-1/2} - u^n_{x,j-3/2}}{2} - \frac{\tau a^2}{2}\,\frac{u^n_j - 2u^n_{j-1} + u^n_{j-2}}{\Delta x^2} = 0.$$

Of course, the numerical properties such as the accuracy, stability and monotonicity depend crucially on the choice of the parameter θ and have to be studied carefully in each particular situation.

2.2. Predictor-corrector schemes on a moving grid

In this section we shall restrict our attention to bounded domains only since almost all numerical simulations in 1D are made on finite intervals. To be more specific we assume that the computational domain is $\mathcal{I} = [0, \ell]$. Consider also the reference domain $Q = [0, 1]$ and a smooth bijective time-dependent mapping $x(\cdot, t): Q \times \mathbb{R}^+ \mapsto \mathcal{I}$, which satisfies the following boundary conditions

$$x(0, t) = 0, \qquad x(1, t) = \ell. \tag{2.4}$$

For our purely numerical purposes we do not even need to have the complete information about this mapping. Consider a uniform mesh of the reference domain $Q_h = \{q_j =$

$j\,\Delta q\}_{j=0}^{N}$, with a constant step $\Delta q \coloneqq \dfrac{1}{N}$. Strictly speaking we need to know only the image of nodes q_j under the map $x(q,t)$, since they constitute the nodes of the moving mesh:

$$x(q_j, t^n) \;=\; x_j^n\,.$$

The linear advection equation (2.1) can be rewritten on domain Q if we introduce the composed function $v(q,t) \;\coloneqq\; (u \circ x)(q,t) \;\equiv\; u\big(x(q,t),t\big)$:

$$v_t \;+\; \frac{\bar a}{J}\,v_q \;=\; 0\,, \tag{2.5}$$

where $\bar a \;\coloneqq\; a - x_t$ is the advection speed relative to the mesh nodes velocity and $J \;\coloneqq\; x_q \;>\; 0$ is the Jacobian of the transformation $x(q,t)$. Equation (2.5) does possess a conservative form as well

$$(Jv)_t \;+\; (\bar a\,v)_q \;=\; 0. \tag{2.6}$$

Now we can write the predictor-corrector scheme for the linear advection equation on a moving mesh:

$$\frac{v^*_{j+1/2} \;-\; v^n_{j+1/2}}{\tau^*_{j+1/2}} \;+\; \left(\frac{\bar a}{J}\,v_q\right)^n_{j+1/2} \;=\; 0, \tag{2.7}$$

$$\frac{(Jv)^{n+1}_j \;-\; (Jv)^n_j}{\tau} \;+\; \frac{\left(\bar a^n\,v^*\right)_{j+1/2} \;-\; \left(\bar a^n\,v^*\right)_{j-1/2}}{\Delta q} \;=\; 0. \tag{2.8}$$

Note that the first step is based on equation (2.5) and the second one on the conservative form (2.6). Here we introduced some new notations:

$$x^n_{t,j+1/2} \;=\; \frac{x^n_{t,j} \;+\; x^n_{t,j+1}}{2}, \qquad x^n_{t,j} \;=\; \frac{x^{n+1}_j \;-\; x^n_j}{\tau},$$

$$J^n_{j+1/2} \;\equiv\; x^n_{q,j+1/2} \;=\; \frac{x^n_{j+1} \;-\; x^n_j}{\Delta q}, \qquad J^n_j \;\equiv\; x^n_{q,j} \;=\; \frac{J^n_{j+1/2} \;+\; J^n_{j-1/2}}{2} \;=\; \frac{x^n_{j+1} \;-\; x^n_{j-1}}{2\Delta q}.$$

As before, one can recast the scheme (2.7), (2.8) in an equivalent one-stage form:

$$\frac{(Jv)^{n+1}_j - (Jv)^n_j}{\tau} + \frac{(\bar a v)^n_{j+1/2} - (\bar a v)^n_{j-1/2}}{\Delta q} - \frac{\tau}{2\Delta q}\left\{\left((1+\theta)\frac{\bar a^2}{J}v_q\right)^n_{j+1/2} - \left((1+\theta)\frac{\bar a^2}{J}v_q\right)^n_{j-1/2}\right\} \;=\; 0.$$

A stability study using the method of frozen coefficients [74] leads to the following restrictions on the scheme parameters

$$\theta^n_{j+1/2} \geqslant 0, \qquad \max_{j\in\{0,\dots,N-1\}}\left(\sqrt{1+\theta}\,C\right)^n_{j+1/2} \leqslant 1, \tag{2.9}$$

where $C^n_{j+1/2}$ is the local CFL number [19] defined as

$$C^n_{j+1/2} \;=\; \frac{\tau}{\Delta q}\left(\frac{|\,\bar a\,|}{J}\right)^n_{j+1/2}.$$

In order to ensure the monotonicity of the numerical solution, it was shown in [77] that it is sufficient to use the following choice of the scheme parameter θ:

$$\theta^n_{j+1/2} = \begin{cases} 0, & \left|\tilde g^n_{j+1/2}\right| \leqslant \left|\tilde g^n_{j+1/2-s}\right| \quad \text{and} \quad \tilde g^n_{j+1/2}\cdot \tilde g^n_{j+1/2-s} \geqslant 0, \\ \left(\theta_0(1-\xi)\right)^n_{j+1/2}, & \left|\tilde g^n_{j+1/2}\right| > \left|\tilde g^n_{j+1/2-s}\right| \quad \text{and} \quad \tilde g^n_{j+1/2}\cdot \tilde g^n_{j+1/2-s} \geqslant 0, \\ \theta^n_{0,j+1/2}, & \tilde g^n_{j+1/2}\cdot \tilde g^n_{j+1/2-s} < 0, \end{cases} \tag{2.10}$$

where $s = \operatorname{sign}(\bar a^n_{j+1/2})$ and

$$\theta^n_{0,j+1/2} = \frac{1}{C^n_{j+1/2}} - 1, \qquad \tilde g^n_{j+1/2} = \left(|\bar a|\,(1-C)\,v_q\right)^n_{j+1/2}, \qquad \xi^n_{j+1/2} = \frac{\tilde g^n_{j+1/2-s}}{\tilde g^n_{j+1/2}}.$$

Using (2.10), it can be shown that stability requirements (2.9) are fulfilled under the following single restriction on local CFL numbers:

$$C^n_{j+1/2} \leqslant 1.$$

Finally, the global CFL number C is defined as

$$C \coloneqq \max_j \{C^n_{j+1/2}\}.$$

It is this value, which will be reported in Tables below.

There is an additional relation, which is called the *geometric conservation law*. In 1D case it takes a very simple form

$$J_t - (x_t)_q = 0,$$

since it translates a trivial fact that $x_{qt} \equiv x_{tq}$. However, it was demonstrated in [85] that it is absolutely crucial to respect the geometric conservation law at the discrete level as well in order to have a consistent fully discrete scheme which preserves all constant solutions exactly. It can be shown that the scheme proposed above satisfies the discrete counterpart of the geometric conservation law in integer and half-integer nodes:

$$\frac{J^{n+1}_j - J^n_j}{\tau} - \frac{x^n_{t,j+1/2} - x^n_{t,j-1/2}}{\Delta q} = 0,$$
$$\frac{J^{n+1}_{j+1/2} - J^n_{j+1/2}}{\tau} - \frac{x^n_{t,j+1} - x^n_{t,j}}{\Delta q} = 0.$$

2.3. Construction and motion of the grid

In the previous sections we explained in some detail the predictor-corrector schemes on fixed and moving meshes. We saw that the mesh motion is parametrized by a bijective mapping $x(q,t) : Q \times \mathbb{R}^+ \mapsto \mathcal{I}$. Now we explain how this map is effectively constructed, since so far it was quite general. Moreover, the predictor-corrector schemes presented above are valid for other constructions of the mesh motion.

The first step consists in choosing a positive valued function $w(x,t) : \mathcal{I} \times \mathbb{R}^+ \mapsto \mathbb{R}^+$, the so-called *monitor function*, which controls the distribution of the nodes. Without

entering into the details for the moment, we suggest as one choice for a monitor function the following expression:

$$w(x,t) = 1 + \alpha \left| u_x(x,t) \right|, \tag{2.11}$$

where $\alpha > 0$ is a real parameter. The monitor function (2.11) is a particular case of the class of monitor functions considered by Beckett, Mackenzie *et al.* [9]. Other choices of the monitor function are discussed in [15, 17, 80]. For instance, to give an example, another popular choice of the monitor function is

$$w(x,t) = \sqrt{1 + \alpha \left| u_x(x,t) \right|^2}.$$

In general, the choice of the monitor function and its parameters might be crucial for the accuracy of the scheme. The adaptive grid construction for simple linear singularly perturbed Sturm–Liouville equation is deeply analysed in [51]. This question seems to be rather open for unsteady nonlinear problems. The current state-of-the-art in this field is described in [49].

It is noted that the monitor function varies in both space and time and takes large (positive) values in areas where the solution has important gradients. The monitor function $w(x,t)$ specified in (2.11) can be readily discretized

$$w^n_{j+1/2} = 1 + \alpha \frac{\left| u^n_{j+1} - u^n_j \right|}{x^n_{j+1} - x^n_j}.$$

The proposed algorithm is slightly different on the very first step when the initial grid is generated. It is important to obtain a high-quality initial mesh, since the numerical error committed in the beginning cannot be corrected afterwards. So, the first step is explained in detail below and the algorithm for unsteady computations is presented in the following section.

2.3.1 Initial grid generation

It is of capital importance to produce a grid adapted to the initial condition as well, since no error committed initially can be corrected during dynamical simulations. At $t = 0$ we compute the monitor function $w(x,0)$ of the initial condition $u(x,0)$ and the mapping $x(q,0)$ is determined as the solution to the following *nonlinear* elliptic problem, which degenerates to a simple second order ordinary differential equation

$$\left(w(x,0)\, x_q \right)_q = 0, \tag{2.12}$$

supplemented with Dirichlet boundary conditions (2.4). This elliptic problem can be readily discretized to produce the following finite difference approximation:

$$\frac{1}{\Delta q} \left\{ w^0_{j+1/2} \frac{x^0_{j+1} - x^0_j}{\Delta q} - w^0_{j-1/2} \frac{x^0_j - x^0_{j-1}}{\Delta q} \right\} = 0, \quad j = 1, \ldots, N-1,$$

with boundary conditions $x^0_0 = 0$, $x^0_N = \ell$. The last nonlinear system of equations is solved iteratively (usually with some fixed point-type iterations). The iterations are initialized

with a uniform grid as the first guess. One can notice also that its solution satisfies the equidistribution principle:

$$w^0_{j+1/2}\,\frac{x^0_{j+1}\;-\;x^0_j}{\Delta q}\;=\;C_{\Delta q},\qquad j\;=\;0,\ldots,N-1,\tag{2.13}$$

where the constant C_h can be computed as

$$C_{\Delta q}\;=\;\sum_{j=0}^{N-1}(\Delta x\cdot w)^0_{j+1/2},\qquad \Delta x^0_{j+1/2}\;\coloneqq\;x^0_{j+1}\;-\;x^0_j.$$

Therefore, $C_{\Delta q}$ is the discrete version of the monitor function integral. The equidistribution principle gives meaning to the monitor function: in the areas where $w^0_{j+1/2}$ takes large values, the spacing between two neighboring nodes x^0_{j+1} and x^0_j has to be inversely proportionally small.

2.3.2 Unsteady computations

Assume that the grid $\{x^n_j\}_{j=0}^N$ at $t\;=\;t^n$ is known and let us evolve it to the next time layer t^{n+1}. It is done by solving the following *linear* fully discrete problem:

$$\frac{1}{\Delta q}\left\{w^n_{j+1/2}\,\frac{x^{n+1}_{j+1}\;-\;x^{n+1}_j}{\Delta q}\;-\;w^n_{j-1/2}\,\frac{x^{n+1}_j\;-\;x^{n+1}_{j-1}}{\Delta q}\right\}\;=\;\beta\,\frac{x^{n+1}_j-x^n_j}{\tau},\quad j\;=\;1,\ldots,N-1,\tag{2.14}$$

with the same boundary conditions $x^{n+1}_0\;=\;0$, $x^{n+1}_N\;=\;\ell$ as above. Here the real positive parameter $\beta>0$ plays the rôle of the diffusion coefficient and it controls the smoothness of nodes trajectories. The *optimal* choice of the parameter β is currently an open problem and it has to be done empirically in case by case basis. It can be easily seen that the scheme (2.14) is an implicit discretization of this nonlinear parabolic equation:

$$\big(\,w\,(x,\,t)\,x_{\,q}\,\big)_{\,q}\;=\;\beta\,x_{\,t}\,.\tag{2.15}$$

However, it has to be stressed out that the fully discrete problem is linear since the monitor function $w^n_{j\pm1/2}$ is taken on the previous time layer. Thus, the overall complexity of the grid motion algorithm at every time step is equal to that of the tridiagonal matrix algorithm [34].

2.3.3 A lyrical digression

It is probably not completely clear to the reader why one has to use a different procedure initially, especially since the solution at every time step might be considered as initial condition to the next time level. This question would be completely legitimate.

The moving grid technique based on the solution of a (nonlinear) parabolic Equation (2.15) was specifically designed for the solution of unsteady problems. It was proposed for the first time in [78] to simulate gas dynamics problems and it allows to ensure the required *smoothness* of grid points trajectories (see numerical examples below). The position of grid points on two consecutive time steps are related by a finite-difference analogue

of Equation (2.15). This modification of the equidistribution principle allows to avoid any rapid changes in the location of grid nodes. Basically, it was understood in the original work [78] and confirmed later in subsequent developments in late (19)80's and early (19)90's [24, 28, 29, 93, 94]. Subsequent developments have been done in [47–49]. Some recommendations against using the steady Equation (2.12) to determine the position of grid nodes on each time step were formulated in [21, 25, 94]. Some examples of grid nodes oscillations and other instabilities were shown there as well. In Appendix A we provide a few *analytical* examples to illustrate eventual problems which may arise if one uses the equidistribution principle (2.12) on every time step. Indeed, we demonstrate in Appendix A that even infinitesimal perturbations of the monitor function may yield finite perturbations in the solution $x(q, t)$.

2.3.4 Smoothing step

In some problems where the solution contains a lot of oscillations (*i.e.* many local extrema), the monitor function $w(x, t)$ has to be filtered out to ensure the smoothness of the mesh motion. Moreover, a smoothing step allows to enlarge the set of acceptable values of the parameter α in the definition (2.11) of the monitor function. Our numerical experiments show that the following implicit procedure (inspired by implicit discretizations of the heat equation) produces very robust results:

$$\bar{w}_{j+1/2} = w_{j+1/2} - \sigma\bar{w}_{j+1/2} + \frac{\sigma}{2}\left(\bar{w}_{j-1/2} + \bar{w}_{j+3/2}\right), \qquad j = 1, \ldots, N-2, \tag{2.16}$$

where $\sigma > 0$ is an ad-hoc smoothing parameter to be set empirically. In our study we do smoothing similarly to Asselin's filter proposed earlier for time integration problems [4]. System (2.16) is completed with boundary conditions

$$\bar{w}_{1/2} = w_{1/2}, \qquad \bar{w}_{N-1/2} = w_{N-1/2}.$$

The smoothed discrete monitor function $\{\bar{w}_j\}_{j=0}^{N}$ is obtained by solving the linear system (2.16) with any favorite method. In the present study we used the simplest tridiagonal matrix algorithm [34]. In contrast, the authors of [84] used an explicit smoothing method. We stress out that one should apply a smoothing operator to the monitor function and not to the numerical solution.

2.3.5 On the choice of the monitor function

In general, when one has a problem (*i.e.* a system of PDEs) to solve, there is a question of the best possible monitor function choice for the grid adaptation. One has to choose also the pertinent solution component as an input for the monitor function. It is extremely difficult to give some general recommendations which would work in all cases*. However, in choosing the monitor function one has to take into account the following aspects:

*It is probably as difficult as the general theory of PDEs.

Physical considerations: any a priori knowledge on the solution can be used to underline important physical aspects (*e.g.* the presence of shock waves or the presence of multiple wave crests, *etc.*);

Geometrical considerations: computational domain geometry (*e.g.* the presence of angles or fractal shorelines in wave run-up problems) and topology (*i.e.* the presence of islands or other obstacles);

Mathematical considerations: nature of equations (differential, partial differential, integro-differential, singularly perturbed, *etc.*), but also one has to pick up the most pertinent variable to compute the monitor function on it (the free surface elevation or the vorticity field in Navier–Stokes simulations);

Numerical considerations: particularities of the employed discretization scheme may also hint the choice of the monitor function.

Once the monitor function is chosen, one has to determine also (nearly) optimal values of parameters which dependent not only on the problem, but also on the solution. Finally, the most important guideline nowadays is the *experience* in solving practical problems using moving grid techniques. It cannot be replaced by any guidelines. We refer also to the review [14].

2.4. Numerical results

In order to illustrate the performance of this algorithm we choose a notorious test case, which consists in simulating the propagation of a discontinuous profile:

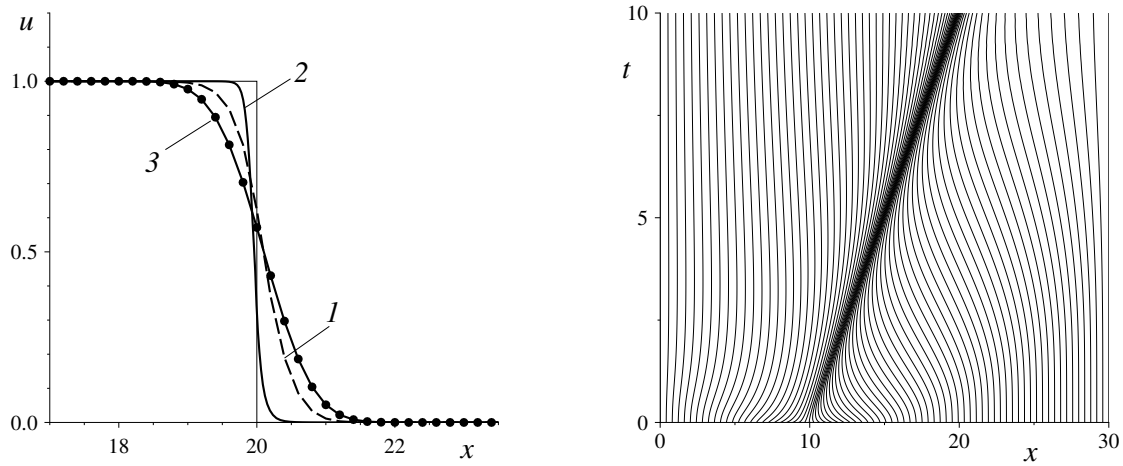
$$u(x,0) = \begin{cases} 1, & x \leqslant x_*, \\ 0, & x > x_*, \end{cases} \qquad x_* \in]\,0, \ell\,[\,.$$

The monitor function employed in this computation is defined in (2.11). The numerical parameters used in this simulations are given in Table 1. The simulation results at time $t = T_f$ is shown in Figure 1. From the comparison of profiles obtained on fixed and moving meshes, one can see that the moving mesh reduces significantly the "*width*" of the numerical shock wave profile.

The second test case is a smooth bell-shaped function advected by the dynamics of (2.1). This test is used to measure the ability of the scheme to preserve the extrema (in other words we assess the strength of the numerical diffusion on a given mesh). The exact solution is given by

$$u(x,t) = \mathrm{e}^{-25(x - x_0 - at)^2}.$$

Thus, the initial condition is simply given by $u(x,0) = \mathrm{e}^{-25(x-x_0)^2}$. Another particularity of the present test case is that we use the monitor function specifically designed with emphasis



(a) (b)

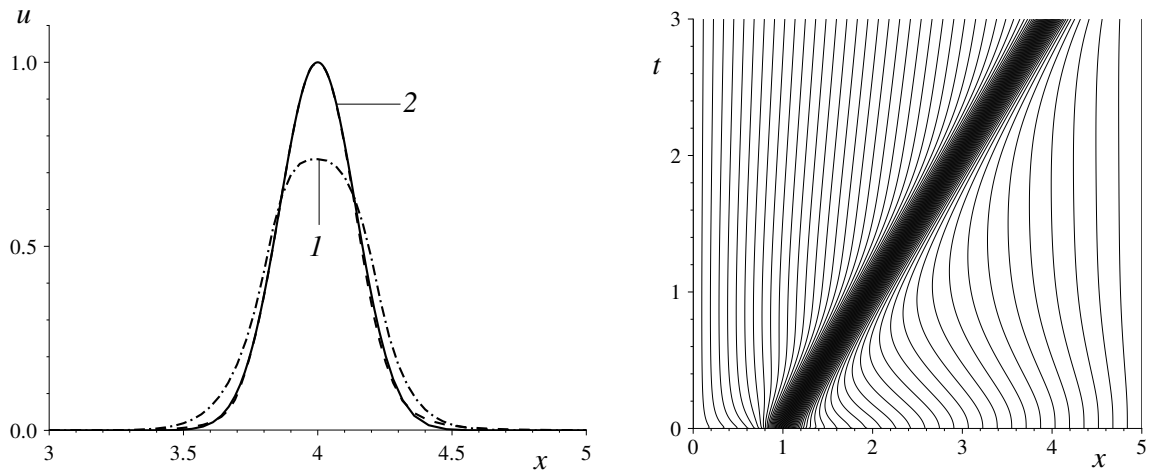
Figure 1. *Discontinuous profile propagation under the linear advection equation (2.1): (a) comparison of different profiles (thin solid line is the exact solution, (1) is the numerical solution on the uniform mesh, (2) is the solution on the moving mesh and (3) is the first order upwind scheme on the uniform mesh); (b): trajectories of grid nodes in space-time. Note that the panel (a) shows only a zoom of the computational domain.*

Parameter	*Value*
Computational domain length, ℓ	30.0
Number of nodes, N	150
CFL number, C	0.8
Final simulation time, T_f	10.0
Propagation speed, a	1.0
Initial position of the discontinuity, x_*	10.0
Monitor function parameter, α	10.0
Grid diffusion parameter, β	150.0
Smoothing parameter, σ	100.0

Table 1. *Numerical parameters used to simulate the discontinuous profile propagation under the linear advection equation (2.1).*

on the fine resolution of the local extrema (those with high positive or high negative values):

$$w(x,t) \;=\; 1 \;+\; \alpha\,|\,u(x,t)\,|\,. \tag{2.17}$$



(a) (b)

Figure 2. *Smooth profile propagation under the linear advection equation (2.1): (a) comparison of different profiles (thin solid line is the exact solution, (1) is the numerical solution on the uniform mesh, (2) is the solution on the moving mesh); (b): trajectories of grid nodes in space-time. Note that the panel (a) shows only a zoom of the computational domain.*

It is noted that there is no differentiation operator in the definition of the last monitor function compared to the one defined in (2.11).

The parameters used in the numerical simulation are provided in Table 2. The numerical results at $t = T_f$ are shown in Figure 2. For the sake of comparison we show also the numerical solution obtained with the same scheme, but on a uniform mesh. The use of a moving grid improves dramatically the preservation of the local maximum. One cannot even distinguish the exact solution from the numerical one up to the graphic resolution.

3. Nonlinear scalar equation

In this section we consider a general hyperbolic scalar nonlinear conservation law of the form:

$$u_t + [f(u)]_x = 0.$$

As a chrestomathic example of such equations we can mention the celebrated Burgers–Hopf equation [16]. As above, we make a change of variables to the fixed reference domain Q:

$$v_t + \frac{1}{J}\left[f_q - x_t v_q\right] = 0,$$

Parameter	*Value*
Computational domain length, ℓ	5.0
Number of nodes, N	150
CFL number, C	0.8
Final simulation time, T_f	3.0
Initial position of the bell, x_0	1.0
Propagation speed, a	1.0
Monitor function parameter, α	20.0
Grid diffusion parameter, β	20.0
Smoothing parameter, σ	10.0

Table 2. *Numerical parameters used to simulate the bell-shaped profile propagation under the linear advection equation* (2.1).

or in the conservative form:

$$(Jv)_t + \left[f - x_t v\right]_q = 0,$$

where $v \coloneqq u \circ x$ and $J \coloneqq x_q$ as before.

For nonlinear equations we need an extra transport equation for the flux function:

$$f_t + a(u)\, f_x = 0,$$

where $a(u) = f_u(u)$. The last equation transforms to the following non-conservative form on the fixed domain Q:

$$f_t + \frac{a(v)}{J}\left[f_q - x_t v_q\right] = 0.$$

Now we can write down the predictor-corrector scheme directly on a moving grid (notice that in the nonlinear case three stages are required):

$$\begin{aligned}
\frac{v^*_{j+1/2} - v^n_{j+1/2}}{\tau^*_{j+1/2}} + \left(\frac{f_q - x_t v_q}{J}\right)^n_{j+1/2} &= 0,\\
\frac{f^*_{j+1/2} - f^n_{j+1/2}}{\tau^*_{j+1/2}} + \left(\frac{a(f_q - x_t v_q)}{J}\right)^n_{j+1/2} &= 0,\\
\frac{(Jv)^{n+1}_j - (Jv)^n_j}{\tau} + \frac{(f^* - x^n_t v^*)_{j+1/2} - (f^* - x^n_t v^*)_{j-1/2}}{\Delta q} &= 0,
\end{aligned}$$

where $v^n_{j+1/2}$, $f^n_{j+1/2}$ were defined above. The last scheme has the second order approximation in space $\mathcal{O}(\Delta q^2)$ for smooth solutions provided that $\theta = \mathcal{O}(\Delta q)$. The only point which remains to be specified is the computation of the numerical Jacobian $a^n_{j+1/2}$. In order to ensure good properties of the scheme, one has to ensure the compatibility condition

$f_q = a(v)\, v_q$ at the discrete level $f^n_{q,j+1/2} = (a\, v_q)^n_{j+1/2}$ for finite differences analogue of the derivatives:

$$f^n_{q,j+1/2} = \frac{f^n_{j+1} - f^n_j}{\Delta q}, \qquad v^n_{q,j+1/2} = \frac{v^n_{j+1} - v^n_j}{\Delta q}.$$

This requirement can be satisfied with the following choice [45]:

$$a^n_{j+1/2} = \begin{cases} \frac{f^n_{j+1} - f^n_j}{v^n_{j+1} - v^n_j}, & v^n_{j+1} \neq v^n_j, \\ a(v^n_j), & v^n_{j+1} = v^n_j. \end{cases}$$

By noticing that the corrector step contains only the combination $f^* - x^n_t\, u^*$, we can write a compact two-stage form of the predictor-corrector scheme:

$$\left(\frac{\hat{f} - f^n}{\tau^*}\right)_{j+1/2} + \left(\frac{\bar{a}^2}{J}\, v_q\right)^n_{j+1/2} = 0,$$

$$\frac{(Jv)^{n+1}_j - (Jv)^n_j}{\tau} + \frac{\left(\hat{f} - (x_t\, v)^n\right)_{j+1/2} - \left(\hat{f} - (x_t\, v)^n\right)_{j-1/2}}{\Delta q} = 0,$$

where $\bar{a}(v, q, t) = a(v) - x_t$. The parameter $\theta^n_{j+1/2}$ is computed according to (2.10) which ensures the TVD property of the scheme. As a result we obtain a generalization of the Harten scheme [45] to the case of moving meshes.

3.1. Numerical results

In order to illustrate the performance of the generalized Harten scheme on moving meshes we take the classical example of the inviscid Burgers–Hopf equation ($f(u) = \frac{u^2}{2}$). As the first illustration let us take the following initial condition on the computational domain $x \in [0, \ell]$:

$$u_0(x) = \begin{cases} u_l, & x \leqslant x_l, \\ \dfrac{x - x_r}{x_l - x_r}\, u_l + \dfrac{x - x_l}{x_r - x_l}\, u_r, & x_l < x < x_r, \\ u_r & x \geqslant x_r. \end{cases}$$

All numerical parameters are given in Table 3. The exact *weak* solution to this problem is given by the following formula:

$$u(x,t) = \begin{cases} u_l, & x \leqslant x_l + u_l\, t, \\ \dfrac{x - x_*}{t - t_*}, & x_l + u_l\, t < x < x_r + u_r\, t, \\ u_r, & x \geqslant x_r + u_r\, t, \end{cases} \qquad t < t_*.$$

The gradient catastrophe takes place at $t = t^*$ and $x = x_*$, which can be computed exactly as well

$$t^* = -\frac{x_r - x_l}{u_r - u_l}, \qquad x_* = x_l + u_l\, t^* \equiv x_r + u_r\, t^*.$$

Parameter	*Value*
Computational domain length, ℓ	30.0
Number of nodes, N	60
Final simulation time, T_f	10.0
CFL number, C	0.2
Left transition coordinate, x_l	10.0
Right transition coordinate, x_r	20.0
Left constant state, u_l	1.0
Right constant state, u_r	−1.0
Monitor function parameter, α	15.0
Grid diffusion parameter, β	80.0
Smoothing parameter, σ	60.0

Table 3. *Numerical parameters used to simulate a stationary shock-wave formation under the* Burgers–Hopf *equation dynamics.*

After $t = t^*$ we obtain a travelling shock-wave profile:

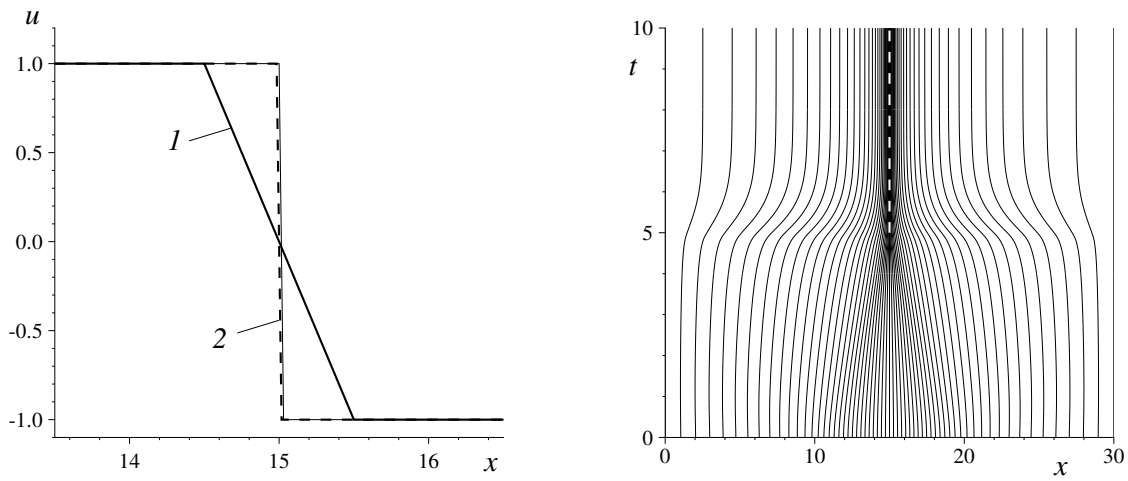
$$u(x,t) = \begin{cases} u_l, & x \leqslant x_* + \{u\}(t - t^*), \\ u_r, & x > x_* + \{u\}(t - t^*), \end{cases} \qquad \{u\} := \frac{u_l + u_r}{2}.$$

For the parameters chosen in our numerical simulation (see Table 3) the shock wave remains stationary at $t = t^* = 5$ and $x = x_* = 15$ since $u_l = -u_r$. The numerical solution and the grid motion are shown in Figure 3. It can be seen that the solution monotonicity is preserved on both fixed and moving meshes. However, on the moving grid we obtain a solution which cannot be distinguished from the exact one up to the graphical resolution. On the fixed grid the shock wave has a finite width equal to two control volumes.

As another numerical illustration, we consider the same problem, but we change the value of $u_r := 0$ instead of -1. In this case the gradient catastrophe takes place at $t = t^* = 10$ and $x = x_* = 20$. The shock wave profile is not steady anymore since $\{u\} = \dfrac{u_l + u_r}{2} = \dfrac{1}{2} \neq 0$. So, it will move in the rightward direction with a constant speed equal to $\dfrac{1}{2}$. The results of numerical simulations are shown on Figure 4. It can be seen again that the adaptive solution is excellent, while the fixed grid produces a shock wave smeared out on four cells only.

4. Nonlinear system of equations

The predictor-corrector scheme for nonlinear systems on moving meshes will be presented on a practically important case of the Nonlinear Shallow Water Equations (NSWE) [81].



(a) (b)

Figure 3. *Stationary shock wave formation in the* Burgers–Hopf *equation: (a) comparison of different profiles (thin solid line is the exact solution, (1) is the numerical solution on the uniform mesh, (2) is the solution on the moving mesh); (b): trajectories of grid nodes in space-time. Note that the panel (a) shows only a zoom of the computational domain.*

This choice reflects also the scientific interests of the authors of the present article. Moving grid techniques have been applied to other systems of conservation laws such as the Euler equations in [80], pressureless gas dynamics [59] and magnetohydrodynamics [89].

So, consider one-dimensional nonlinear shallow water equations written in a vectorial form:

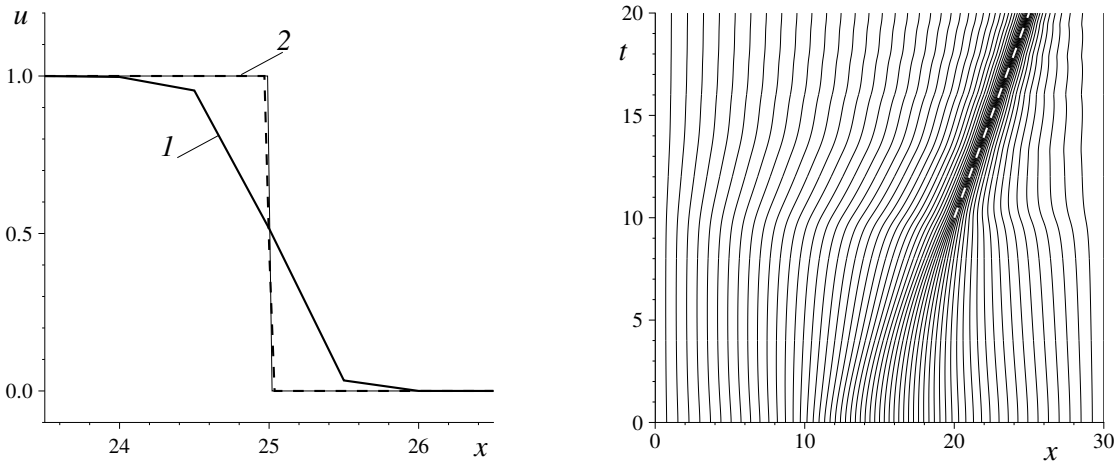
$$\mathbf{u}_t + \left[\, \boldsymbol{f}(\mathbf{u}) \,\right]_x = \mathbf{G}(\mathbf{u}, x)\,, \tag{4.1}$$

where $\mathbf{u}$ is the vector of conservative variables, $\boldsymbol{f}(\mathbf{u})$ is the flux function and $\mathbf{G}(\mathbf{u}, x)$ contains the source terms:

$$\mathbf{u} = \begin{pmatrix} H \\ Hu \end{pmatrix}, \qquad \boldsymbol{f}(\mathbf{u}) = \begin{pmatrix} Hu \\ Hu^2 + \frac{g}{2}H^2 \end{pmatrix}, \qquad \mathbf{G}(\mathbf{u}, x) = \begin{pmatrix} 0 \\ gHh_x \end{pmatrix}.$$

Here $u(x,t)$ is the depth-averaged velocity and $H(x,t) = \eta(x,t) + h(x)$ is the total water depth defined as the sum of the bathymetry function $h(x)$ and the free surface elevation $\eta(x,t)$ over the mean water level. The sketch of the physical domain with all these definitions is shown in Figure 5. The system (4.1) is considered on a finite segment $\mathcal{I} = [0, \ell]$ with appropriate boundary conditions. As before, we consider a smooth bijective time-dependent transformation $x = x(q,t)$ from the reference domain $Q = [0,1]$ into $\mathcal{I}$. The NSWE on the reference domain are written in the form [8]:

$$\boldsymbol{v}_t + \frac{1}{J}\left[\boldsymbol{f}_q - x_t\, \boldsymbol{v}_q\right] = \frac{1}{J}\, \mathbf{G}(\boldsymbol{v}, q),$$



(a) (b)

Figure 4. *Stationary shock wave formation in the* Burgers–Hopf *equation: (a) comparison of different profiles (thin solid line is the exact solution, (1) is the numerical solution on the uniform mesh, (2) is the solution on the moving mesh); (b): trajectories of grid nodes in space-time. Note that the panel (a) shows only a zoom of the computational domain.*

where $\boldsymbol{v}(q,t) = \mathbf{u} \circ x = \mathbf{u}\big(x(q,t),t\big)$ and the source term contains a derivative with respect to q:

$$\mathbf{G}(\boldsymbol{v},q) \;=\; \begin{pmatrix} 0 \\ gHh_q \end{pmatrix}, \qquad h(q,t) \;=\; h\big(x(q,t)\big).$$

It is noted that the bathymetry becomes time-dependent under this change of variables, even if initially the bottom was fixed. There exists also a conservative form NSWE on the reference domain:

$$(J\boldsymbol{v})_t \;+\; [\boldsymbol{f} \;-\; x_t\,\boldsymbol{v}]_q \;=\; \mathbf{G}(\boldsymbol{v},q), \qquad J \;\coloneqq\; x_q \;>\; 0.$$

Finally, the flux vector satisfies the following equation:

$$\boldsymbol{f}_t \;+\; \frac{1}{J}\,\mathscr{A}[\boldsymbol{f}_q \;-\; x_t\boldsymbol{v}_q] \;=\; \frac{1}{J}\,\mathscr{A}\mathbf{G}(\boldsymbol{v},q),$$

where $\mathscr{A}$ is the advective flux's Jacobian matrix, which can be easily computed:

$$\mathscr{A}(\boldsymbol{v}) \;=\; \frac{\mathrm{D}\boldsymbol{f}(\boldsymbol{v})}{\mathrm{D}\boldsymbol{v}} \;=\; \begin{pmatrix} 0 & 1 \\ -v^2 + gH & 2v \end{pmatrix}.$$

Since the system of NSWE (4.1) is hyperbolic, the Jacobian $\mathscr{A}$ can be decomposed into the product of left $\mathscr{L}$ and right $\mathscr{R}$ eigenvectors:

$$\mathscr{A} \;=\; \mathscr{R}\cdot\mathscr{D}\cdot\mathscr{L},$$

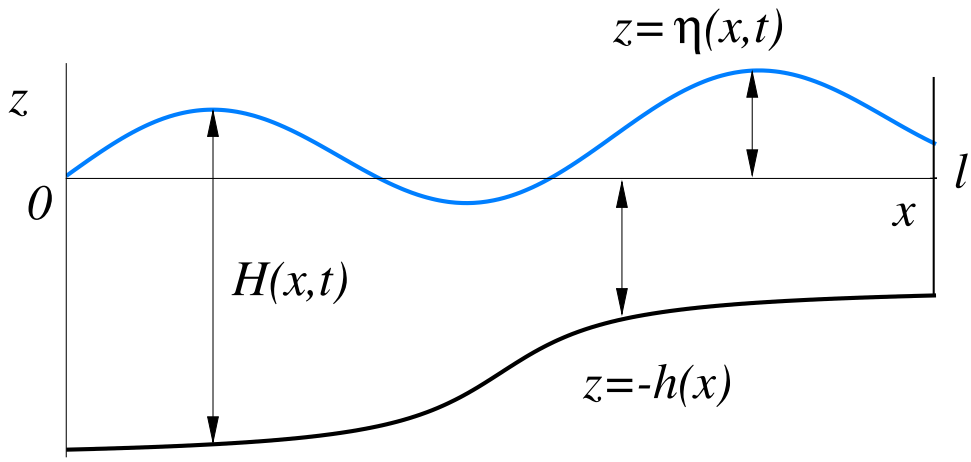


Figure 5. *Sketch of the fluid domain in shallow water flows.*

where the matrices $\mathscr{L}$, $\mathscr{R}$ and $\mathscr{D}$ are defined as

$$\mathscr{L} = \frac{1}{c^2}\begin{pmatrix} -\lambda_2 & 1 \\ -\lambda_1 & 1 \end{pmatrix}, \qquad \mathscr{R} = \frac{c}{2}\begin{pmatrix} -1 & 1 \\ -\lambda_1 & \lambda_2 \end{pmatrix}, \qquad \mathscr{D} = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}.$$

Here $\lambda_{1,2} = v \mp c$ ($c = \sqrt{gH}$ being the speed of gravity waves) are eigenvalues of the JACOBIAN $\mathscr{A}(\boldsymbol{v})$. They are real and distinct provided that $H(q,t) > 0$. Mathematically it implies the strict hyperbolicity property for the system (4.1).

4.1. Finite volume discretization

The reference domain Q is discretized into N equal control volumes $\left[q_j, q_{j+1}\right]$ of the width $\Delta q \equiv \frac{1}{N}$. The nodes x_j^n are obtained as images of uniform nodes under the mapping $x(q,t)$:

$$x_j^n = x(q_j, t^n).$$

In the predictor-corrector scheme on the first step we evaluate the flux vector in cell centers:

$$\hat{\boldsymbol{f}}_{j+1/2} = \frac{\boldsymbol{f}_j^n + \boldsymbol{f}_{j+1}^n}{2} - \frac{\tau}{2}\left[\frac{1}{J}\mathscr{R}\cdot\mathcal{D}\cdot\bar{\mathscr{D}}\cdot(\bar{\mathscr{D}}\cdot\mathscr{P} - \mathscr{L}\cdot\mathbf{G})\right]_{j+1/2}^n, \qquad \mathscr{P} := \mathscr{L}\cdot\boldsymbol{v}_q.$$

Vector $\mathscr{P}$ is an analogue* of the RIEMANN's invariants in the nonlinear case. The corrector step is based on the conservative form of the equation:

$$\frac{(J\boldsymbol{v})_j^{n+1} - (J\boldsymbol{v})_j^n}{\tau} + \frac{\left(\hat{\boldsymbol{f}} - (x_t\boldsymbol{v})^n\right)_{j+1/2} - \left(\hat{\boldsymbol{f}} - (x_t\boldsymbol{v})^n\right)_{j-1/2}}{\Delta q} = \mathbf{G}_j^*, \tag{4.2}$$

*The analogy is understood in the following sense: if we linearize our problem, the components of vector $\mathscr{P}$ will provide *exactly* two RIEMANN's invariants.

where τ is the time step and the following discretizations are used

$$\mathbf{G}^{n}_{j+1/2} = \begin{pmatrix} 0 \\ g\,(H\,h_q)^{n}_{j+1/2} \end{pmatrix}, \quad \mathbf{G}^{*}_{j} = \begin{pmatrix} 0 \\ g(Hh_q)^{n+1/2}_{j} \end{pmatrix}, \quad \mathcal{D}^{n}_{j+1/2} = \begin{pmatrix} 1+\theta^{1,n}_{j+1/2} & 0 \\ 0 & 1+\theta^{2,n}_{j+1/2} \end{pmatrix}.$$

The total water depth H and local depth gradient h_q have to be computed in the following way:

$$H^{n+1/2}_{j} = \frac{H^{n+1}_{j+1} + H^{n+1}_{j-1} + H^{n}_{j+1} + H^{n}_{j-1}}{4}, \; h^{n+1/2}_{q,j} = \frac{h^{n+1}_{j+1} - h^{n+1}_{j-1} + h^{n}_{j+1} - h^{n}_{j-1}}{4\Delta q}, \; h^{n}_{q,\,j+1/2} = \frac{h^{n}_{j+1} - h^{n}_{j}}{\Delta q}.$$

Despite the fact that $H^{n+1}_{j\pm1}$ is present in the formulas above, the scheme remains explicit, since we compute first $\{H^{n+1}_{j}\}_{j=1}^{N-1}$ from the mass conservation and, then, this value is used in the correction of the momentum conservation equation.

The matrix $\bar{\mathscr{D}}$ plays the rôle of $\bar{a}$ (the wave speed relative to the grid nodes) in the scalar case:

$$\bar{\mathscr{D}} \coloneqq \mathscr{D}^{n}_{j+1/2} - x^{n}_{t,\,j+1/2}\cdot\mathbb{I}, \qquad \mathscr{D}^{n}_{j+1/2} = \begin{pmatrix} \lambda^{n}_{1,\,j+1/2} & 0 \\ 0 & \lambda^{n}_{2,\,j+1/2} \end{pmatrix}, \qquad \mathbb{I} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

Here $\lambda^{n}_{k,\,j+1/2}$, $k = 1, 2$ are eigenvalues of the JACOBIAN matrix $\mathscr{A}^{n}_{j+1/2}$:

$$\mathscr{A}^{n}_{j+1/2} = \begin{pmatrix} 0 & 1 \\ -v^{n}_{j}v^{n}_{j+1} + gH^{n}_{j+1/2} & 2v^{n}_{j+1/2} \end{pmatrix}, \qquad (\lambda_{1,2})^{n}_{j+1/2} = (v \mp c)^{n}_{j+1/2},$$

the local gravity wave speed $c^{n}_{j+1/2}$ can be computed analytically:

$$c^{n}_{j+1/2} = \sqrt{(v^{n}_{j+1/2})^2 - v^{n}_{j}v^{n}_{j+1} + gH^{n}_{j+1/2}} \geqslant \sqrt{gH^{n}_{j+1/2}} > 0.$$

This choice of the JACOBIAN $\mathscr{A}$ discretization is dictated by the requirement to preserve the compatibility condition $\boldsymbol{f}_q = \mathscr{A}(\boldsymbol{v})\,\boldsymbol{v}_q$ at the discrete level as well. It is analogous to a similar algebraic condition on the JACOBIAN in the ROE scheme [73], but substantially different in details.

Finally, the scheme parameter $\theta^{k}_{j+1/2}$ is computed according to the same strategy as above, with the only difference being that it is computed separately for each component $k = 1, 2$:

$$\theta^{k}_{j+1/2} = \begin{cases} 0, & \left|\tilde{g}^{k}_{j+1/2}\right| \leqslant \left|\tilde{g}^{k}_{j+1/2-s_k}\right| \text{ and } \tilde{g}^{k}_{j+1/2}\cdot\tilde{g}^{k}_{j+1/2-s_k} \geqslant 0, \\ \theta^{k}_{0,j+1/2}(1 - \xi^{k}_{j+1/2}), & \left|\tilde{g}^{k}_{j+1/2}\right| > \left|\tilde{g}^{k}_{j+1/2-s_k}\right| \text{ and } \tilde{g}^{k}_{j+1/2}\cdot\tilde{g}^{k}_{j+1/2-s_k} \geqslant 0, \\ \theta^{k}_{0,j+1/2}, & \tilde{g}^{k}_{j+1/2}\cdot\tilde{g}^{k}_{j+1/2-s_k} < 0, \end{cases} \tag{4.3}$$

where $\xi^{k}_{j+1/2} \coloneqq \frac{\tilde{g}^{k}_{j+1/2-s_k}}{\tilde{g}^{k}_{j+1/2}}$, $s_k \equiv s^{n}_{k,j+1/2} \coloneqq \operatorname{sign}\big(\bar{\lambda}^{n}_{k,j+1/2}\big)$. Here $\bar{\lambda}^{n}_{k,j+1/2} = (\lambda_k - x_t)^{n}_{j+1/2}$ are simply the diagonal elements of the matrix $\bar{\mathscr{D}}$. The coefficient $\theta^{k}_{0,j+1/2}$ is related to the local CFL number as

$$\theta^{k}_{0,j+1/2} \coloneqq \frac{1}{C^{n}_{k,\,j+1/2}} - 1, \qquad C^{n}_{k,\,j+1/2} \coloneqq \frac{\tau}{\Delta q}\left(\frac{\left|\bar{\lambda}_k\right|}{J}\right)^{n}_{j+1/2}.$$

Finally, we have to specify how to compute the indicators

$$\tilde{g}^k_{j+1/2} \coloneqq \left(\left|\bar{\lambda}_k\right| (1 - C_k)\, \widetilde{p}^k\right)^n_{j+1/2},$$

where $\widetilde{p}^k$, $k = 1, 2$ are components of the following vector:

$$\widetilde{\mathscr{P}}^n_{j+1/2} \coloneqq \frac{1}{(c^n_{j+1/2})^2} \begin{pmatrix} -c\,\eta_q + H\,v_q \\ c\,\eta_q + H\,v_q \end{pmatrix}^n_{j+1/2}. \tag{4.4}$$

The fully discrete scheme is stable under the following CFL-type restriction [19] on the time step τ:

$$\max_{k,\,j} \left\{ C^n_{k,\,j+1/2} \right\} \leqslant 1\,.$$

Remark 2. *We would like to explain better the motivation behind the choice of the vector* $\widetilde{\mathscr{P}}$ *(*cf. *Equation* (4.4)*) in our scheme. This vector appears in the choice of the scheme parameter* (4.3)*, whose rôle consists in ensuring the monotonicity of the* Riemann*'s invariants. First of all, let us have a look at the components of vector* $\mathscr{P}$ *:*

$$\mathscr{P}^n_{j+1/2} = \left(\mathscr{L}\cdot \boldsymbol{v}_q\right)^n_{j+1/2} = \underbrace{\frac{1}{(c^n_{j+1/2})^2} \begin{pmatrix} -c\,\eta_q + H\,v_q \\ c\,\eta_q + H\,v_q \end{pmatrix}^n_{j+1/2}}_{\equiv\ \widetilde{\mathscr{P}}^n_{j+1/2}} + \left(\frac{h_q}{c}\right)^n_{j+1/2} \begin{pmatrix} -1 \\ 1 \end{pmatrix}.$$

We can see that vector $\mathscr{P}$ *is a sum of two vectors: the first addend depends on the solution gradient* (η_q, v_q)*, the second one depends on the bathymetry gradient* h_q*. We would like that the scheme parameters* $\theta^k_{j+1/2}$ *depend on the solution and not on the bathymetry. Thus, we made a choice to include only the first addend in the vector* $\widetilde{\mathscr{P}}$*.*

4.1.1 Well-balanced property

The following property can be shown for the scheme presented above:

Theorem 1. *Assume that we have a fixed, but possibly non-uniform, grid* $\{x^n_j\}$*. If on the* n^{th} *time layer we have the* lake at rest state, i.e.

$$\eta^n_j \equiv 0\,, \qquad v^n_j \equiv 0\,, \tag{4.5}$$

then, the predictor–corrector scheme described above will preserve this state at the $(n+1)^{th}$ *time layer.*

Proof. Since $x_{t,j} \equiv 0$, $\forall j$ and taking into account the definition (4.5) of the lake at rest state, we have:

$$\left(\bar{\mathscr{D}}\cdot\mathscr{P} - \mathscr{L}\cdot\mathbf{G}\right)^n_{j+1/2} \equiv 0.$$

Consequently, the predictor step leads

$$\hat{\boldsymbol{f}}_{j+1/2} = \frac{\boldsymbol{f}^n_j + \boldsymbol{f}^n_{j+1}}{2}.$$

Then, from the obvious identities $h_j^{n+1} \equiv h_j^n$, $J_j^{n+1} \equiv J_j^n$ and the way how we compute $\mathbf{G}_j^*$, it follows that $\eta_j^{n+1} \equiv 0$, $v_j^{n+1} \equiv 0$. Thus, the lake at rest state is preserved at the next time layer as well. □

The well-balanced property is absolutely crucial to compute qualitatively correct numerical solutions to conservation (balance) laws [42]. The Theorem 1 shows that the discretization proposed above does possess this property regardless some complications coming from the grid motion.

4.1.2 Conservation property on moving grids

For conservation laws with source terms (the so-called *balance laws*), the conservation property of a numerical scheme is understood in the sense that the *fully* discrete solution verifies some thoroughly chosen finite difference analogues of continuous balance laws. For the NSWE we have the conservation of mass and the balance of horizontal (depth-averaged) momentum. Please, note that on a flat bottom the balance of momentum becomes a conservation law as well. The conservation property is crucial as well to compute qualitatively correct numerical solutions [42].

On moving grids we have an additional difficulty: the computation of the speeds of mesh nodes has to be consistent with the calculation of the cell volumes. If it is done in the right way, the discrete geometric conservation law will be verified as well [85]. In 1D it can be written as

$$J_j^{n+1} \;=\; J_j^n \;+\; \tau\, x_{tq,j}^n. \tag{4.6}$$

We shall show below that other schemes can by put into the conservative form similar to (4.2). The peculiarity here consists in the fact that the Jacobian matrix $\mathscr{A}$ is non-constant, in contrast to the linear advection equation, for which any reasonable scheme is conservative (see also Remark 3 below about a nonlinear scalar equation). Moreover, the moving mesh is another complication, which makes this task rather non-trivial. For example, the scheme parameter $\theta_{j\pm1/2}^k$ can be chosen as

$$\theta_{j\pm1/2}^k \;=\; \theta_{0,j\pm1/2}^k, \qquad k \;=\; 1,2.$$

This choice corresponds to the first order upwind scheme (as it was discussed above in Section 2.1). So, the scheme (4.2) can be rewritten as

$$\frac{(J\boldsymbol{v})_j^{n+1} \;-\; (J\boldsymbol{v})_j^n}{\tau} \;+$$
$$\frac{1}{h}\Bigg[\Bigg(\frac{\boldsymbol{f}_{j+1}^n + \boldsymbol{f}_j^n}{2} \;-\; \frac{\left(\bar{\mathscr{A}}^+ - \bar{\mathscr{A}}^-\right)_{j+1/2}^n}{2}(\boldsymbol{v}_{j+1}^n - \boldsymbol{v}_j^n) \;-\; (x_t\,\boldsymbol{v})_{j+1/2}^n \;+\; \frac{h}{2}(\mathscr{RSL}\mathbf{G})_{j+1/2}^n\Bigg) \;-$$
$$\Bigg(\frac{\boldsymbol{f}_j^n + \boldsymbol{f}_{j-1}^n}{2} \;-\; \frac{\left(\bar{\mathscr{A}}^+ - \bar{\mathscr{A}}^-\right)_{j-1/2}^n}{2}(\boldsymbol{v}_j^n - \boldsymbol{v}_{j-1}^n) \;-\; (x_t\,\boldsymbol{v})_{j-1/2}^n \;+\; \frac{h}{2}(\mathscr{RSL}\mathbf{G})_{j-1/2}^n\Bigg)\Bigg] = \mathbf{G}_j^*,$$

where $\bar{\mathscr{A}}^{\pm} = \mathscr{R}\bar{\Lambda}^{\pm}\mathscr{L}$, $\bar{\lambda}_k^{\pm} = \frac{1}{2}(\bar{\lambda}_k \pm |\bar{\lambda}_k|)$ and

$$\mathcal{S} = \begin{pmatrix} \operatorname{sign}(\lambda_1) & 0 \\ 0 & \operatorname{sign}(\lambda_2) \end{pmatrix}, \qquad \bar{\Lambda}^{\pm} = \begin{pmatrix} \bar{\lambda}_1^{\pm} & 0 \\ 0 & \bar{\lambda}_2^{\pm} \end{pmatrix}.$$

One can see also that the following identities hold:

$$\bar{\Lambda} \equiv \bar{\Lambda}^{+} + \bar{\Lambda}^{-}, \qquad \bar{\mathscr{A}}^{+} + \bar{\mathscr{A}}^{-} \equiv \bar{\mathscr{A}} \equiv \mathscr{A} - x_t \mathbb{I}.$$

By using these identities and the discrete geometric conservation law (4.6), one can derive also the following equivalent form of the scheme (4.2):

$$\frac{\boldsymbol{v}_j^{n+1} - \boldsymbol{v}_j^n}{\tau} + \frac{1}{J_j^{n+1}}\left[\left(\mathscr{A}^{+}\boldsymbol{v}_q\right)_{j-1/2}^n + \left(\mathscr{A}^{-}\boldsymbol{v}_q\right)_{j+1/2}^n\right] = \frac{1}{J_j^{n+1}}\left[\mathbf{G}_j^{*} - \tfrac{1}{2}\left((\mathscr{R}\mathcal{S}\mathscr{L}\mathbf{G})_{j+1/2}^n - (\mathscr{R}\mathcal{S}\mathscr{L}\mathbf{G})_{j-1/2}^n\right)\right].$$

The choice (4.3) of the scheme parameter $\theta_{j\pm1/2}^k$ leads to a conservative scheme as well, but the underlying computations (to show it) are less elegant.

Remark 3. *In order to illustrate even better the interplay between conservative and non-conservative forms, let us take again the inviscid* Burgers–Hopf *equation [61]:*

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

For the sake of simplicity let us discretize this equation on a uniform grid using the following non-conservative scheme:

$$\frac{u_j^{n+1} - u_j^n}{\tau} + \frac{u_j^n + u_{j-1}^n}{2} \cdot \frac{u_j^n - u_{j-1}^n}{\Delta x} = 0.$$

It is not difficult to see that this non-conservative scheme is equivalent to the following conservative upwind scheme:

$$\frac{u_j^{n+1} - u_j^n}{\tau} + \frac{1}{\Delta x}\left(\frac{(u_j^n)^2}{2} - \frac{(u_{j-1}^n)^2}{2}\right) = 0.$$

However, if one takes a different non-conservative scheme[*]

$$\frac{u_j^{n+1} - u_j^n}{\tau} + u_j^n \cdot \frac{u_j^n - u_{j-1}^n}{\Delta x} = 0, \tag{4.7}$$

it is straightforward to see that one cannot transform it in a conservative form. Moreover, one can even show that scheme (4.7) is not converging. So, the conservation and convergence properties are intrinsically related.

[*]The difference is in the estimation of propagation speed, *cf.* u_j^n *vs.* $\frac{1}{2}(u_j^n + u_{j-1}^n)$.

4.2. Numerical results

In this section we present two test cases of complex wave propagation in shallow water equations. The first test case is a validation against an implicit analytic solution, derived below specifically for this purpose, while the second test case is related to real world applications.

4.2.1 Exact solution derivation

In order to validate the numerical scheme, we derive an exact solution to nonlinear shallow water equations which will serve as a reference solution. We already saw this particular solution in the literature [90], however, the derivation procedure is published for the first time to our knowledge.

First of all, we assume that the bottom is flat, *i.e.* $h(x) = h_0 = \text{const} > 0$ and we consider an initial value problem on the real line:

$$\eta(x,0) = \eta_0(x), \qquad u(x,0) = u_0(x).$$

The nonlinear shallow water equations are rewritten using the Riemann invariants [88]:

$$r_t + \frac{3r + s}{4} r_x = 0, \qquad s_t + \frac{r + 3s}{4} s_x = 0,$$

where

$$r := u - 2c, \qquad s := u + 2c, \qquad c := \sqrt{gH}.$$

We seek for a particular solution in the form of an r-wave propagating in the leftward direction. In other words we assume $s = s_0 = \text{const}$. Assuming that at the infinity the water is at rest, we deduce that $s_0 = 2c_0 = 2\sqrt{gh_0}$. Then, from formula $u + 2c = s_0$ we deduce that the speed $u(x,t)$ is related to the total depth:

$$u(x,t) = s_0 - 2c = 2c_0 - 2\sqrt{gH(x,t)}. \tag{4.8}$$

Consequently, taking the limit $t \to 0$ we have a similar relation between the initial conditions:

$$u_0(x) = 2c_0 - 2\sqrt{g(\eta_0(x) + h_0)}.$$

If we introduce a new variable

$$p(x,t) := \frac{3r(x,t) + s_0}{4}$$

the governing equation for the r-invariant becomes simply the Burgers–Hopf equation:

$$p_t + p p_x = 0.$$

The solution can be obtained using the method of characteristics by the following implicit formula [74, 92]:

$$p(x,t) = p_0\big(x - p(x,t)t\big),$$

where $p_0(x)$ is the initial condition

$$p_0(x) \;=\; \frac{3r(x,0) \;+\; s_0}{4} \;=\; 2c_0 \;-\; 3\sqrt{g(\eta_0(x)+h_0)}.$$

Consequently, the solution $p(x,t)$ can be obtained at any space location x and time instance t by solving the following implicit equation

$$p(x,t) \;=\; 2c_0 \;-\; 3\sqrt{g(\eta_0\big(x-p(x,t)\,t\big)+h_0)}. \tag{4.9}$$

Once, $p(x,t)$ is found, the free surface elevation $\eta(x,t)$ can be reconstructed by using this formula

$$\eta(x,t) \;=\; \left(\frac{2c_0 \;-\; p(x,t)}{3\sqrt{g}}\right)^2 \;-\; h_0, \tag{4.10}$$

and the flow speed $u(x,t)$ from (4.8).

Remark 4. *Please, note that this method works only until the gradient catastrophe occurs at some time instance $t = t^*$, which depends on the initial condition.*

Remark 5. *In order to guarantee the solvability of equation (4.9), it is enough to check that the following function*

$$f(p;x,t) \;=\; p \;+\; 3\sqrt{g(\eta_0\big(x-pt\big)+h_0)} \;-\; 2c_0$$

is monotonic in p for $0 < t < t^*$.

4.2.2 Validation

Using the method described above, we obtain the analytical solution for the following initial value problem:

$$\eta_0(x) \;=\; \begin{cases} \frac{a}{2}\Big[1 \;+\; \cos\big(\frac{2\pi(x \;-\; x_w)}{\lambda}\big)\Big], & |x \;-\; x_w| \;\leqslant\; \lambda/2, \\ 0, & |x \;-\; x_w| \;>\; \lambda/2, \end{cases}$$

where x_w is the wave crest position and a is the wave amplitude. For $0 < t < t^*$ the profile $\eta_0(x)$ will propagate to the left with the speed c_0 preserving its wavelength λ and amplitude a. However, the gradient catastrophe will inevitably take place since the characteristics $\frac{\mathrm{d}x}{\mathrm{d}t} = p_0(x)$ departing from the interval $(x_w - \lambda/2,\; x_w)$ form a converging beam. On the other hand, the characteristics from $(x_w,\; x_w + \lambda/2)$ diverge. Thus, the solution looks like a compression wave followed by a rarefaction wave. With the parameters specified in Table 4 the gradient catastrophe takes place about $t = t^* \approx 5.57$. Numerical results are shown in Figure 6 right before the gradient catastrophe occurs. The adaptive grid was constructed using the following monitor function, which contains an additional term with respect to (2.11)

$$w(x,t) \;=\; 1 \;+\; \alpha_0\,|\eta(x,t)| \;+\; \alpha_1\,|\eta_x(x,t)|, \tag{4.11}$$

Parameter	*Value*
Computational domain length, ℓ	40.0
Number of nodes, N	100
Final simulation time, T_f	5.0
CFL number, C	0.95
Gravity acceleration, g	9.81
Initial wave amplitude, a	0.2
Wave crest initial position, x_w	30.0
Initial wavelength, λ	10.0
Undisturbed water depth, h_0	1.0
Monitor function parameter, α_0	10.0
Monitor function parameter, α_1	10.0
Grid diffusion parameter, β	5.0
Smoothing parameter, σ	5.0

Table 4. *Numerical parameters used in validations of the nonlinear shallow water equations solver.*

which is discretized as

$$w^n_{j+1/2} = 1 + \alpha_0 \left| \eta^n_{j+1/2} \right| + \alpha_1 \frac{\left| \eta^n_{j+1} - \eta^n_j \right|}{x^n_{j+1} - x^n_j}. \tag{4.12}$$

Thin solid line in Figure 6 represents the exact solution computed by solving equation (4.9) as it was explained in the previous Section 4.2.1. The agreement between these two profiles is exemplary. On the right panel (b) one can see how the nodes follow the wave propagation and how their density increases around the steep gradient. On the other hand, the use of a fixed grid leads to a noticeable smearing of the numerical solution.

The exact solution presented in the previous Section 4.2.1 can be used in order to study the absolute accuracy of proposed numerical methods. To this end we introduce two discrete norms:

$$\| f_h - \boldsymbol{f}_{\text{ref}} \|_\infty := \max_{1 \leqslant j \leqslant N-1} \left| f^n_j - f(x_j, T_f) \right|,$$

$$\| f_h - \boldsymbol{f}_{\text{ref}} \|_2 := \left(\sum_{j=1}^{N-1} \left(f^n_j - f(x_j, T_f) \right)^2 \cdot \frac{x^n_{j+1} - x^n_{j-1}}{2} \right)^2,$$

where n is the time step number corresponding to the final simulation time $t = T_f$. Here f denotes the free surface elevation η or the depth-averaged horizontal velocity u. The reference solution for u and η is provided by formulas (4.8) and (4.10) correspondingly. We would like to mention that the above definition of the norm $\|\cdot\|_2$ is valid for the fixed and moving grids. The computational results are reported for the fixed and moving grids in

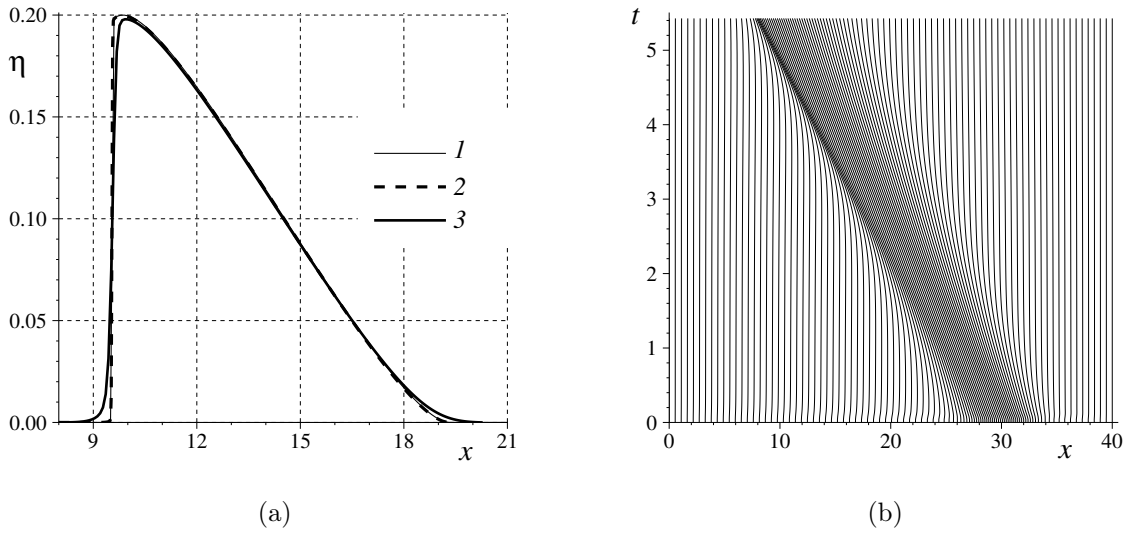


Figure 6. *Shock wave formation in nonlinear shallow water equations: (a) comparison of different profiles (thin solid line (1) is the exact solution, (2) is the numerical solution on the moving mesh, (3) is the solution on the fixed uniform mesh); (b): trajectories of grid nodes in space-time. Notice that the panel (a) shows only a zoom of the computational domain.*

Tables 5 and 6 respectively. All the parameters in these simulations were fixed and taken from Table 4 (except the parameter N, which was varied). The second order convergence can be observed in Table 5. In general, the results obtained on a moving grid are more accurate than those on a fixed one as it follows from the comparison of corresponding cells in Tables 5 and 6. In the last column of these tables we report also the CPU times measured on a computer equiped with an Intel Pentium CPU 3.2 Ghz and 1Gb of RAM. The code was implemented in Fortran and compiled with Compaq Visual Fortran compiler (version 6.6c). In particular, for the finest grid ($N = 6\,400$) the overhead related to the grid motion represents less than 7% of the overall CPU time and the error (measured in the most stringent norm $\|\cdot\|_\infty$) is reduced by a factor of four. Thus, we may conclude that the application of moving grid methods is very interesting from the practical point of view.

We underline the fact that even better results could be obtained if one optimizes the monitor function parameters $\alpha_{0,1}$ along with mesh motion parameters β and σ. In our numerical simulations we just took some *reasonable* values. At the current stage of the development of these methods, it is difficult to say something on the optimal values of these parameters *a priori*. Finally, we would like to mention that this test case is relatively difficult for both analytical and numerical methods since we deal with a nonlinear system of hyperbolic equations and at the final time $t = T_f$ the wave has almost developed a discontinuity (from a smooth initial condition). Thus, we operate at the edge of validity of the analytical solution (4.8), (4.10).

Table 5. *Numerical solution error measured at the final simulation time $t = T_f$ on a fixed grid. The reference solution is given be formulas (4.8) and (4.10). All numerical parameters for this simulation are reported in Table 4.*

N	η		u		CPU time, s
	$\|\eta_h - \eta_{\text{ref}}\|_\infty$	$\|\eta_h - \eta_{\text{ref}}\|_2$	$\|u_h - u_{\text{ref}}\|_\infty$	$\|u_h - u_{\text{ref}}\|_2$	
100	$0.388 \cdot 10^{-1}$	$0.289 \cdot 10^{-1}$	$0.363 \cdot 10^{-1}$	$0.270 \cdot 10^{-1}$	$< 10^{-2}$
200	$0.245 \cdot 10^{-1}$	$0.127 \cdot 10^{-1}$	$0.229 \cdot 10^{-1}$	$0.118 \cdot 10^{-1}$	0.03
400	$0.104 \cdot 10^{-1}$	$0.451 \cdot 10^{-2}$	$0.977 \cdot 10^{-2}$	$0.424 \cdot 10^{-2}$	0.20
800	$0.418 \cdot 10^{-2}$	$0.144 \cdot 10^{-2}$	$0.394 \cdot 10^{-2}$	$0.136 \cdot 10^{-2}$	1.36
1600	$0.128 \cdot 10^{-2}$	$0.393 \cdot 10^{-3}$	$0.122 \cdot 10^{-2}$	$0.373 \cdot 10^{-3}$	9.05
3200	$0.353 \cdot 10^{-3}$	$0.103 \cdot 10^{-3}$	$0.337 \cdot 10^{-3}$	$0.981 \cdot 10^{-4}$	71.08
6400	$0.885 \cdot 10^{-4}$	$0.257 \cdot 10^{-4}$	$0.843 \cdot 10^{-4}$	$0.245 \cdot 10^{-4}$	573.11

Table 6. *Numerical solution error measured at the final simulation time $t = T_f$ on a moving grid. The reference solution is given be formulas (4.8) and (4.10). All numerical parameters for this simulation are reported in Table 4.*

N	η		u		CPU time, s
	$\|\eta_h - \eta_{\text{ref}}\|_\infty$	$\|\eta_h - \eta_{\text{ref}}\|_2$	$\|u_h - u_{\text{ref}}\|_\infty$	$\|u_h - u_{\text{ref}}\|_2$	
100	$0.125 \cdot 10^{-1}$	$0.907 \cdot 10^{-2}$	$0.123 \cdot 10^{-1}$	$0.900 \cdot 10^{-2}$	0.03
200	$0.591 \cdot 10^{-2}$	$0.415 \cdot 10^{-2}$	$0.586 \cdot 10^{-2}$	$0.413 \cdot 10^{-2}$	0.20
400	$0.266 \cdot 10^{-2}$	$0.170 \cdot 10^{-2}$	$0.264 \cdot 10^{-2}$	$0.170 \cdot 10^{-2}$	1.13
800	$0.897 \cdot 10^{-3}$	$0.608 \cdot 10^{-3}$	$0.894 \cdot 10^{-3}$	$0.606 \cdot 10^{-3}$	5.76
1600	$0.317 \cdot 10^{-3}$	$0.215 \cdot 10^{-3}$	$0.316 \cdot 10^{-3}$	$0.214 \cdot 10^{-3}$	27.66
3200	$0.881 \cdot 10^{-4}$	$0.566 \cdot 10^{-4}$	$0.878 \cdot 10^{-4}$	$0.563 \cdot 10^{-4}$	130.00
6400	$0.221 \cdot 10^{-4}$	$0.142 \cdot 10^{-4}$	$0.220 \cdot 10^{-4}$	$0.142 \cdot 10^{-4}$	612.60

4.2.3 Solitary wave run-up

As the last test case we consider a practically important problem of a solitary wave run-up on a sloping beach. The main difficulty here consists in the fact that the fluid domain may vary in time and its evolution is not known *a priori*. It has to be determined dynamically during the simulation. The wetting/drying algorithm was described in detail in our previous study [56]. Here we focus mainly on the mesh motion algorithm behaviour.

Consider a closed 1D channel with the bathymetry prescribed by the following function

$$z = -h(x) = \begin{cases} h_0 - x\tan\theta & 0 \leqslant x \leqslant x_s, \\ -h_0, & x_s \leqslant x \leqslant \ell, \end{cases}$$

where θ is the sloping beach angle. The variable bathymetry region is located leftwards from $x_s = 2\cot\theta \approx 38.16$. At the initial time the shoreline is located at $x_{00} = \cot\theta \approx 19.08$.

The initial condition is a solitary wave prescribed by the following formulas (which come from the analytical solitary wave solution to the Serre equations, see [31] for example):

$$\eta(x,0) = a\,\mathrm{sech}^2\left[\frac{\sqrt{3ga}}{2h_0\sqrt{g(h_0+a)}}\,(x - x_{0,w})\right],$$

$$u(x,0) = -\sqrt{g(h_0+a)}\,\frac{\eta(x,0)}{h_0 + \eta(x,0)},$$

Variables a and $x_{0,w} = x_s + 30 \approx 68.16$ are the initial solitary wave amplitude and position correspondingly. The right boundary is located at $\ell = x_{0,w} + 30 \approx 98.16$. All numerical parameters used in this experiment are provided in Table 7.

The considered situation is complex enough so that the analytical solution is not available. However, we can simulate it numerically with the predictor–corrector scheme on moving grids presented above. The wave run-up process simulated *in silico* is depicted in Figure 7(a), while the nodes motion is illustrated in the right panel (b). Please, note that only every fifth node is depicted in Figure 7(b) in order to improve the visibility of the image. The rightmost node is fixed (*i.e.* its trajectory is a straight line), while th leftmost node's trajectory coincides with the shoreline motion. One can see that even on domains with variable spatial extent, the mesh motion algorithm is robust enough and it puts the resolution where it is needed.

The monitor function used in this computation is

$$w(x,t) = 1 + \alpha\,|\eta(x,t)|\,. \tag{4.13}$$

It is important to use the wave elevation $\eta(x,t)$ instead of the total water depth $H(x,t)$ because of the solutions such as the lake at rest state. In these situations the total depth H varies in space, while the free surface elevation $\eta(x,t) \equiv 0$. Thus, the usage of the monitor function (4.13) leads to the uniform grid (since $w(x,t) \equiv 1$) and, trivially, to the well-balanced property shown in Theorem 1.

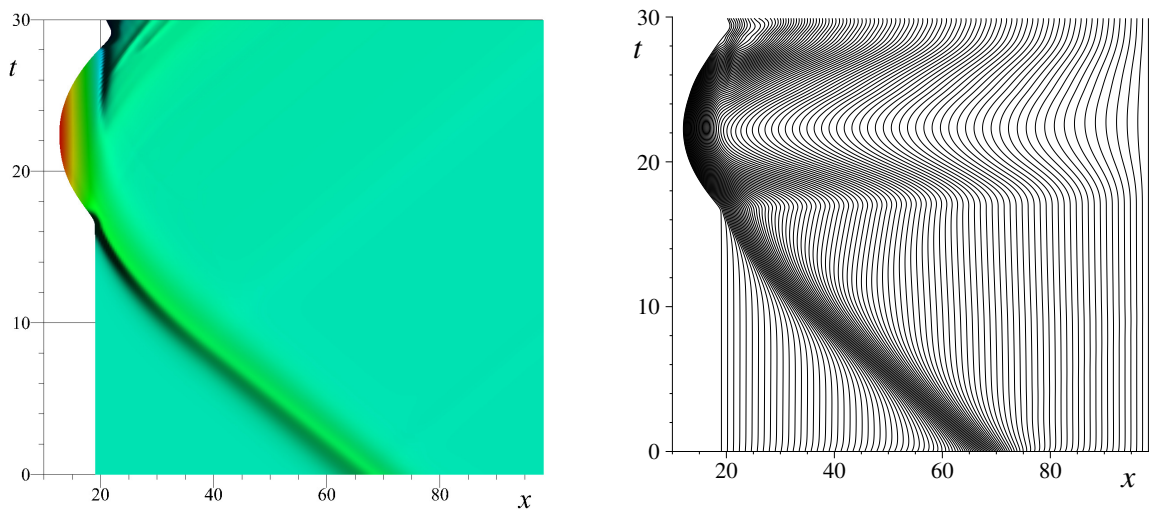
Some further validation tests and numerical experiments with this moving grid technique were presented in our preceding work [56] (without giving many details about the algorithm in use).

5. Some open problems

Only in very simple problems (linear and steady) one can study theoretically the choice of the optimal monitor function. Recently we performed such an analysis for a singularly perturbed Sturm–Liouville problem [51]. In particular, we showed that a judicious

Parameter	*Value*
Computational domain length, ℓ	98.16
Number of nodes, N	1000
Final simulation time, T_f	30.0
CFL number, C	0.95
Gravity acceleration, g	9.81
Initial wave amplitude, a/h_0	0.05
Wave crest initial position, $x_{0,w}$	68.16
Beach slope position, x_s	38.16
Beach slope, θ	3°
Undisturbed water depth, h_0	1.0
Monitor function parameter, α	60.0
Grid diffusion parameter, β	20.0
Smoothing parameter, σ	10.0

Table 7. *Numerical parameters used in a solitary wave run-up simulation in nonlinear shallow water equations.*



(a) (b)

Figure 7. *Solitary wave run-up simulation on a sloping beach with nonlinear shallow water equations: (a) free surface elevation evolution; (b): trajectories of grid nodes in space-time.*

choice of the monitor function allows to increase the order of convergence from the 2nd to the 4th order (the so-called *supraconvergence* phenomenon), even if the scheme is just 2nd order accurate on uniform meshes. So, the first open problem consists in providing a similar analysis for, at least, 1D unsteady or nonlinear problem. The method employed in [51] is suitable for smooth solutions. For discontinuous ones a different technique has to be proposed. Even a linear transport equation is not completely understood from the perspective of constructing a suitable moving grid. It is even possible that for every discretization scheme there is its own "optimal" monitor function. So, one choice clearly does not fit all possible schemes. Moreover, even if we choose the monitor function as in (4.13), it was shown in [51] that the error may depend non-monotonically on the coefficient α. Thus, even if the monitor function form is fixed, one has to set optimal values of coefficients in this function.

6. Discussion

The main conclusions and perspectives of this study are outlined below.

6.1. Conclusions

In the present work our goal was to present the moving grid methodology, which does not involve any interpolation procedure (thus, avoiding the unnecessary smearing of the numerical solution). We showed some reasonable choices of the monitor function, which are currently used in practice, but these choices have to be refined further in the forthcoming works. The current state-of-the-art in this field is described in [49].

In this manuscript we presented a detailed description of a family of predictor-corrector schemes up to the second order in space, which were generalized to the case of non-uniform moving meshes, while having all these properties *simultaneously*:

- Conservation in space;
- Second order accurate (in smooth regions);
- TVD property for scalar problems;
- Well-balanced character;
- Preservation of the discrete conservation law in cell centers and cell interfaces;
- Adaptivity in space and in time*;
- No interpolation between the meshes, thus, we do not add any unnecessary numerical diffusion;
- Implementation is trivial, despite all good properties listed above.

This is the main novelty and contribution of our study to the growing field of adaptive numerical methods. This method was illustrated on a number of relatively serious tests coming from linear, nonlinear and vectorial cases. Every time the gain of using a moving

*The time step is chosen the largest possible while meeting the stability requirement.

grid (with respect to the fixed one) was clearly visible. Consequently, we can only encourage the community to use these (or similar) adaptive strategies to improve the quality and efficiency of their numerical simulations. In particular, the field of tsunami wave modelling could greatly benefit from the moving grid nodes to improve the accuracy and computational efficiency of operational tsunami wave predictions.

6.2. Perspectives

The adaptive moving mesh technology presented in this study relies on the choice of the monitor function along with several free real parameters. Numerical experiments show that the method performance may depend crucially on the choice of these real numbers. Consequently, it would be highly desirable to obtain some theoretical estimations of the optimal choice of these parameters at least in the case of the simplest linear advection equation (2.1). It constitutes one of the main challenges raised by our study. In the absence of such theoretical indications, one might envisage extensive numerical studies in order to issue some practical recommendations on how to choose reasonable values of parameters. Most probably the case of nonlinear systems of conservation laws will remain essentially empiric in the foreseeable future due to the important theoretical difficulties in its analysis.

From a more practical point of view, the predictor-corrector schemes *on moving grids* presented in this study have to be generalized at least to two-dimensional case as it was done for the Godunov scheme in [5]. We already reported an implementation of moving grid technique in two space dimensions for an elliptic problem in [52]. Moreover, the spatial adaptivity has to be coupled with time-adaptive strategies such as the PI step size control [44] in order to achieve the full adaptivity.

Finally, there is another goal behind this study. Ultimately, we would like to generalize moving grid methods to some dispersive wave equations [31], which are going to be addressed with the operator splitting approach along the lines sketched in [55]. In this way, the hyperbolic part would be addressed with methods outlined above. The particularity of dispersive wave equations is that they possess localised solitary wave solutions. In other words, all the dynamics is concentrated in small portions of space, which move perpetually. Consequently, the grid redistribution techniques seem to be the natural choice to simulate complex solitonic gas dynamics [33].

Acknowledgments

This research was supported by RSCF project No 14-17-00219. D. Dutykh acknowledges the support of the CNRS under the PEPS InPhyNiTi project FARA and project N°EDC26179 — “Wave interaction with an obstacle” as well as the hospitality of the Institute of Computational Technologies of SB RAS during his visit in October 2015. D. Mitsotakis was supported by the Marsden Fund administered by the Royal Society of New Zealand. We would like to thank Prof. Laurent Gosse (CNR, Italy) for insightful

discussions on the topics covered in our study. Moreover, we would like to thank the anonymous Referee on our previous manuscript [56] who incited us by his comments to produce a study which explains in more details our grid motion technique.

References

[1] Alalykin, G. B., Godunov, S. K., Kireyeva, L. L., and Pliner, L. A. *Solution of One-Dimensional Problems in Gas Dynamics on Moving Grids*. Nauka, Moscow, 1970. 6

[2] Arvanitis, C., and Delis, A. I. Behavior of Finite Volume Schemes for Hyperbolic Conservation Laws on Adaptive Redistributed Spatial Grids. *SIAM J. Sci. Comput. 28*, 5 (jan 2006), 1927–1956. 6

[3] Arvanitis, C., Katsaounis, T., and Makridakis, C. Adaptive Finite Element Relaxation Schemes for Hyperbolic Conservation Laws. *ESAIM: Mathematical Modelling and Numerical Analysis 35*, 1 (2010), 17–33. 5

[4] Asselin, R. Frequency Filter for Time Integrations. *Monthly Weather Review 100*, 6 (jun 1972), 487–490. 14

[5] Azarenok, B. N., Ivanenko, S. A., and Tang, T. Adaptive Mesh Redistibution Method Based on Godunov's Scheme. *Commun. Math. Sci. 1*, 1 (2003), 152–179. 6, 36

[6] Babuska, I., Henshaw, W. D., Oliger, J. E., Flaherty, J. E., Hopcroft, J. E., and Tezduyar, T., Eds. *Modeling, Mesh Generation, and Adaptive Numerical Methods for Partial Differential Equations*, vol. 75 of *The IMA Volumes in Mathematics and its Applications*. Springer, New York, NY, 1995. 5

[7] Babuska, I., and Suri, M. The p and h-p Versions of the Finite Element Method, Basic Principles and Properties. *SIAM Review 36*, 4 (dec 1994), 578–632. 5

[8] Barakhnin, V. B., and Borodkin, N. V. The second order approximation TVD scheme on moving adaptive grids for hyperbolic systems. *Sib. Zh. Vychisl. Mat. 3*, 2 (2000), 109–121. 21

[9] Beckett, G., Mackenzie, J. A., Ramage, A., and Sloan, D. M. Computational Solution of Two-Dimensional Unsteady PDEs Using Moving Mesh Methods. *J. Comp. Phys. 182*, 2 (nov 2002), 478–495. 6, 12

[10] Berger, M. J., and Colella, P. Local adaptive mesh refinement for shock hydrodynamics. *J. Comp. Phys. 82*, 1 (may 1989), 64–84. 5

[11] Berger, M. J., and Oliger, J. Adaptive mesh refinement for hyperbolic partial differential equations. *J. Comp. Phys. 53*, 3 (mar 1984), 484–512. 5

[12] Bona, J. L., and Varlamov, V. Wave generation by a moving boundary. *Nonlinear partial differential equations and related analysis 371* (2005), 41–71. 7

[13] Breuss, M. About the Lax-Friedrichs scheme for the numerical approximation of hyperbolic conservation laws. *PAMM 4*, 1 (dec 2004), 636–637. 8

[14] Budd, C. J., Huang, W., and Russell, R. D. Adaptivity with moving grids. *Acta Numerica 18* (may 2009), 111–241. 6, 15

[15] Budd, C. J., and Williams, J. F. Moving Mesh Generation Using the Parabolic Monge-Ampère Equation. *SIAM J. Sci. Comput. 31*, 5 (jan 2009), 3438–3465. 12

[16] Burgers, J. M. A mathematical model illustrating the theory of turbulence. *Advances in Applied Mechanics 1* (1948), 171–199. 17

[17] Cao, W., Huang, W., and Russell, R. D. A Study of Monitor Functions for Two-Dimensional Adaptive Mesh Generation. *SIAM J. Sci. Comput. 20*, 6 (jan 1999), 1978–1994. 6, 12

[18] Chhay, M., Hoarau, E., Hamdouni, A., and Sagaut, P. Comparison of some Lie-symmetry-based integrators. *J. Comp. Phys. 230(5)* (2011), 2174–2188. 7

[19] Courant, R., Friedrichs, K., and Lewy, H. Über die partiellen Differenzengleichungen der mathematischen Physik. *Mathematische Annalen 100*, 1 (1928), 32–74. 8, 10, 25

[20] Courant, R., Isaacson, E., and Rees, M. On the solution of nonlinear hyperbolic differential equations by finite differences. *Comm. Pure Appl. Math. 5*, 3 (aug 1952), 243–255. 8

[21] Coyle, J. M., Flaherty, J. E., and Ludwig, R. On the stability of mesh equidistribution strategies for time-dependent partial differential equations. *J. Comp. Phys. 62*, 1 (jan 1986), 26–39. 14

[22] Dalrymple, R. A., Grilli, S. T., and Kirby, J. T. Tsunamis and challenges for accurate modeling. *Oceanography 19* (2006), 142–151. 4

[23] Dao, M. H., and Tkalich, P. Tsunami propagation modelling - a sensitivity study. *Nat. Hazards Earth Syst. Sci. 7* (2007), 741–754. 4

[24] Dar'in, N. A., Mazhukin, V. I., and Samarskii, A. A. A finite-difference method for solving the equations of gas dynamics using adaptive nets which are dynamically associated with the solution. *USSR Computational Mathematics and Mathematical Physics 28*, 4 (jan 1988), 164–174. 14

[25] Daripa, P. Iterative schemes and algorithms for adaptive grid generation in one dimension. *J. Comp. Phys. 100*, 2 (jun 1992), 284–293. 14

[26] Darmaev, L. M., and Liseikin, V. D. A method of construction of multidimensional adaptive grids. *Modelling in Mechanics 1*, 18 (1987), 49–58. 6

[27] Degtyarev, L. M., Drozdov, V. V., and Ivanova, T. S. The method of adaptive grids for the solution of singularly perturbed one dimensional boundary value problems. *Differ. Uravn. 23*, 7 (1987), 1160–1169. 6

[28] Degtyarev, L. M., and Ivanova, T. S. The adaptive-grid method in one-dimensional nonstationary convection-diffusion problems. *Differ. Uravn. 29*, 7 (1993), 1179–1192. 6, 14

[29] Dorfi, E. A., and Drury, L. O. Simple adaptive grids for 1 - D initial value problems. *J. Comp. Phys. 69*, 1 (mar 1987), 175–195. 14

[30] Dorodnitsyn, V. A. Finite Difference Models Entirely Inheriting Continuous Symmetry of Original Differential Equations. *Int. J. Mod. Phys. C 05*, 04 (aug 1994), 723–734. 7

[31] Dutykh, D., Clamond, D., Milewski, P., and Mitsotakis, D. Finite volume and pseudo-spectral schemes for the fully nonlinear 1D Serre equations. *Eur. J. Appl. Math. 24*, 05 (2013), 761–787. 33, 36

[32] Dutykh, D., and Mitsotakis, D. On the relevance of the dam break problem in the context of nonlinear shallow water equations. *Discrete and Continuous Dynamical Systems - Series B 13(4)* (2010), 799–818. 5

[33] Dutykh, D., and Pelinovsky, E. Numerical simulation of a solitonic gas in KdV and KdV-BBM equations. *Phys. Lett. A 378*, 42 (aug 2014), 3102–3110. 36

[34] Fedorenko, R. P. *Introduction to Computational Physics*. MIPT Press, Moscow, 1994. 13, 14

[35] Fuhrmann, J., Ohlberger, M., and Rohde, C. *Finite Volumes for Complex Applications VII - Elliptic, Parabolic and Hyperbolic Problems*. Springer, Berlin, 2014. 5

[36] Gasparin, S., Berger, J., Dutykh, D., and Mendes, N. An innovative method to determine optimum insulation thickness based on non-uniform adaptive moving grid. *Journal of the Brazilian Society of Mechanical Sciences and Engineering 41*, 4 (apr 2019), 173. 6

[37] George, D. L., and Leveque, R. J. Finite volume methods and adaptive refinement for global tsunami propagation and local inundation. *Sci. Tsunami Hazards 24(5)* (2006), 319. 5

[38] Glimsdal, S., Pedersen, G. K., Harbitz, C. B., and Løvholt, F. Dispersion of tsunamis: does it really matter? *Natural Hazards and Earth System Science 13*, 6 (jun 2013), 1507–1526. 4

[39] Godunov, S. K. A finite difference method for the numerical computation of discontinuous solutions of the equations of fluid dynamics. *Mat. Sb. 47* (1959), 271–290. 5

[40] Godunov, S. K. Reminiscences about Difference Schemes. *J. Comput. Phys. 153* (1999), 6–25. 5

[41] Godunov, S. K., Zabrodin, A., Ivanov, M. Y., Kraiko, A. N., and Prokopov, G. P. *Numerical solution of multidimensional problems of gas dynamics*. Nauka, Moscow, 1976. 6

[42] Gosse, L. *Computing Qualitatively Correct Approximations of Balance Laws: Exponential-Fit, Well-Balanced and Asymptotic-Preserving*, 1 ed., vol. 2 of *SIMAI Springer Series*. Springer Milan, Milano, 2013. 26

[43] Grilli, S. T., Ioualalen, M., Asavanant, J., Shi, F., Kirby, J. T., and Watts, P. Source Constraints and Model Simulation of the December 26, 2004, Indian Ocean Tsunami. *Journal of Waterway, Port, Coastal, and Ocean Engineering 133*, 6 (nov 2007), 414–428. 4

[44] Hairer, E., Nørsett, S. P., and Wanner, G. *Solving ordinary differential equations: Nonstiff problems*. Springer, 2009. 36

[45] Harten, A. High resolution schemes for hyperbolic conservation laws. *J. Comp. Phys. 49* (1983), 357–393. 19

[46] Houston, P., and Süli, E. hp-Adaptive Discontinuous Galerkin Finite Element Methods for First-Order Hyperbolic Problems. *SIAM J. Sci. Comput. 23*, 4 (jan 2001), 1226–1252. 5

[47] Huang, W., Ren, Y., and Russell, R. D. Moving Mesh Methods Based on Moving Mesh Partial Differential Equations. *J. Comp. Phys. 113*, 2 (aug 1994), 279–290. 6, 14

[48] Huang, W., and Russell, R. D. Adaptive mesh movement - the MMPDE approach and its applications. *J. Comp. Appl. Math. 128*, 1-2 (mar 2001), 383–398. 7

[49] Huang, W., and Russell, R. D. *Adaptive Moving Mesh Methods*, vol. 174 of *Applied Mathematical Sciences*. Springer New York, New York, NY, 2011. 6, 12, 14, 35

[50] Imamura, F. Simulation of wave-packet propagation along sloping beach by TUNAMI-code. In *Long-wave Runup Models*, H. Yeh, P. L.-F. Liu, and C. E. Synolakis, Eds. World Scientific, Singapore, 1996, pp. 231–241. 4

[51] Khakimzyanov, G., and Dutykh, D. On supraconvergence phenomenon for second order centered finite differences on non-uniform grids. *J. Comp. Appl. Math. 326* (dec 2017), 1–14. 6, 12, 33, 35

[52] Khakimzyanov, G. S., and Dutykh, D. Numerical Modelling of Surface Water Wave Interaction with a Moving Wall. *Commun. Comput. Phys. 23*, 5 (2018), 1289–1354. 7, 36

[53] Khakimzyanov, G. S., Dutykh, D., and Fedotova, Z. I. Dispersive shallow water wave modelling. Part III: Model derivation on a globally spherical geometry. *Commun. Comput. Phys. 23*, 2 (2018), 315–360. 4

[54] Khakimzyanov, G. S., Dutykh, D., and Gusev, O. Dispersive shallow water wave

modelling. Part IV: Numerical simulation on a globally spherical geometry. *Commun. Comput. Phys. 23*, 2 (2018), 361–407. 4, 5

[55] Khakimzyanov, G. S., Dutykh, D., Gusev, O., and Shokina, N. Y. Dispersive shallow water wave modelling. Part II: Numerical modelling on a globally flat space. *Commun. Comput. Phys. 23*, 1 (2018), 30–92. 6, 36

[56] Khakimzyanov, G. S., Shokina, N. Y., Dutykh, D., and Mitsotakis, D. A new run-up algorithm based on local high-order analytic expansions. *J. Comp. Appl. Math. 298* (may 2016), 82–96. 32, 33, 37

[57] Kirby, J. T., Shi, F., Tehranirad, B., Harris, J. C., and Grilli, S. T. Dispersive tsunami waves in the ocean: Model equations and sensitivity to dispersion and Coriolis effects. *Ocean Modelling 62* (feb 2013), 39–55. 4

[58] Knobloch, E., and Krechetnikov, R. Problems on Time-Varying Domains: Formulation, Dynamics, and Challenges. *Acta Appl. Math. 137*, 1 (jun 2015), 123–157. 7

[59] Krejic, N., Krunic, T., and Nedeljkov, M. Numerical verification of delta shock waves for pressureless gas dynamics. *J. Math. Anal. Appl. 345*, 1 (sep 2008), 243–257. 21

[60] Lax, P., and Wendroff, B. Systems of conservation laws. *Comm. Pure Appl. Math. 13*, 2 (may 1960), 217–237. 8

[61] LeVeque, R. J. *Numerical Methods for Conservation Laws*, 2 ed. Birkhäuser Basel, Basel, 1992. 9, 27

[62] Liseikin, V. D. The construction of structured adaptive grids - a review. *Comp. Math. Math. Phys. 36*, 1 (1996), 1–32. 6

[63] Liseikin, V. D. *Grid Generation Methods*, 2 ed. Scientific Computation. Springer Netherlands, Dordrecht, 2010. 6

[64] Løvholt, F., Pedersen, G., and Gisler, G. Oceanic propagation of a potential tsunami from the La Palma Island. *J. Geophys. Res. 113*, C9 (sep 2008), C09026. 4

[65] Lovholt, F., Pedersen, G., and Glimsdal, S. Coupling of Dispersive Tsunami Propagation and Shallow Water Coastal Response. *The Open Oceanography Journal 4*, 1 (may 2010), 71–82. 4

[66] MacCormack, R. W. The effect of viscosity in hypervelocity impact cratering. *AIAA Paper* (1969), 69–354. 7

[67] Michalopoulos, P. G., Yi, P., and Lyrintzis, A. S. Continuum modelling of traffic dynamics for congested freeways. *Transportation Research Part B: Methodological 27*, 4 (aug 1993), 315–332. 5

[68] Mirchina, N. R., and Pelinovsky, E. N. Nonlinear and dispersive effects for tsunami waves in the open ocean. *Int. J. Tsunami Soc. 2*, 4 (1982), 1073–1081. 4

[69] Murty, T. S., Rao, A. D., Nirupama, N., and Nistor, I. Numerical modelling concepts for tsunami warning systems. *Current Science 90*, 8 (2006), 1073–1081. 4

[70] Peregrine, D. H. Long waves on a beach. *J. Fluid Mech. 27* (1967), 815–827. 4

[71] Popinet, S. Gerris: a tree-based adaptive solver for the incompressible Euler equations in complex geometries. *J. Comp. Phys. 190* (2003), 572–600. 5

[72] Prokopov, G. P. About organization of comparison of algorithms and programs for 2D regular difference mesh construction. *Vopr. At. Nauki Tekh., Ser. Mat. Model. Fiz. Processov 3* (1989), 98–108. 6

[73] Roe, P. L. Approximate Riemann solvers, parameter vectors and difference schemes. *J. Comput. Phys. 43* (1981), 357–372. 5, 24

[74] Rozhdestvenskiy, B. L., and Yanenko, N. N. *Systems of quasilinear equations and*

their application to gas dynamics. Nauka, Moscow, 1978. 10, 28

[75] Serezhnikova, T. I., Sidorov, A. F., and Ushakova, O. V. On one method of construction of optimal curvilinear grids and its applications. *Russ. J. Numer. Anal. Math. Modelling 4*, 2 (1989), 137–156. 6

[76] Shokin, Y. I., Babailov, V. V., Beisel, S. A., Chubarov, L. B., Eletsky, S. V., Fedotova, Z. I., and Gusiakov, V. K. Mathematical Modeling in Application to Regional Tsunami Warning Systems Operations. In *Computational Science and High Performance Computing III.* 2008, pp. 52–68. 4

[77] Shokin, Y. I., Sergeeva, Y. V., and Khakimzyanov, G. S. Construction of monotonic schemes by the differential approximation method. *Russ. J. Numer. Anal. Math. Modelling 20*, 5 (2005), 463–481. 9, 11

[78] Shokin, Y. I., and Urusov, A. I. On the construction of adaptive algorithms for unsteady problems of gas dynamics in arbitrary coordinate systems. In *Eighth International Conference on Numerical Methods in Fluid Dynamics.* Springer, Berlin, Heidelberg, 1982, pp. 481–486. 6, 13, 14

[79] Shokin, Y. I., and Yanenko, N. N. *Method of Differential Approximation. Application to Gas Dynamics.* Nauka, Novosibirsk, 1985. 6

[80] Stockie, J. M., Mackenzie, J. A., and Russell, R. D. A Moving Mesh Method for One-dimensional Hyperbolic Conservation Laws. *SIAM J. Sci. Comput. 22*, 5 (jan 2001), 1791–1813. 12, 21

[81] Stoker, J. J. *Water Waves: The Mathematical Theory with Applications.* John Wiley & Sons, Inc., Hoboken, NJ, USA, jan 1992. 20

[82] Sudobicher, V. G., and Shugrin, S. M. Flow along a dry channel. *Izv. Akad. Nauk SSSR 13*, 3 (1968), 116–122. 6

[83] Synolakis, C. E., and Bernard, E. N. Tsunami science before and beyond Boxing Day 2004. *Phil. Trans. R. Soc. A 364* (2006), 2231–2265. 4

[84] Tang, H., and Tang, T. Adaptive Mesh Methods for One- and Two-Dimensional Hyperbolic Conservation Laws. *SIAM J. Numer. Anal. 41*, 2 (jan 2003), 487–515. 6, 7, 14

[85] Thomas, P. D., and Lombart, C. K. Geometric conservation law and its application to flow computations on moving grid. *AIAA Journal 17*, 10 (1979), 1030–1037. 11, 26

[86] Titov, V. V., and González, F. I. Implementation and testing of the method of splitting tsunami (MOST) model. Tech. Rep. ERL PMEL-112, Pacific Marine Environmental Laboratory, NOAA, 1997. 4

[87] Titov, V. V., and Synolakis, C. E. Numerical modeling of 3-D long wave runup using VTCS-3. In *Long wave runup models*, H. Yeh, P. L.-F. Liu, and C. E. Synolakis, Eds. World Scientific, Singapore, 1996, pp. 242–248. 4

[88] Toro, E. F. *Riemann Solvers and Numerical Methods for Fluid Dynamics.* Springer, Berlin, Heidelberg, 2009. 28

[89] van Dam, A., and Zegeling, P. A. A robust moving mesh finite volume method applied to 1D hyperbolic conservation laws from magnetohydrodynamics. *J. Comp. Phys. 216*, 2 (aug 2006), 526–546. 21

[90] Voltsinger, N. E., Pelinovsky, E. N., and Klevannyi, K. A. *Long wave dynamics of coastal regions.* Gidrometeoizdat, Leningrad, 1989. 28

[91] Wang, X., and Power, W. L. COMCOT: a tsunami generation, propagation and run-up model. Tech. rep., GNS Science, Lower Hutt, NZ, 2011. 4

[92] Whitham, G. B. *Linear and nonlinear waves.* John Wiley & Sons Inc., New York, 1999. 5,

28
[93] Zegeling, P. A., and Blom, J. G. An evaluation of the gradient-weighted moving-finite-element method in one space dimension. *J. Comp. Phys. 103*, 2 (dec 1992), 422–441. 14
[94] Zegeling, P. A., Verwer, J. G., and Van Eijkeren, J. C. H. Application of a moving grid method to a class of 1D brine transport problems in porous media. *Int. J. Num. Meth. Fluids 15*, 2 (jul 1992), 175–191. 14

A. Illustrative examples

Let us consider the 'unsteady' version of the equidistribution principle expressed as the following nonlinear elliptic Boundary Value Problem (BVP):

$$\left(w(x,t)\,x_q\right)_q = 0, \qquad q \in]0,1[, \qquad 0 \leqslant t \leqslant T_f, \tag{A.1}$$

with the following boundary conditions:

$$x(0,t) = 0, \qquad x(1,t) = \ell \qquad 0 \leqslant t \leqslant T_f. \tag{A.2}$$

The first inherent problem with Equation (A.1) is that small perturbations of the monitor function $w(x,t)$ may lead to finite changes in the solution $x(q,t)$ as it is illustrated in the following example.

A.1. Example 1

Let for $t = t_n$ the monitor function w be constant, *i.e.* $w(x,t_n) \equiv 1$. Then, solution to BVP (A.1), (A.2) takes the following simple form:

$$x(q,t_n) = q\ell, \qquad q \in [0,1]. \tag{A.3}$$

This solution prescribes the uniform distribution of nodes on $[0,\ell]$. Let us also assume that at the next time step $t = t_{n+1}$ the monitor function w was perturbed with quantity ε on a portion of the segment $[0,\ell]$:

$$w(x,t_{n+1}) = \begin{cases} 1+\varepsilon, & x \leqslant x_*, \\ 1, & x > x_*, \end{cases} \qquad x_* \in]0,\ell[.$$

Then, solution to BVP (A.1), (A.2) will be given by the following piece-wise linear function:

$$x(q,t_{n+1}) = \begin{cases} q\,\dfrac{x_*}{q_*}, & q \leqslant q_*, \\ x_* + (q-q_*)\,\dfrac{\ell - x_*}{1-q_*}, & q > q_*, \end{cases} \tag{A.4}$$

where $q_* = \dfrac{(1+\varepsilon)x_*}{\ell + \varepsilon x_*}$. From here, the following estimation follows:

$$\|x(\cdot, t_{n+1}) - x(\cdot, t_n)\|_{C(Q)} = |x(q_*, t_{n+1}) - x(q_*, t_n)| = \frac{\varepsilon(\ell - x_*)x_*}{\ell + \varepsilon x_*}.$$

It is not difficult to check that this norm of the difference takes the maximal value for $x_*^{\max} = \dfrac{\ell(\sqrt{1+\varepsilon} - 1)}{\varepsilon}$. For this given $x_*^{\max}$ and for small values of ε we have:

$$\|x(\cdot, t_{n+1}) - x(\cdot, t_n)\|_{C(Q)} = \frac{\ell(2 + \varepsilon - 2\sqrt{1+\varepsilon})}{\varepsilon} = \frac{\varepsilon\ell}{4} + \mathcal{O}(\varepsilon^2).$$

Hence, by choosing the domain length ℓ sufficiently large, we can obtain a finite* perturbation in the solution to BVP (A.1), (A.2).

In numerical practice, it means that the grid nodes in the vicinity of the point $x = x_*^{\max}$ will undergo a shift proportional to $\propto \varepsilon\ell$ when we advance the solution from the time layer $t = t_n$ to $t = t_{n+1}$. If, for some reason, on the time layer $t = t_{n+1}$ the monitor function w becomes constant† again, the grid nodes will experience an opposite move towards a uniform grid (A.3) again. Such oscillations have been observed numerically and the present example provides a theoretical explanation of this phenomenon.

A.2. Example 2

In order to complete the description of possible shortcomings of the equidistribution principle for the mesh motion in evolutionary PDEs, in this Section we consider a simple unsteady problem where the scenario described in Example 1 is actually realized.

Consider the linear advection equation:

$$u_t + a u_x = f(x, t), \qquad a = \text{const} > 0,$$

where the source term is chosen to be

$$f(x, t) = \begin{cases} \varepsilon(x_* - x - at), & x \leqslant x_*, \\ 0 & x > x_*. \end{cases}$$

In this Section ε is a small parameter and $x_* \in]0, \ell[$. An exact solution to the homogeneous Initial-Boundary Value Problem (IBVP) in domain $[0, \ell] \setminus \{x_*\}$ is given by the following formula:

$$u(x, t) = \begin{cases} \varepsilon t(x_* - x), & x < x_*, \\ 0, & x > x_*. \end{cases}$$

Let us choose the monitor function as

$$w(x, t) = 1 + \alpha |u_x(x, t)|$$

*To obtain truly finite perturbations in $x(q, t)$, it is sufficient to take $\ell = \mathcal{O}(\varepsilon^{-1})$.

†In this case, the monitor function may assume two possible constant values $w(x, t) \equiv 1$ or $w(x, t) \equiv 1 + \varepsilon$.

and take the parameter $\alpha \equiv 1$ for simplicity. Then, it is not difficult to check that

$$w(x,t) = \begin{cases} 1 + \varepsilon t, & x < x_*, \\ 1, & x > x_*. \end{cases}$$

The non-uniform grid given by the equidistribution principle is obtained as a solution to BVP (A.1), (A.2). In this example, it takes precisely the form (A.4), but the point q_* depends on time:

$$q_* = \frac{(1 + \varepsilon t)\, x_*}{\ell + \varepsilon t x_*}.$$

It is not difficult to show that the maximal speed of grid nodes $-x_t$ is achieved at $t = 0$ and $x_* = \frac{\ell}{2}$. Moreover, $\max\{-x_t\} = \frac{\varepsilon \ell}{4}$. The grid at $t = 0$ is uniform and at the first time step it will experience a brutal shift to the left. The maximal displacement being equal to $\dfrac{\varepsilon \ell}{4}$. This shift may provoke, at least, a loss of accuracy or, in the worst case scenario, blow-up of the numerical solution. The latter may routinely happen in some explicit schemes, where the time step does not take into account the grid nodes displacements.

Henceforth, we have just demonstrated analytically that the grid constructed with equidistribution principle (A.1) turns out to be unstable when the equidistribution principle is applied in unsteady problems. Indeed, arbitrary small perturbations in the monitor function or in the governing equation may lead to finite perturbations in the grid. Thus, we highly recommend to use the unsteady parabolic Equation (2.15) in order to move grid nodes smoothly.

B. Abbreviations

1D: One-dimensional

2D: Two-dimensional

BVP: Boundary Value Problem

CFL: Courant–Friedrichs–Lewy

DART: Deep-ocean Assessment and Reporting Tsunamis

FEM: Finite Elements Method

FNWD: Fully Nonlinear Weakly Dispersive

IBVP: Initial-Boundary Value Problem

MOST: Method Of Splitting Tsunami

NSWE: Nonlinear Shallow Water Equations

PDE: Partial Differential Equation

TVD: Total Variation Diminishing

WNWD: Weakly Nonlinear Weakly Dispersive

G. Khakimzyanov: Institute of Computational Technologies of Siberian Branch of the Russian Academy of Sciences, Novosibirsk 630090, Russia
E-mail address: Khak@ict.nsc.ru
URL: https://www.researchgate.net/profile/Gayaz_Khakimzyanov3/

D. Dutykh: Univ. Grenoble Alpes, Univ. Savoie Mont Blanc, CNRS, LAMA, 73000 Chambéry, France and LAMA, UMR 5127 CNRS, Université Savoie Mont Blanc, Campus Scientifique, 73376 Le Bourget-du-Lac Cedex, France
E-mail address: Denys.Dutykh@univ-smb.fr
URL: http://www.denys-dutykh.com/

D. Mitsotakis: Victoria University of Wellington, School of Mathematics, Statistics and Operations Research, PO Box 600, Wellington 6140, New Zealand
E-mail address: dmitsot@gmail.com
URL: http://dmitsot.googlepages.com/

N. Yu. Shokina: Department of Radiology, Medical Physics, Medical Center – University of Freiburg, Faculty of Medicine, University of Freiburg, Freiburg, Germany
E-mail address: Nina.Shokina@gmail.com
URL: https://www.researchgate.net/profile/Nina_Shokina/