Journal of Computational Physics 394 (2019) 385–416

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

An efficient hyperbolic relaxation system for dispersive non-hydrostatic water waves and its solution with high order discontinuous Galerkin schemes

C. Escalante [a], M. Dumbser [b,*], M.J. Castro [a]

[a] Departamento de Análisis Matemático, Estadística e Investigación Operativa, y Matemática Aplicada, Universidad de Málaga, Spain
[b] Laboratory of Applied Mathematics, Department of Civil, Environmental and Mechanical Engineering, University of Trento, Italy

ARTICLE INFO

Article history:
Received 25 July 2018
Received in revised form 23 April 2019
Accepted 23 May 2019
Available online 30 May 2019

Keywords:
Non-hydrostatic shallow water flows
Hyperbolic reformulation
ADER discontinuous Galerkin schemes
Path-conservative finite volume methods
Breaking waves
Efficient parallel implementation on GPU

ABSTRACT

In this paper we propose a novel set of first-order hyperbolic equations that can model dispersive non-hydrostatic free surface flows. The governing PDE system is obtained via a hyperbolic approximation of the family of non-hydrostatic free-surface flow models recently derived by Sainte-Marie et al. in [1]. Our new hyperbolic reformulation is based on an augmented system in which the divergence constraint of the velocity is coupled with the other conservation laws via an evolution equation for the depth-averaged non-hydrostatic pressure, similar to the hyperbolic divergence cleaning applied in generalized Lagrangian multiplier methods (GLM) for magnetohydrodynamics (MHD). We suggest a formulation in which the divergence errors of the velocity field are transported with a large but finite wave speed that is directly related to the maximal eigenvalue of the governing PDE.

We then use arbitrary high order accurate (ADER) discontinuous Galerkin (DG) finite element schemes with an *a posteriori* subcell finite volume limiter in order to solve the proposed PDE system numerically. The final scheme is highly accurate in smooth regions of the flow and very robust and positive preserving for emerging topographies and wet-dry fronts. It is well-balanced making use of a path-conservative formulation of HLL-type Riemann solvers based on the straight line segment path. Furthermore, the proposed ADER-DG scheme with *a posteriori* subcell finite volume limiter adapts very well to modern GPU architectures, resulting in a very accurate, robust and computationally efficient computational method for non-hydrostatic free surface flows. The new model proposed in this paper has been applied to idealized academic benchmarks such as the propagation of solitary waves, as well as to more challenging physical situations that involve wave runup on a shore including wave breaking in both one and two space dimensions. In all cases the achieved agreement with analytical solutions or experimental data is very good, thus showing the validity of both, the proposed mathematical model and the numerical solution algorithm.

* Corresponding author.
E-mail addresses: escalante@uma.es (C. Escalante), michael.dumbser@unitn.it (M. Dumbser), castro@anamat.cie.uma.es (M.J. Castro).

https://doi.org/10.1016/j.jcp.2019.05.035

1. Introduction

When modelling and simulating geophysical flows, the non-linear shallow-water equations, hereinafter denoted by SWE, are often a good choice as an approximation of the free surface Navier-Stokes equations. Nevertheless, the SWE do not take into account effects associated with dispersive waves and non-hydrostatic pressure. In particular, it is well known that the simple shallow water equations cannot explain the physics of solitary waves, nor do they reproduce the correct dispersion characteristics of non-hydrostatic surface waves. In recent years, a great effort has been devoted to the derivation of relatively simple mathematical models for shallow water flows that include also long non-linear water waves. As computational power increases, Boussinesq-type models ([2], [3], [4], [5], [6], [7], [8], [9], [10]) become more accessible. This means that one can use more sophisticated models in order to describe reality more accurately, despite the higher computational cost. For efficient semi-implicit methods that can be used to simulate fully three-dimensional non-hydrostatic free surface flows, we refer to the well–known work of Casulli et al. [11–15], including a recent extension [16] that is even able to deal with free surface profiles that are not a single-valued function.

One may use different approaches to improve the non-linear dispersive properties of the simplified depth–averaged shallow water models. Possible options to achieve this are: to consider a Taylor expansion of the velocity potential in powers of the vertical coordinate and in terms of the depth-averaged velocity [5] or the particle velocity components (u, w) at a chosen level [6]; to include two scalars representing the vertical profile of the non-hydrostatic pressure [17]; to use a better flow resolution in the vertical direction with a multi-layer approach [18–20]; to include non-hydrostatic effects in the depth-averaging process ([1,21,22]). The mathematical and numerical study of such dispersive models represents a difficult problem and usually the inversion of an elliptic operator is needed at each time step when the model is numerically solved [23–25,22,26–29]. As a consequence, the computational effort increases drastically. A natural idea is thus to replace the dispersive equations by approximate hyperbolic equations. The idea itself is not new and comes from the pioneering work by Cattaneo [30], who replaced, in particular, the heat equation by a hyperbolic system of equations with relaxation. A very recent development in this direction concerning dispersive water waves has been made in [31], where a hyperbolic reformulation of the Serre-Green-Naghdi equations for flat bottom has been rigorously derived from a variational principle and discretized at the aid of a finite volume scheme. Other related work on hyperbolic models for dispersive water waves can be found, for example, in [32,33]. For a very recent hyperbolic reformulation of the unsteady compressible Navier-Stokes and the resistive MHD equations, the reader is referred to [34–36].

In this paper, we propose a new first-order hyperbolic depth-averaged system that can be seen as a modification of the family of systems presented in [1]. The systems proposed in [1] consist in an approximation of the incompressible Euler equations with free surface. This non-hydrostatic system is capable of solving many relevant features of coastal water waves, such as dispersion, non-linearity, shoaling, refraction, diffraction, run-up and breaking waves (see [23]). Our novel hyperbolic system is obtained using a hyperbolic reformulation of the original governing PDE [1] by coupling the divergence constraint of the velocity with the remaining conservation laws at the aid of an evolution equation for the depth-integrated non-hydrostatic pressure, similar to the method of artificial compressibility for the solution of the incompressible Navier-Stokes equations and also similar to the so-called hyperbolic divergence cleaning introduced in the generalized Lagrangian multiplier approach (GLM) of Munz et al. [37,38] for the Maxwell and the magnetohydrodynamics (MHD) equations. We suggest a formulation in which the divergence errors of the velocity field are transported with a finite speed that is related to the maximum eigenvalues of the governing PDE system. The augmented hyperbolic system maintains the momentum equations for the horizontal and vertical velocities and still satisfies an energy balance equation, as the original system [1].

The final governing PDE system proposed in this paper is a system of hyperbolic balance laws, and is thus amenable for discretization via high order numerical schemes. Higher order methods are desirable due to their improved dissipation and dispersion properties compared to simple second order TVD finite volume schemes. This is particularly important for the accurate propagation of solitary waves over long distances, as it will be also shown later in the numerical results section. In this paper, we choose to discretize the PDE system at the aid of an arbitrary high order accurate discontinuous Galerkin (DG) scheme. The DG finite element method goes back to work of Reed and Hill [39], but it has become particularly popular for the solution of hyperbolic conservation laws thanks to a well-known series of papers by Cockburn and Shu and coworkers, see [40–43]. In subsequent work, Cockburn and Shu [44] extended the DG framework also to the solution of convection-diffusion equations at the aid of the so-called local discontinuous Galerkin (LDG) method. The key idea was to rewrite a PDE with higher order spatial derivatives under the form of an augmented (but not hyperbolic) first order system of PDEs and then formally apply the original DG method to this enlarged first order system. Alternative high order DG methods for PDE with second order diffusion terms can be found, for example, in [45–48]. Yan et al. [49–51] were the first to develop an LDG method for general dispersive Korteweg-de-Vries (KdV) equations including up to third order spatial derivatives. To discretize these higher order spatial derivatives with the LDG approach, the authors again rewrite the governing PDE as an enlarged (non-hyperbolic) first order system, where special care about the choice of the numerical fluxes has been taken in order to get a provably stable scheme. Later, Eskilsson and Sherwin [52,53] and Engsig-Karup et al. [54] were the first to propose higher order DG schemes for the solution of Boussinesq-type equations for the numerical simulation of nonlinear dispersive water waves. For high order WENO schemes applied in the context of dispersive water waves with moving free boundaries, the reader is referred to [55]. Several DG schemes have been proposed by now in order to discretize dispersive PDE systems, but most of them are explicit in time and thus have to obey a very severe stability

condition of the type $\Delta t \sim \Delta x^3$, where Δt is the size of the time step and Δx is the mesh spacing. Therefore, in [56] a new local space-time DG scheme has been introduced for the solution of Boussinesq-type equations, which is provably unconditionally stable for any time step size and which has also been applied to the simulation of nonlinear dispersive water waves. The key idea there was the combination of the LDG method with the space-time DG approach originally proposed for the simulation of compressible fluid flows by Van der Vegt et al. in [57–59] and further analysed for convection-diffusion problems in the work of Feistauer et al. [60,61].

Since the new mathematical model proposed in this paper is a classical first order system of hyperbolic balance laws, it is instead suitable for standard explicit DG schemes subject to a usual CFL-type stability condition ($\Delta t \sim \Delta x$) and does therefore *not* necessarily require the use of an implicit time stepping algorithm. This also makes an efficient parallel implementation very simple and straightforward and is therefore a key advantage of the model proposed in this paper. In particular, in this work we have chosen the family of ADER-DG schemes, which has been introduced in [62–64] and which goes back to the family of ADER finite volume schemes of Toro and Titarev [65–68]. The ADER methodology is based on the approximate solution of the generalized Riemann problem at element interfaces and naturally leads to fully-discrete one-step schemes of arbitrary order of accuracy in both space and time. Due to the well-known Godunov theorem, any better than first order accurate linear scheme is oscillatory and therefore not suitable for the discretization of problems with discontinuities or strong gradients in the solution. Following the ideas introduced in [69,70] we therefore supplement the high order ADER-DG method with a suitable *a posteriori* subcell finite volume limiter. The main idea here is to use first an unlimited high order ADER-DG scheme, which produces a so-called candidate solution at the end of each time step. This candidate solution is then checked *a posteriori* against some physical and numerical detection criteria, such as positivity of the solution, absence of floating point errors and satisfaction of a relaxed discrete maximum principle (DMP). If a cell does not satisfy all these conditions (a so-called troubled cell), the discrete solution is *discarded* and locally *recomputed*, starting again from a valid solution at the old time level, but using now a more robust scheme on a finer subgrid within the troubled cells. This approach corresponds to an element-local checkpointing and restarting of the solver, but using a more robust and more dissipative scheme after the restart. For the recomputation of the troubled cells, in principle, any robust finite volume scheme can be used. Here, we employ the family of path-conservative finite volume schemes, which has already been successfully used for the solution of shallow-water type systems in a series of papers, see e.g. [23,71–76]. Here, we use a robust path-conservative HLL-type Riemann solver [77,78], together with a second order TVD reconstruction. The subcell finite-volume limiter used here is the natural extension of the numerical scheme presented in [23], but applied to the new augmented hyperbolic system proposed in this work. This new concept of *a posteriori* limiting has been introduced for the first time in the context of finite volume schemes via the MOOD approach, see [79–82].

The rest of the paper is organized as follows. In Section 2 we present the original model equations of Sainte-Marie et al. [1] and the proposed novel hyperbolic reformulation. The eigenstructure, energy balance equation and the linear phase celerity of the new system are discussed and compared with those of the original model. The proposed system results in a new and simple hyperbolic system which phase speed depends on the wave number and depth, and thus it can simulate dispersive water waves. In Section 3 we briefly present the ingredients of our ADER-DG schemes with *a posteriori* subcell finite volume limiter used to discretize the governing PDE system. Since the proposed system is hyperbolic, explicit numerical schemes can be considered under a suitable CFL stability condition. This is a real highlight when compared with most other numerical schemes proposed in the literature to discretize hyperbolic-elliptic problems for the simulation of nonlinear dispersive water waves. Due to the combination of an arbitrary high order DG scheme with a robust TVD finite volume scheme, the proposed discretization can deal with a great variety of complex situations involving wet-dry fronts and emerging topographies. In Section 4 a simple wave breaking mechanism is presented. At this point we stress that detailed small-scale wave breaking flow physics is not described by the model, but only the net effect of wave breaking on energy dissipation. This means that we include the breaking mechanism in the depth-integrated equations via a simple sub-scale viscosity model, with a breaking criterion similar to the one proposed in [83]. In Appendix A some guidelines for the implementation of the proposed numerical scheme on GPU architectures are given. The numerical scheme is highly parallelizable, and a strategy for an efficient implementation of the *a posteriori* subcell finite volume limiter is given. Finally, in Section 5, a set of numerical tests including comparisons with analytical solutions for the original system derived in [1] and laboratory data are shown. It can be seen that the numerical scheme is robust, of arbitrary high order and that its implementation on GPU is efficient. The paper is rounded–off by some concluding remarks given in Section 6.

2. Governing equations

We consider the following family of non-hydrostatic pressure systems, depending on a parameter $\gamma \in \mathbb{R}$:

$$\begin{cases} \partial_t h + \partial_x (hu) = 0, \\ \partial_t (hu) + \partial_x \left(hu^2 + \frac{1}{2} g h^2 + hp \right) = (gh + \gamma p)\, \partial_x H - \tau_b, \\ \partial_t (hw) + \partial_x (uhw) = \gamma p, \\ \partial_x u + \dfrac{w + u \partial_x H}{h/2} = 0, \end{cases} \tag{1}$$

where $h = h(x,t)$ is the water depth, $H = H(x)$ is the known still water depth and the surface elevation measured from the still-water level is denoted by $\eta = h - H$. Furthermore, t denotes time, g is the gravitational acceleration and u and w are the depth averaged velocities for each layer in the x and z direction respectively. The non-hydrostatic pressure evaluated at the centre of the water column $z = (\eta + H)/2$ is denoted by p. The bottom friction is included in the model via a usual Manning-type friction formula for the bottom shear stress τ_b that reads

$$\tau_b = n_m^2\, g\, hu\, \frac{|u|}{h^{4/3}},$$

where $n_m \in \mathbb{R}^+$ is an empirical bottom friction coefficient (see [84]).

Within the general formulation (1), $\gamma = 2$ recovers the system first derived by Sainte-Marie et al. [1] and later obtained in [21] as a particular case of a family of non-hydrostatic multilayer systems. For $\gamma = 3/2$, and for the case of a flat bottom, we obtain the Green-Naghdi equations [4] written as a non-hydrostatic pressure system. Notwithstanding, the Green-Naghdi equations can be approximated in the case of stationary bottom and under the hypothesis of mild bottom variation ($\partial_{xx} H(x)$ and $(\partial_x H)^2$ are neglected) by the system (1) with $\gamma = 3/2$.

The system (1) has been numerically discretized in [1,23] in the case of $\gamma = 2$. The governing PDE system is obtained by a process of depth-averaging with respect to the vertical of the incompressible Euler equations. The total pressure is decomposed into a sum of hydrostatic and non-hydrostatic pressure. In the deduction of the equations carried out in [1] for $\gamma = 2$, it is assumed that the horizontal velocity has a constant vertical profile, while the vertical velocity and the associated non-hydrostatic pressure distribution are linear.

It can be easily shown that the model (1) satisfies an extra energy balance law that reads

$$\partial_t E_\gamma + \partial_x \left(u \left(E_\gamma + \frac{g}{2} h^2 + hp \right) \right) = (\gamma - 2)\, up \partial_x H - u \tau_b, \tag{2}$$

where

$$E_\gamma = \frac{h}{2} \left(u^2 + \frac{1}{\gamma} w^2 + g\,(\eta - H) \right). \tag{3}$$

Remark 1. Note that in the case of a flat bottom and in the absence of bottom friction, the system admits an extra conservation law for all values of γ. In the general case of an arbitrary bottom slope, only the case of $\gamma = 2$ ensures the conservation of the energy.

In [21], the Green-Naghdi system [4] is written as a non-hydrostatic pressure system with no assumption of mild bottom slope. In this case, it is shown that the system can be derived after a standard depth-averaging on the vertical of the Euler equations, and assuming a quadratic vertical profile for the non-hydrostatic pressure. This leads to a more complicated system than the one considered here (1) and admits an energy conservation law independently of the bottom slope.

In this work we propose the modified system

$$\begin{cases} \partial_t h + \partial_x (hu) = 0, & \text{(a)} \\ \partial_t (hu) + \partial_x \left(hu^2 + \frac{1}{2} g h^2 + hp \right) = (gh + \gamma p)\, \partial_x H - \tau_b, & \text{(b)} \\ \partial_t (hw) + \partial_x (uhw) = \gamma p, & \text{(c)} \\ \partial_t (hp) + \partial_x (uhp) + hc^2 \left(\partial_x u + \dfrac{w + u \partial_x H}{h/2} \right) = 0, & \text{(d)} \end{cases} \tag{4}$$

where $c = \alpha \sqrt{g H_0}$ is a given constant celerity, H_0 being a typical average still water depth and $\alpha > 1$. The approximation is based on a modified system in which the divergence constraint on the velocity field is coupled with the other conservation laws via an evolution equation for the nonhydrostatic pressure. This approach is based on the underlying ideas

of the method of artificial compressibility for the numerical solution of the incompressible Navier-Stokes equations, see [85–87] and is also directly related to the so-called hyperbolic divergence cleaning applied in the context of the generalized Lagrangian multiplier (GLM) method for the Maxwell and the magnetohydrodynamics (MHD) equations put forward in [37,38]. We suggest a formulation in which the divergence errors are transported with a finite speed c. In the subsequent section we study the hyperbolicity of the augmented system and give a proof that also the augmented system satisfies an additional energy conservation law. For hyperbolic systems with convex extensions, we refer also to the pioneering work of Godunov and Romenski, [88–91], who derived a theoretical framework on symmetric hyperbolic and thermodynamically compatible (SHTC) systems that are all endowed with such an extra conservation law.

2.1. Energy balance of the modified equations

In this subsection it is shown that the proposed hyperbolic relaxation system (4) is also endowed with an extra energy balance law, as the original PDE system (1). For the sake of clarity, it is shown for the case of $\gamma = 2$. In such a case the system (4) satisfies:

$$\partial_t E_\gamma + \partial_x \left(u \left(E + \frac{g}{2} h^2 + hp \right) \right) = -u \tau_b, \tag{5}$$

where

$$E = \frac{h}{2} \left(u^2 + w^2 + \frac{p^2}{c^2} + g\,(\eta - H) \right). \tag{6}$$

Note that for $\gamma = 2$ and in the absence of bottom friction, the equation becomes an extra energy conservation law.

Proof. As it is usually done, by adding ($u \cdot$ (4b) + $w \cdot$ (4c)), and using the mass conservation equation (4a), one has

$$\partial_t \widetilde{E} + \partial_x \left(u \left(\widetilde{E} + \frac{g}{2} h^2 + hp \right) \right) = hp \left(\partial_x u + \frac{w + u \partial_x H}{h/2} \right) - u \tau_b,$$

where

$$\widetilde{E} = \frac{h}{2} \left(u^2 + w^2 + g\,(\eta - H) \right).$$

For the original system (1), one has $\left(\partial_x u + \frac{w + u \partial_x H}{h/2} \right) = 0$, and thus the energy equality (2). For the proposed model, it can be easily checked by using the mass conservation equation (4a), that the following equality holds:

$$\partial_t \left(hp^2 \right) + \partial_x \left(hup^2 \right) + 2c^2 hp \left(\partial_x u + \frac{w + u \partial_x H}{h/2} \right) = 0. \tag{7}$$

Thus,

$$hp \left(\partial_x u + \frac{w + u \partial_x H}{h/2} \right) = -\frac{1}{2c^2} \left(\partial_t \left(hp^2 \right) + \partial_x \left(hup^2 \right) \right)$$

and the relation (5) completes the proof. □

Remark 2. Note that when $c \to \infty$, we recover the original system (1) along with the energy balance (2).

Remark 3. Note that when $c = 0$, and we consider an initial condition $w_0 = p_0 = 0$, then we recover the classical shallow-water system (SWE).

2.2. Eigenstructure of the modified equations

The system (4) can be written in compact matrix-vector form as

$$\partial_t \mathbf{U} + \partial_x \mathbf{F}(\mathbf{U}) + \mathbf{B}(\mathbf{U}) \partial_x \mathbf{U} = \mathbf{S}(\mathbf{U}), \tag{8}$$

with $\mathbf{U} = \left(h, hu, hw, hp \right)$, $\mathbf{F}(\mathbf{U}) = \left(hu, uhu + hp, uhw, hu(p + c^2) \right)$, $\mathbf{S}(\mathbf{U}) = \left(0, -\tau_b, \gamma p, -2c^2 w \right)$, and

$$\mathbf{B}(\mathbf{U}) \partial_x \mathbf{U} = \left(0, (gh + \gamma p) \partial_x \eta - \gamma p \partial_x h, 0, c^2 u \partial_x (h - 2\eta) \right),$$

where equation (4d) has previously been rewritten as

$$\partial_t (hp) + \partial_x (uhp) + c^2 \left(\partial_x (hu) - u \partial_x h + 2w + 2u \partial_x H \right) = 0.$$

Therefore, in quasi-linear form the system reads

$$\partial_t \mathbf{U} + \mathbf{A}(\mathbf{U})\partial_x \mathbf{U} = \mathbf{S}(\mathbf{U}), \tag{9}$$

with

$$\mathbf{A}(\mathbf{U}) = \mathbf{J_F}(\mathbf{U}) + \mathbf{B}(\mathbf{U}),$$

where $\mathbf{J_F} = \partial \mathbf{F}/\partial \mathbf{U}$ is the Jacobian of the flux $\mathbf{F}$ with respect to the conserved variables $\mathbf{U}$. The eigenvalues of the matrix $\mathbf{A}(\mathbf{U})$ are

$$\lambda_{1,2} = u,\ \lambda_{3,4} = u \pm C_e$$

where $C_e = \sqrt{gh + p + c^2}$. A set of linearly independent eigenvectors is given by

$$\begin{aligned} &\mathbf{v}_1 = \left(1,\ u,\ 0,\ -gh\right), \qquad \mathbf{v}_2 = \left(0,\ 0,\ 1,\ 0\right), \\ &\mathbf{v}_{3,4} = \left(1,\ w\left(p + c^2\right)\lambda_{3,4},\ p + c^2,\ 0\right). \end{aligned} \tag{10}$$

2.3. Linear dispersion relation

In this subsection, the linear dispersion relation of the proposed system (4) is studied as usual (see [5,18,29,92]). System (4) is now linearised around the steady state solution $h = H$, $u = 0$, $w = 0$, $p = 0$, and the asymptotic expansion

$$f = f^{(0)} + \epsilon f^{(1)} + \mathcal{O}(\epsilon^2),$$

is considered, where f denotes a generic variable of the system. The resulting linearised model for the perturbations, which is obtained after neglecting $\mathcal{O}(\epsilon^2)$ terms, reads as follows:

$$\begin{cases} \partial_t \eta^{(1)} + H\partial_x u^{(1)} = 0, \\ \partial_t u^{(1)} + g\partial_x \eta^{(1)} + \partial_x p^{(1)} = 0, \\ H\partial_t w^{(1)} - \gamma p^{(1)} = 0, \\ H\partial_t p^{(1)} + Hc^2 \partial_x u^{(1)} + 2c^2 w^{(1)} = 0. \end{cases} \tag{11}$$

We shall now make a Stokes-type Fourier analysis and look for solutions of the form,

$$\eta^{(1)}(x,t) = \eta_0 e^{i(\omega t - kx)},\ u^{(1)}(x,t) = u_0 e^{i(\omega t - kx)},\ w^{(1)}(x,t) = w_0 e^{i(\omega t - kx)},\ p^{(1)}(x,t) = p_0 e^{i(\omega t - kx)}, \tag{12}$$

where ω is the angular frequency and k is the wave number. By substituting (12) into (11), we get the linear system

$$\begin{pmatrix} -\omega & Hk & 0 & 0 \\ gk & -\omega & 0 & k \\ 0 & 0 & H\omega & -\gamma i \\ 0 & -c^2 Hk & 2ic^2 & H\omega \end{pmatrix} \cdot \begin{pmatrix} \eta_0 \\ u_0 \\ w_0 \\ p_0 \end{pmatrix} = \mathbf{0}. \tag{13}$$

Since we assume $c^2 = \alpha^2 gH$, looking for non-trivial solutions, the matrix of the linear system (13) must be singular, yielding the linear dispersion relation

$$(kH)^2 \frac{C_p^2}{gH}\left(\frac{C_p^2}{gH} - 1\right) + 2\gamma\alpha^2\left(1 - \frac{C_p^2}{gH}\left(1 + \frac{(kH)^2}{2\gamma}\right)\right) = 0, \tag{14}$$

where $C_p = \dfrac{\omega}{k}$ is the phase velocity. Note that for $\alpha^2 \to \infty$, we recover the linear dispersion relation of the original system (1):

$$\frac{C_J^2}{gH} = J(kH), \qquad J(kH) = 1 + \frac{(kH)^2}{2\gamma}.$$

From (14), we find $\dfrac{C_p^2}{gH}$:

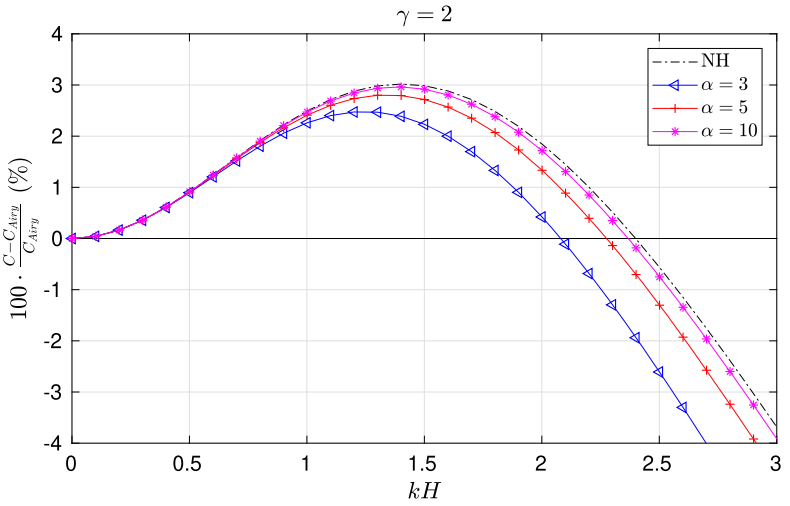

Fig. 1. Relative error of the phase velocities for $\gamma = 2$, with respect to the linear theory of Stokes for the original system (1) (black) and for the new hyperbolic approach (4a)-(4d) using $\alpha = 3$ (blue), $\alpha = 5$ (red) and $\alpha = 10$ (magenta). (For interpretation of the colours in the figure(s), the reader is referred to the web version of this article.)

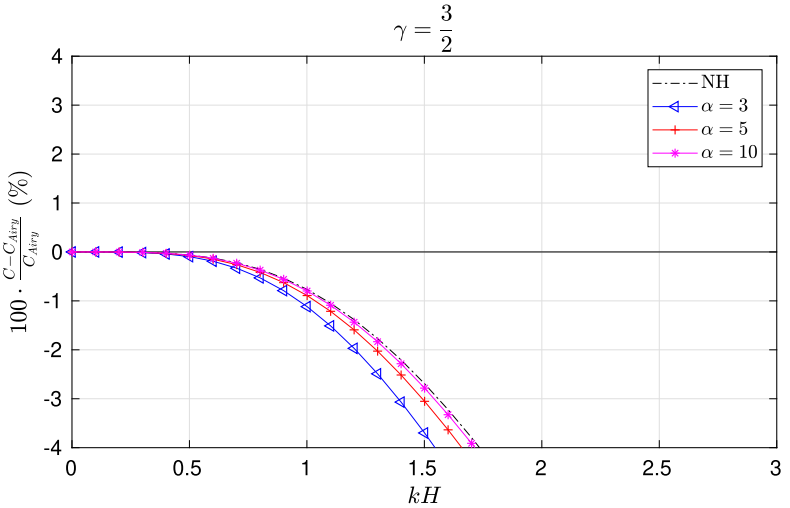

Fig. 2. Relative error of the phase velocities for $\gamma = 3/2$, with respect to the linear theory of Stokes for the original system (1) (black) and for the new hyperbolic approach (4a)-(4d) using $\alpha = 3$ (blue), $\alpha = 5$ (red) and $\alpha = 10$ (magenta).

$$\frac{(C_p^2)^{\pm}}{gH} = \left(\frac{J(kH) + \mu}{2} \mp \sqrt{\left(\frac{J(kH) + \mu}{2} \right)^2 - \mu} \right)^{-1}, \qquad \mu = \frac{1}{2\gamma} \left(\frac{kH}{\alpha} \right)^2. \tag{15}$$

The phase velocities C_p^+ and C_p^- are called the rapid and slow phase velocity, respectively. The velocity C_p^+ is always larger than the one of system (1). It does not have any physical meaning and describes the evolution of artificial high-frequency waves related to the modification of the system.

Figs. 1 and 2 show the error of the phase velocities C_p^- for several values of α, and for the original system (1), with $\gamma = 2$ and $\gamma = 3/2$ respectively, compared to the phase velocity given by the linear theory of Stokes

$$\frac{C_s^2}{gH} = \frac{\tanh(kH)}{kH},$$

in a range of $kH \in [0, 3]$. This interval is chosen according to the range in which the original weakly non-linear weakly dispersive system [1] shows a good match with respect to the linear theory of Stokes. It can be stated that for a value of $c = 5\sqrt{gH}$, the linear dispersion relation of the proposed hyperbolic system is very close to the original one.

As the equations are largely used to simulate shallow water flows, the linear analysis is supplemented with an asymptotic analysis in the limit $kH \to 0$. To do that, we compare the resulting Taylor expansion of the slow phase velocity with the one coming from the Stokes linear theory at order $\mathcal{O}(kH)^4$:

$$\frac{C_p^{-,2}}{gH} = 1 - \frac{1}{2\gamma}(kH)^2 + \mathcal{O}(kH)^4, \quad \frac{C_s^2}{gH} = 1 - \frac{1}{3}(kH)^2 + \mathcal{O}(kH)^4,$$

which coincides up to order $\mathcal{O}(kH)^4$ for $\gamma = 3/2$.

Remark 4. Note that the model for $\gamma = 3/2$, which stands for a hyperbolic reformulation of the Serre-Green-Naghdi system, has a dispersion relation for the phase velocity which is asymptotical of order $\mathcal{O}(kH)^4$, and thus is more appropriate for long-waves as it can be seen in Fig. 2. However, this error is larger for shorter waves when compared with the choice of $\gamma = 2$, that keeps a good error on the phase velocity, approximately bounded by 3 percent, in an extended range of kH up to 3 (see Fig. 1).

2.4. Governing equations in two space dimensions

The corresponding governing equations of the original system (1) for two-dimensional domains can be found in [93]. The equations generalizing the idea applied in (4) for two-dimensional domains read

$$\partial_t \mathbf{U} + \nabla \cdot \mathbf{F}(\mathbf{U}) + \mathbf{B}(\mathbf{U}) \cdot \nabla(\mathbf{U}) = \mathbf{S}(\mathbf{U}), \tag{16}$$

where

$$\mathbf{U} = (h, hu, hv, hw, hp), \quad \mathbf{F}(\mathbf{U}) = \big(\mathbf{f}(\mathbf{U}), \mathbf{g}(\mathbf{U})\big), \quad \mathbf{S}(\mathbf{U}) = \big(0, -\tau_{bx}, -\tau_{by}, \gamma p, -2c^2 w\big),$$

with

$$\mathbf{f}(\mathbf{U}) = \big(hu, uhu + hp, uhv, uhw, hu(p + c^2)\big), \qquad \mathbf{g}(\mathbf{U}) = \big(hv, vhu, vhv + hp, vhw, hv(p + c^2)\big)$$

and the non-conservative product

$$\mathbf{B}(\mathbf{U}) \cdot \nabla(\mathbf{U}) = \mathbf{B}_x(\mathbf{U})\partial_x \mathbf{U} + \mathbf{B}_y(\mathbf{U})\partial_y \mathbf{U}$$

is given by

$$\mathbf{B}_x(\mathbf{U})\partial_x \mathbf{U} = \big(0, (gh + \gamma p)\partial_x \eta - \gamma p \partial_x h, 0, 0, c^2 u \partial_x (h - 2\eta)\big),$$
$$\mathbf{B}_y(\mathbf{U})\partial_y \mathbf{U} = \big(0, 0, (gh + \gamma p)\partial_y \eta - \gamma p \partial_y h, 0, c^2 v \partial_y (h - 2\eta)\big).$$

Here, $H = H(x, y)$ is the known still water depth, u and v denote the depth averaged velocities in the x and y directions, respectively, while w is the depth-averaged velocity component in the z direction and p is the depth-averaged non-hydrostatic pressure.

3. Numerical scheme

In this section the ADER-DG scheme on rectangular grids with *a posteriori* subcell finite volume limiter (SCL) and applied to the hyperbolic system (16) written in compact form

$$\partial_t \mathbf{U} + \nabla \cdot \mathbf{F}(\mathbf{U}) + \mathbf{B}(\mathbf{U}) \cdot \nabla(\mathbf{U}) = \mathbf{S}(\mathbf{U})$$

is briefly recalled. For more details, the reader is referred to [69,70].

3.1. Unlimited ADER-DG scheme and approximate Riemann solvers

The computational domain Ω is covered with a set of non-overlapping Cartesian control volumes in space $\Omega_i = [x_i - \frac{1}{2}\Delta x_i, x_i + \frac{1}{2}\Delta x_i] \times [y_i - \frac{1}{2}\Delta y_i, y_i + \frac{1}{2}\Delta y_i]$, where the vector $\mathbf{x}_i = (x_i, y_i)$ describes the location of the barycentre of cell Ω_i. Furthermore, we denote the vector of the mesh spacings in each direction by $\Delta\mathbf{x}_i = (\Delta x_i, \Delta y_i)$. As usual, the computational domain is the union of all spatial control volumes, hence $\Omega = \bigcup \Omega_i$.

In the following, the discrete solution of the PDE system (8) at time t^n is denoted by $\mathbf{u}_h(\mathbf{x}, t^n)$ and is defined in terms of tensor products of piecewise polynomials of degree N in each spatial direction. In this paper we adopt a nodal basis that is spanned by the Lagrange interpolation polynomials defined on the $(N+1)^d$ Gauss-Legendre quadrature nodes on the element Ω_i, where $d \in \{1, 2\}$ is the number of spatial dimensions. Within the discontinuous Galerkin (DG) finite element framework, the discrete solution $\mathbf{u}_h$ is allowed to *jump* across element interfaces, exactly as in finite volume schemes.

Within each control volume Ω_i the discrete solution is written in terms of the nodal spatial basis functions $\Phi_l(\mathbf{x})$ and some unknown degrees of freedom $\hat{\mathbf{u}}_{i,l}^n$:

$$\mathbf{u}_h(\mathbf{x}, t^n) = \sum_l \hat{\mathbf{u}}_{i,l} \Phi_l(\mathbf{x}) := \hat{\mathbf{u}}_{i,l}^n \Phi_l(\mathbf{x}) \,, \tag{17}$$

where $l = (l_1, l_2)$ is a multi-index. As already mentioned before, the spatial basis functions $\Phi_l(\mathbf{x}) = \varphi_{l_1}(\xi)\varphi_{l_2}(\eta)$ are generated via tensor products of one-dimensional nodal basis functions $\varphi_k(\xi)$ on the reference interval $[0, 1]$, for which we have used the Lagrange interpolation polynomials passing through the Gauss-Legendre quadrature nodes. The transformation from physical coordinates $\mathbf{x} \in \Omega_i$ to reference coordinates $\boldsymbol{\xi} = (\xi, \eta) \in [0, 1]^d$ is given by the linear mapping $\mathbf{x} = \mathbf{x}_i - \frac{1}{2}\Delta\mathbf{x}_i + (\xi\Delta x_i, \eta\Delta y_i)^T$. With this choice, the nodal basis functions satisfy the interpolation property $\varphi_k(\xi_j) = \delta_{kj}$, where δ_{kj} is the usual Kronecker symbol, and the resulting basis is by construction *orthogonal*. Furthermore, due to this particular choice of a *nodal* tensor-product basis, the entire scheme can be written in a dimension-by-dimension fashion, where all integral operators can be decomposed into a sequence of one-dimensional operators acting only on the $N + 1$ degrees of freedom in the respective dimension.

In order to derive the ADER-DG method, we first multiply the governing PDE system (16) with a test function $\Phi_k \in \mathcal{U}_h$ and integrate over the space-time control volume $\Omega_i \times [t^n; t^{n+1}]$. This leads to

$$\int_{t^n}^{t^{n+1}} \int_{\Omega_i} \Phi_k \partial_t \mathbf{U} \, d\mathbf{x}\, dt + \int_{t^n}^{t^{n+1}} \int_{\Omega_i} \Phi_k \left(\nabla \cdot \mathbf{F}(\mathbf{U}) + \mathbf{B}(\mathbf{U}) \cdot \nabla \mathbf{U}\right) d\mathbf{x}\, dt = \int_{t^n}^{t^{n+1}} \int_{\Omega_i} \Phi_k \mathbf{S}(\mathbf{U}) \, d\mathbf{x}\, dt \,, \tag{18}$$

with $d\mathbf{x} = dx\, dy$. As already mentioned before, the discrete solution is allowed to jump across element interfaces, which means that the resulting jump terms have to be taken properly into account. In our scheme this is achieved via numerical flux functions (approximate Riemann solvers) and via the path-conservative approach that was developed by Parés and Castro et al. in the finite volume framework [73,74] and which has later been also extended to the discontinuous Galerkin finite element framework in [94–96]. In classical Runge-Kutta DG schemes [43], only a weak form in space of the PDE is obtained, while time is still kept continuous, thus reducing the problem to a nonlinear system of ODE, which is subsequently integrated with classical Runge-Kutta methods in time. In the ADER-DG framework, a completely different paradigm is used. Here, higher order in time is achieved with the use of an element-local space-time predictor, denoted by $\mathbf{q}_h(\mathbf{x}, t)$ in the following, and which will be discussed in more detail later. Using (17), integrating the first term by parts in time and integrating the flux divergence term by parts in space, taking into account the jumps between elements and making use of this local space-time predictor solution $\mathbf{q}_h$ instead of $\mathbf{U}$, the weak formulation (18) can be rewritten as

$$\left(\int_{\Omega_i} \Phi_k \Phi_l \, d\mathbf{x}\right) \left(\hat{\mathbf{u}}_{i,l}^{n+1} - \hat{\mathbf{u}}_{i,l}^n\right) + \int_{t^n}^{t^{n+1}} \int_{\partial\Omega_i} \Phi_k \mathcal{D}^-\left(\mathbf{q}_h^-, \mathbf{q}_h^+\right) \cdot \mathbf{n} \, dS\, dt - \int_{t^n}^{t^{n+1}} \int_{\Omega_i^\circ} \left(\nabla \Phi_k \cdot \mathbf{F}(\mathbf{q}_h)\right) d\mathbf{x}\, dt +$$
$$+ \int_{t^n}^{t^{n+1}} \int_{\Omega_i^\circ} \Phi_k \left(\mathbf{B}(\mathbf{q}_h) \cdot \nabla \mathbf{q}_h\right) d\mathbf{x}\, dt = \int_{t^n}^{t^{n+1}} \int_{\Omega_i} \Phi_k \mathbf{S}(\mathbf{q}_h) \, d\mathbf{x}\, dt \,, \tag{19}$$

where the first integral leads to the element mass matrix, which is diagonal since our basis is orthogonal. The boundary integral contains the approximate Riemann solver and accounts for the jumps across element interfaces, also in the presence of non-conservative products. The third and fourth integral account for the smooth part of the flux and the non-conservative product, while the right hand side takes into account the presence of the algebraic source term. According to the framework of path-conservative schemes [73,74,95,96], the jump terms are defined via a path-integral in phase space between the boundary extrapolated states at the left $\mathbf{q}_h^-$ and at the right $\mathbf{q}_h^+$ of the interface as follows:

$$\mathcal{D}^-\left(\mathbf{q}_h^-, \mathbf{q}_h^+\right) \cdot \mathbf{n} = \frac{1}{2}\left(\mathbf{F}(\mathbf{q}_h^+) + \mathbf{F}(\mathbf{q}_h^-)\right) \cdot \mathbf{n} + \frac{1}{2}\left(\int_0^1 \mathbf{B}(\boldsymbol{\psi}) \cdot \mathbf{n} \, ds - \boldsymbol{\Theta}\right)\left(\mathbf{q}_h^+ - \mathbf{q}_h^-\right), \tag{20}$$

with $\mathbf{B} \cdot \mathbf{n} = \mathbf{B}_x n_x + \mathbf{B}_y n_y$. In order to achieve exactly well-balanced schemes for certain classes of hyperbolic equations with non-conservative products and source terms, the segment path is not sufficient and a more elaborate choice of the path becomes necessary, see e.g. [97–100]. In this work, we have and choose $\boldsymbol{\psi}$ the simple straight-line segment path for the variables h, η, u and p. In relation (20) above the symbol $\boldsymbol{\Theta} > 0$ denotes an appropriate numerical viscosity matrix. Following [95,96,101], the path integral that appears in (20) can be simply evaluated via some sufficiently accurate numerical quadrature formulae. We typically use a three-point Gauss-Legendre rule in order to approximate the path-integral. For a simple path-conservative Rusanov-type method [76,95], the viscosity matrix reads

$$\mathbf{\Theta}_{\text{Rus}} = s_{\max}\mathbf{I}, \qquad \text{with} \qquad s_{\max} = \max\left(\left|\lambda(\mathbf{q}_h^-)\right|, \left|\lambda(\mathbf{q}_h^+)\right|\right), \tag{21}$$

where $\mathbf{I}$ denotes the identity matrix and $s_{\max}$ is the maximum wave speed (eigenvalue λ of matrix $\mathbf{A}\cdot\mathbf{n}$, $\mathbf{A} = \mathbf{J_F} + \mathbf{B}$) at the element interface. In order to reduce numerical dissipation, one can use better Riemann solvers, such as the Osher-type schemes proposed in [101,102], or the recent extension of the original HLLEM method of Einfeldt and Munz [103] to general conservative and non-conservative hyperbolic systems recently forwarded in [78]. The choice of the approximate Riemann solver and therefore of the viscosity matrix $\mathbf{\Theta}$ completes the numerical scheme (19). In the next subsection, we shortly discuss the computation of the element–local space-time predictor $\mathbf{q}_h$, which is a key ingredient of our high order accurate and communication-avoiding ADER-DG schemes.

3.2. ADER-DG space-time predictor

As already mentioned previously, the element-local space-time predictor is an important *key feature* of ADER-DG schemes and is briefly discussed in this section. The computation of the predictor solution $\mathbf{q}_h(\mathbf{x},t)$ is based on a weak formulation of the governing PDE system in space-time and was first introduced in [63,64]. Starting from the known solution $\mathbf{u}_h(\mathbf{x},t^n)$ at time t^n and following the terminology of Harten et al. [104], we solve a so-called Cauchy problem *in the small*, i.e. without considering the interaction with the neighbour elements. In the ENO scheme of Harten et al. [104] and in the original ADER approach of Toro and Titarev [66–68] the strong differential form of the PDE was used, together with a combination of Taylor series expansions and the so-called Cauchy-Kovalewskaya procedure. The latter is very cumbersome or gets even unfeasible, since it requires a lot of analytic manipulations of the governing PDE system, in order to replace time derivatives with known space derivatives at time t^n. This is achieved by successively differentiating the governing PDE system with respect to space and time and inserting the resulting terms into the Taylor series. Instead, the local space-time discontinuous Galerkin predictor introduced in [63,64], requires only point–wise evaluations of the fluxes, source terms and non-conservative products. For element Ω_i the predictor solution $\mathbf{q}_h$ is now expanded in terms of a local space-time basis

$$\mathbf{q}_h(\mathbf{x},t) = \sum_l \theta_l(\mathbf{x},t)\hat{\mathbf{q}}_l^i := \theta_l(\mathbf{x},t)\hat{\mathbf{q}}_l^i, \tag{22}$$

with the multi-index $l = (l_0, l_1, l_2)$ and where the space-time basis functions $\theta_l(\mathbf{x},t) = \varphi_{l_0}(\tau)\varphi_{l_1}(\xi)\varphi_{l_2}(\eta)$ are again generated from the same one-dimensional nodal basis functions $\varphi_k(\xi)$ as before, i.e. the Lagrange interpolation polynomials of degree N passing through $N+1$ Gauss-Legendre quadrature nodes. The spatial mapping $\mathbf{x} = \mathbf{x}(\boldsymbol{\xi})$ is also the same as before and the physical time is mapped to the reference time $\tau \in [0,1]$ via $t = t^n + \tau\Delta t$. Multiplication of the PDE system (16) with a test function θ_k and integration over the space-time control volume $\Omega_i \times [t^n, t^{n+1}]$ yields the following weak form of the governing PDE, which is *different* from (18), since now the test and basis functions are both time dependent:

$$\int_{t^n}^{t^{n+1}}\int_{\Omega_i} \theta_k(\mathbf{x},t)\partial_t\mathbf{q}_h\,d\mathbf{x}\,dt + \int_{t^n}^{t^{n+1}}\int_{\Omega_i} \theta_k(\mathbf{x},t)\left(\nabla\cdot\mathbf{F}(\mathbf{U}) + \mathbf{B}(\mathbf{q}_h)\cdot\nabla\mathbf{q}_h\right)d\mathbf{x}\,dt = \int_{t^n}^{t^{n+1}}\int_{\Omega_i} \theta_k(\mathbf{x},t)\mathbf{S}(\mathbf{q}_h)\,d\mathbf{x}\,dt\,. \tag{23}$$

Since we are only interested in an element local predictor solution, i.e. without considering interactions with the neighbour elements we do not yet take into account the jumps in $\mathbf{q}_h$ across the element interfaces, since this will be done in the final corrector step of the ADER-DG scheme (19). Instead, we introduce the known discrete solution $u_h(\mathbf{x},t^n)$ at time t^n. For this purpose, the first term is integrated by parts in time. This leads to

$$\begin{aligned}\int_{\Omega_i} \theta_k(\mathbf{x},t^{n+1})\mathbf{q}_h(\mathbf{x},t^{n+1})\,d\mathbf{x} - \int_{t^n}^{t^{n+1}}\int_{\Omega_i} \partial_t\theta_k(\mathbf{x},t)\mathbf{q}_h(\mathbf{x},t)\,d\mathbf{x}\,dt - \int_{\Omega_i} \theta_k(\mathbf{x},t^n)\mathbf{u}_h(\mathbf{x},t^n)\,d\mathbf{x} = \\ - \int_{t^n}^{t^{n+1}}\int_{\Omega_i^\circ} \theta_k(\mathbf{x},t)\nabla\cdot\mathbf{F}(\mathbf{q}_h)\,d\mathbf{x}\,dt + \int_{t^n}^{t^{n+1}}\int_{\Omega_i^\circ} \theta_k(\mathbf{x},t)\left(\mathbf{S}(\mathbf{q}_h) - \mathbf{B}(\mathbf{q}_h)\cdot\nabla\mathbf{q}_h\right)d\mathbf{x}\,dt.\end{aligned} \tag{24}$$

Using the local space-time ansatz (22) Eq. (24) becomes an element-local nonlinear system for the unknown degrees of freedom $\hat{\mathbf{q}}_{i,l}$ of the space-time polynomials $\mathbf{q}_h$. The solution of (24) can be easily found via a simple and fast converging fixed point iteration detailed e.g. in [63,105]. For linear homogeneous systems, the iteration converges in a finite number of at most $N+1$ steps.

We emphasize that the choice of an appropriate *initial guess* $\mathbf{q}_h^0(\mathbf{x},t)$ for $\mathbf{q}_h(\mathbf{x},t)$ is of fundamental importance to obtain a faster convergence and thus a computationally more efficient scheme. For this purpose, one can either use an extrapolation of $\mathbf{q}_h$ from the previous time interval $[t^{n-1}, t^n]$, as suggested e.g. in [106], or one can employ a second-order accurate MUSCL-Hancock-type approach, as forwarded in [105], which is based on discrete derivatives computed at time t^n. As alternative, one can also use a Taylor series expansion of the solution $\mathbf{q}_h(\mathbf{x},t)$ about time t^n and then use a continuous

extension Runge-Kutta scheme (CERK) in order to generate the initial guess for the space-time predictor, as recently pointed out in [107]. For details, see [107] and [108,109]. If an initial guess with polynomial degree $N-1$ in time is chosen, it is sufficient to use *one single* Picard iteration to solve (24) to the desired accuracy, see [64]. For an efficient task-based formalism of ADER-DG schemes, see [110]. This completes the description of the unlimited high order accurate and fully discrete ADER-DG schemes.

3.3. A posteriori subcell finite volume limiter

In regions where the discrete solution is smooth, there is indeed no need for using nonlinear limiters. However, in the presence of shock waves, discontinuities or strong gradients, and taking into account the fact that even a *smooth signal* may become *non-smooth* on the discrete level if it is *underresolved* on the grid, we have to supplement our high order unlimited ADER-DG scheme described above with a nonlinear limiter.

In order to build a simple, robust and accurate limiter, we follow the ideas outlined in [69,70,111,112], where a novel *a posteriori* limiting strategy for ADER-DG schemes was developed, based on the ideas of the MOOD paradigm introduced in [79–82] in the finite volume context. In a first run, the unlimited ADER-DG scheme is used and produces a so-called *candidate solution*, denoted by $\mathbf{u}_h^*(\mathbf{x}, t^{n+1})$ in the following. This candidate solution is then checked *a posteriori* against several physical and numerical detection criteria. For example, we require some relevant physical quantities of the solution to be positive (e.g. pressure and density), we require the absence of floating point errors (`NaN`) and we impose a relaxed discrete maximum principle (DMP) in the sense of polynomials, see [69]. As soon as one of these detection criteria is not satisfied, a cell is marked as troubled zone and is scheduled for limiting.

A cell Ω_i that has been marked for limiting is now split into $(2N+1)^d$ finite volume subcells, which are denoted by $\Omega_{i,s}$ and that satisfy $\Omega_i = \bigcup_s \Omega_{i,s}$. Note that this very fine division of a DG element into finite volume subcells does *not* reduce the time step of the overall ADER-DG scheme, since the CFL number of explicit DG schemes scales with $1/(2N+1)$, while the CFL number of finite volume schemes (used on the subgrid) is of the order of unity. The discrete solution in the subcells $\Omega_{i,s}$ is represented at time t^n in terms of *piecewise constant* subcell averages $\bar{\mathbf{u}}_{i,s}^n$, i.e.

$$\bar{\mathbf{u}}_{i,s}^n = \frac{1}{|\Omega_{i,s}|} \int_{\Omega_{i,s}} \mathbf{U}(\mathbf{x}, t^n)\, d\mathbf{x}. \tag{25}$$

These subcell averages are now evolved in time with a second or third order accurate finite volume scheme, which actually looks very similar to the previous ADER-DG scheme (19), with the difference that now the test function is unity and the spatial control volumes Ω_i are replaced by the sub-volumes $\Omega_{i,s}$:

$$|\Omega_{i,s}| \left(\bar{\mathbf{u}}_{i,s}^{n+1} - \bar{\mathbf{u}}_{i,s}^{n} \right) + \int_{t^n}^{t^{n+1}} \int_{\partial\Omega_{i,s}} \mathcal{D}^- \left(\mathbf{q}_h^-, \mathbf{q}_h^+ \right) \cdot \mathbf{n}\, dS\, dt + \int_{t^n}^{t^{n+1}} \int_{\Omega_{i,s}^\circ} \left(\mathbf{B}(\mathbf{q}_h) \cdot \nabla \mathbf{q}_h \right) d\mathbf{x}\, dt = \int_{t^n}^{t^{n+1}} \int_{\Omega_{i,s}} \mathbf{S}(\mathbf{q}_h)\, d\mathbf{x}\, dt. \tag{26}$$

Here $\mathbf{q}_h$ is now computed with a *well-balanced* second order finite volume HLL method written as a *Polynomial Viscosity Matrix* and positive-preserving path-conservative method (see [72]).

Second order is achieved by using a TVD polynomial reconstruction procedure using the minmod slope limiter (see [113]) which takes into account the positivity of the water height (see [23]). The time evolution can either be achieved via a fully-discrete MUSCL-Hancock-type approach, or via a second order TVD Runge-Kutta method (see [114]).

Once all subcell averages $\bar{\mathbf{u}}_{i,s}^{n+1}$ inside a cell Ω_i have been computed according to (26), the limited DG polynomial $\mathbf{u}_h'(\mathbf{x}, t^{n+1})$ at the next time level is obtained again via a classical constrained least squares reconstruction procedure requiring

$$\frac{1}{|\Omega_{i,s}|} \int_{\Omega_{i,s}} \mathbf{u}_h'(\mathbf{x}, t^{n+1})\, d\mathbf{x} = \bar{\mathbf{u}}_{i,s}^{n+1} \qquad \forall \Omega_{i,s} \in \Omega_i, \qquad \text{and} \qquad \int_{\Omega_i} \mathbf{u}_h'(\mathbf{x}, t^{n+1})\, d\mathbf{x} = \sum_{\Omega_{i,s} \in \Omega_i} |\Omega_{i,s}| \bar{\mathbf{u}}_{i,s}^{n+1}. \tag{27}$$

Here, the second relation is a constraint and means conservation at the level of the control volume Ω_i. In all troubled cells, in addition to the reconstructed degrees of freedom $\hat{\mathbf{u}}_{i,l}^{n+1}$ of the DG polynomial, we also keep in memory the subcell finite volume averages $\bar{\mathbf{u}}_{i,s}^{n+1}$ because they serve as initial condition for the limiter in case that a cell is troubled also in the next time step, see [69]. This completes the brief description of the subcell finite volume limiter used here. For more details, see [69,70,111].

Remark 5. The subcell finite volume limiter considered here is well-balanced for water at rest solutions, linearly L^∞-stable under the usual CFL condition

$$\Delta t < \mathrm{CFL} \frac{\min(\Delta x, \Delta y)}{|\lambda_{max}|}, \qquad 0 < \mathrm{CFL} \leq \frac{1}{d}, \qquad |\lambda_{max}| = \max_i \left\{ |u_i| + \sqrt{g h_i + p_i + c^2} \right\}$$

and positivity preserving for the water height. Note again that at the end of a time step in all troubled cells, both, the reconstructed degrees of freedom of the high order DG scheme as well as the subcell finite volume averages are kept in memory. In case a troubled cell is troubled also in the next time step, the subcell finite volume limiter will directly start from the previous subcell finite volume averages and *not* from the projected DG polynomial, see [69].

Remark 6. For the test cases, c is chosen proportional to a characteristic celerity as $c = \alpha\sqrt{gH_0}$, where H_0 is the characteristic depth of the medium. However, larger values of α result in a more restrictive CFL condition. In practice, numerical tests show that $\alpha = 3$ is a good choice to keep a good balance between accurate numerical results and dispersion relations (see Figs. 1–2) and good computational performance.

3.4. Boundary conditions

For the numerical tests studied in the next section, boundary conditions (BC) are imposed weakly, by enforcing suitable relations at virtual exterior nodes, at each boundary. Periodic boundary conditions (as the computational domains of the cases studied here are rectangular) can be easily enforced by logically linking grid cells at periodic boundaries together.

Nevertheless, to mimic free-outflow boundary conditions, reflections at the boundaries might perturb the numerical solution at the inner domain. As in many other works (see [26,28,29,115] among others), this condition is sometimes supplemented here with an absorbing BC.

Periodic wave generation as well as absorbing BCs are achieved by using a generation/relaxation zone method similar to the one proposed in [115]. Generation/absorption of waves is achieved by simply defining a relaxation coefficient $0 \leq m(x) \leq 1$, and a target solution $\mathbf{U}^*$.

Given a width L_{Rel} of the relaxation zone, the solution within the relaxation zone is then redefined to be

$$\widetilde{\mathbf{U}}_i = m_i \mathbf{U}_i + (1 - m_i)\mathbf{U}_i^*$$

for every i in the relaxation zone. m_i is defined as

$$m_i = \sqrt{1 - \left(\frac{d_i}{L_{Rel}}\right)^2},$$

where d_i is the distance between the centre of the cells I_i the closest boundary. In our numerical experiments we set

$$L \leq L_{Rel} \leq 1.5L,$$

being L the typical wavelength of the outgoing wave. Absorbing BC is the special case $\mathbf{U}_i^* = 0$, that will damp all the waves passing through.

4. Modelling of breaking waves and treatment of wet-dry fronts

As pointed out in [83], in shallow water flow complex events can be observed related to turbulent processes. One of these processes corresponds to the breaking of waves near the coast. As it will be seen in the numerical tests shown in the next section, the governing PDE system (4a) - (4d) cannot describe this process without an additional term that allows the model to dissipate the required amount of energy in such situations. When wave breaking processes occur, mostly close to shallow areas, two different approaches are usually employed when dispersive Boussinesq-type models are considered.

Close to the coast, where wave breaking starts, the simple shallow water equations propagate breaking waves under the form of a travelling shock wave at the correct speed, since kH is small, and energy dissipation of the breaking wave is also well reproduced by the shock. Due to that, the simplest way to deal with breaking waves in dispersive systems consists in simply neglecting the dispersive part of the governing equations, which means to force the non-hydrostatic pressure to be zero where breaking occurs. Due to that, this technique has the advantage that only a breaking criterion is needed to stop and start it. However, the main disadvantage is that the grid-convergence is not ensured when the mesh is refined, and a global and eventually costly breaking criterion should be taken into account, see e.g. [26].

The other strategy, that will be adopted in this work, consists in dissipation of breaking bores with a diffusive term. Again, a breaking criterion to switch on and off this extra dissipation term is needed. Usually, an eddy viscosity approach (see [83]) solves the matter, where an empirical parameter is defined, based on a quasi-heuristic strategy to determine when the breaking occurs. Usually, this extra diffusive term needs to be discretized implicitly due to the higher order spatial derivatives coming from the diffusion. Otherwise, it may lead to a severe restriction on the time step. As a consequence, a linear system has to be solved, losing thus computational efficiency. A new technique to overcome this challenge has been proposed in [23] for the non-hydrostatic pressure system derived in [22]. A similar idea is applied in this work for the case of two-dimensional domains. To do so, let us consider first the original hyperbolic-elliptic non-hydrostatic pressure system (1) and let us consider the vertical component of the stress-tensor:

$$\tau_{zz} = 2\nu\partial_z w,$$

where $\nu(x,z,t)$ is a positive function. We use the same process as in [1], to depth-average the vertical component of the stress-tensor. By taking into account the incompressibility condition from (1) and that the vertical velocity has a linear profile within the fluid layer, as it was supposed in [1]:

$$\int_{-H}^{\eta} \partial_z \tau_{zz}\, dz = 2\varsigma \left(w + u\partial_x H + v\partial_y H \right),$$

where $\varsigma = \int_{-H}^{\eta} \partial_z \nu$ is the eddy viscosity. In this work, as in [23,83] we choose ς to be

$$\varsigma = -Bh|\partial_x(hu) + \partial_y(hv)|,$$

where B is a coefficient related to the breaking criterion described below. Let us denote

$$\mathbf{R_b}\left(\mathbf{U}, \nabla\mathbf{U}, \nabla H\right) = \left(0, 0, 0, 2\varsigma \left(w + u\partial_x H + v\partial_y H \right), 0\right).$$

Adding the proposed integrated viscosity terms to the vertical momentum equation of the system (1), and after the relaxation technique proposed in this work, the system reads

$$\partial_t \mathbf{U} + \nabla \cdot \mathbf{F}(\mathbf{U}) + \mathbf{B}(\mathbf{U}) \cdot \nabla \mathbf{U} = \mathbf{S}(\mathbf{U}) + \mathbf{R_b}\left(\mathbf{U}, \nabla\mathbf{U}, \nabla H\right).$$

Note that the fourth component of $\mathbf{R_b}$ contains up to first order derivatives, and therefore the numerical scheme proposed in this work adapts straightforwardly to this new term. Moreover, for the case of a mild slope bottom, the fourth component of $\mathbf{R_b}$ reduces to $-2B|\partial_x(hu) + \partial_x(hv)|w$.

Finally, a breaking criterion to switch on/off the dissipation is needed. A natural and simple extension of the criterion proposed by [83] to two space dimensions reads

$$B = 1 - \frac{\partial_x(hu) + \partial_y(hv)}{U_1} \qquad \text{for} \qquad |\partial_x(hu) + \partial_x(hv)| \geq U_2.$$

Energy dissipation associated with wave breaking starts when $|\partial_x(hu) + \partial_y(hv)| \geq U_1$ and continues as long as $|\partial_x(hu) + \partial_y(hv)| \geq U_2$, where

$$U_1 = B_1\sqrt{gh}, \qquad U_2 = B_2\sqrt{gh},$$

denote the flow speeds at the onset and termination of the wave-breaking process and B_1, B_2 are calibration coefficients that should be calibrated through laboratory experiments. In this work, as in [23,83], we use $B_1 = 0.5$ and $B_2 = 0.15$ for all the test cases studied.

The breaking mechanism employed in this work can be considered either with the breaking criterion given in [83], or with the one proposed in [26], which is more sophisticated but also computationally more expensive. Nevertheless, although we have chosen a fast and simple breaking criterion, the numerical tests in Section 5 will show that this technique performs adequately. Moreover, the simple breaking mechanism considered in this work corrects the classical overshoot that dispersive models present for the run-up of waves (see Fig. 9); it can deal well with hydraulic jumps (see Fig. 10); it ensures the grid convergence even if the breaking mechanism is dynamically switching on/off during the simulation (see Fig. 12).

Concerning the wet-dry treatment, the unlimited numerical solution obtained with the ADER-DG is checked against the condition

$$h < \epsilon_{DG},\ \epsilon_{DG} = 10^{-3},$$

which means that for those cells with a water height less than a certain ϵ_{DG}, the cell is immediately marked as a troubled cell, and a numerical solution is computed with the subcell finite-volume solver strategy described in Section 3.3 above. A wet-dry treatment, as described in [23,116], in regions with emerging bottom is considered in the finite-volume solver. No special treatment is required for the non-hydrostatic pressure, since in the presence of wet-dry fonts it vanishes automatically.

Note that at wet-dry fronts the bottom friction term τ_b may become stiff. Therefore, this term is discretized explicitly only in the unlimited ADER-DG solver, while τ_b is discretized in a semi-implicit way inside the subcell finite volume limiter, in order to deal with the potential stiffness of the bottom friction source term.

5. Numerical tests

In this section we show some numerical tests for the model (4a) - (4d) presented in Section 2, with the breaking and friction terms introduced previously. Since the numerical scheme presented here exhibits a high potential for data parallelization, a parallel implementation of the numerical scheme has been carried out on GPU architectures. Some guidelines about the implementation are given in Appendix A.

The chosen tests have already been widely studied in the literature on dispersive water waves modelling (see [23–25, 117,118,11,18,26,28,119,120]). Thus, this choice constitutes a proper way to validate the presented dispersive model and the numerical scheme with analytical solutions and real laboratory tests.

The first test aims at validating the mathematical model and the numerical scheme with a comparison against exact analytical solutions for the original dispersive system (1). The other four tests show that the proposed numerical scheme applied to PDE system (4) can properly simulate complex laboratory experiments, showing also comparisons of numerical results with real experimental data.

The quantities of the parameters concerning the following numerical simulations are expressed in units of measure of the International System of Units. The simulations are performed with a third order ADER-DG scheme ($\mathbb{P}_2$), except for the test where a systematic convergence analysis of the scheme is carried out, as well as for the computation of the GPU-performance tables. The limiting strategy presented in Section 3 is employed. The breaking mechanism is used with the parameters $B_1 = 0.15$ and $B_2 = 0.5$. The CFL number is set to 0.9 of the linear stability limit of the scheme. The gravitational acceleration is set to $g = 9.81$ in all tests. The artificial non-hydrostatic pressure wave speed c is set to $c = \alpha\sqrt{gH}$, $\alpha = 3$ for all test cases, except for the first test case where an analytical solution of the original system is studied, and thus $\alpha \in \{5, 10\}$. For some experiments, hydrostatic simulations (SWE) will be computed to see the influence of dispersive/non-hydrostatic effects. To do that, α is set to zero.

5.1. Solitary waves

The propagation of a solitary wave over a long distance is a standard test of the stability and conservative properties of numerical schemes for weakly-nonlinear weakly-dispersive models ([1,22,27,28,83,121]). The analytical solution for the original system (1) consists of a solitary wave of amplitude A that travels at a speed of $c_A = \sqrt{g(A+H)}$. In [1] analytical expressions for the Serre Green-Naghdi system can be found as well, and similar results can be obtained in this test case when $\gamma = 3/2$. For a quasi-exact solution of the hyperbolic system (4), see the next subsection 5.1.1.

5.1.1. Computation of quasi-exact solitary waves for the hyperbolic model

Since the analytical solitary wave solution of the original elliptic problem (1) is *not* the exact solution of the hyperbolic system (4), in this section we construct some quasi-exact solitary wave solutions for the proposed hyperbolic system. Let us consider a solitary wave moving along the x-axis at a constant velocity c_A as in [1]. Therefore, a solitary wave solution of the system (8) depends only on the similarity variable

$$\xi = \frac{x - c_A t}{l}, \quad c_A = \sqrt{g(A+H)}, \quad l = H\sqrt{\frac{A+H}{H}},$$

i.e. we obtain $\mathbf{U} = \mathbf{U}(\xi)$. Hence, the time and space derivative are given by

$$\partial_t \mathbf{U} = -\frac{c}{l}\mathbf{U}', \quad \partial_x \mathbf{U} = \frac{1}{l}\mathbf{U}',$$

where the prime symbol denotes differentiation with respect to ξ. In the case of solitary waves, the proposed PDE system written in quasi-linear form (9) reduces to the nonlinear ODE system

$$\left(A(\mathbf{U}(\xi)) - c_A I\right)\mathbf{U}' = l \cdot S(\mathbf{U}(\xi)). \tag{28}$$

After integrating the mass conservation equation, and imposing the asymptotic conditions

$$h(\xi) \to H, \; u(\xi) \to 0 \text{ when } \xi \to \pm\infty,$$

one can obtain the exact relation

$$hu = c_A(h - H), \tag{29}$$

and similarly, integrating the horizontal momentum equation and imposing the asymptotic condition $p(\xi) \to 0$ when $\xi \to \pm\infty$, the following relation holds:

$$hp = c_a^2(h - H)\frac{H}{h} + \frac{1}{2}g\left(H^2 - h^2\right). \tag{30}$$

Note that the horizontal velocity and the non-hydrostatic pressure have been solely expressed in terms of the water depth. Concerning the remaining equations in the system, they read

$$w' = \frac{-2l}{c_A H}p, \tag{31}$$

$$h' = -\frac{2lc^2 h}{c_A H}\left(c^2 + c_A^2\frac{H}{h} - 2c_A\frac{H^2}{h^2} + \frac{1}{2}g\frac{H^2}{h} + \frac{1}{2}gh\right)^{-1} w. \tag{32}$$

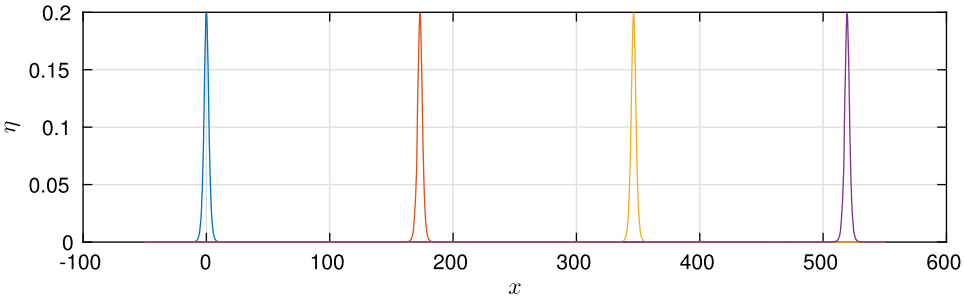

Fig. 3. Solitary wave of amplitude $A = 0.2$ and $\gamma = 2$ at $t = 0,\ 50,\ 100,\ 150$. $\mathbb{P}_2$ scheme.

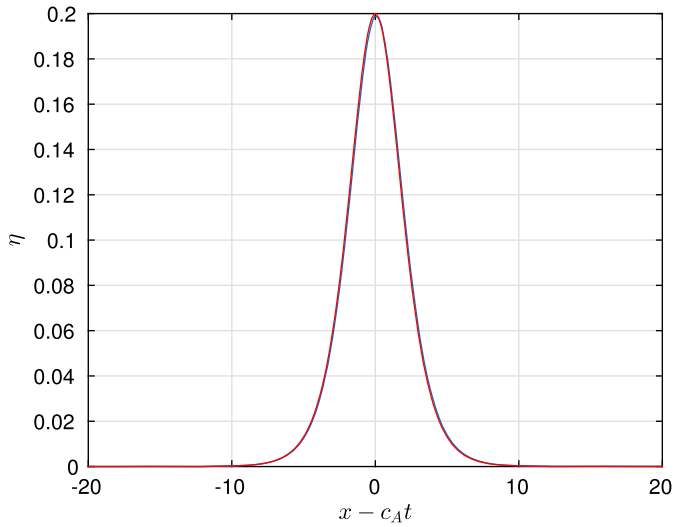

Fig. 4. Comparison of the free surface at time $t = 120$ shifted by $(x - c_{A_i} t)$ m (blue) and initial condition (red) for the solitary wave of amplitude $A = 0.2$ and $\gamma = 2$. $\mathbb{P}_2$ scheme.

To obtain a quasi-exact solution, the nonlinear ODE system (31)–(32) can be solved in a numerical way up to any desired accuracy using a high order ODE solver with a rather small time step size. For this purpose could either use a classical high order Runge-Kutta scheme, or a high order time discontinuous Galerkin scheme as the one employed in [122,105]. Note that once the ODE system has been solved for h and w, the rest of the variables can be obtained from the exact relations previously obtained (29)–(30). In this work a classical fourth-order Runge-Kutta ODE solver has been employed with a small time step $\Delta\xi = 10^{-6}$ for the generation of the quasi-exact solution of the solitary waves of the hyperbolic system (8). In the following, we will consider this highly accurate numerical solution of the ODE system as exact solution of the problem.

5.1.2. Long-time evolution of a large amplitude solitary wave

In our first test we use the quasi-exact solution of the hyperbolic system (4) described in the previous subsection for $\gamma = 2$.

A solitary wave of amplitude $A = 0.2$ propagates over a constant still-water depth $H = 1$ in a channel of length 600 m along the x direction. The domain is divided into 600 cells along the x axis. Periodic boundary conditions are used. The final simulation time is $t = 150$ s. The artificial non-hydrostatic pressure wave speed c is set to $c = \alpha\sqrt{gH},\ \alpha = 5$. Although the *a posteriori* limiter strategy is implemented, it does not detect any troubled elements, hence during the entire simulation of the solitary wave the pure unlimited ADER-DG scheme is used.

Fig. 3 shows the evolution of the solitary wave of amplitude $A = 0.2$ at different times using the ADER-DG $\mathbb{P}_2$ scheme. Figs. 4, 5, 6 and 7 show a perfect match between the numerical and the quasi-exact solution after a large integration time for the free surface, the horizontal and vertical velocities as well as for the non-hydrostatic pressure.

It is worth mentioning that excellent results are obtained with a spatial discretization of only $\Delta x = 1\ m$. In other works (see [23,31]) usually it was necessary to refine the grid substantially in order to maintain the correct amplitude of the solitary wave for large integration times. This especially occurs when flux limiters such as minmod ([23]) are present in the numerical scheme, resulting in an undesired clipping of local extrema. This makes the high order accurate DG schemes an appropriate framework for the propagation of dispersive water waves, and in particular, for solitary waves, see also the numerical results obtained in [54,56].

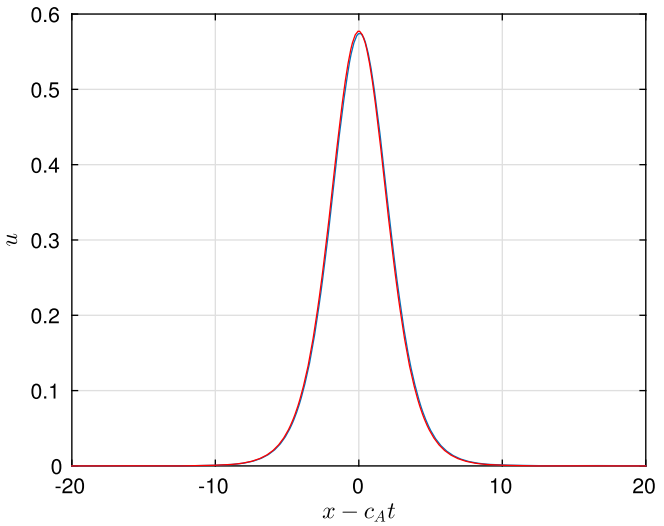

Fig. 5. Comparison of the horizontal velocity u at time $t = 120$ shifted by $(x - c_{A_i} t)$ m (blue) and initial condition (red) for the solitary wave of amplitude $A = 0.2$ and $\gamma = 2$. $\mathbb{P}_2$ scheme.

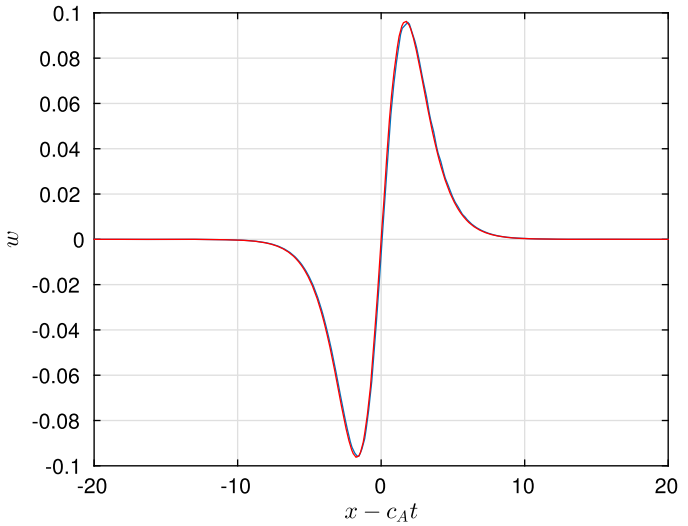

Fig. 6. Comparison of vertical velocity w at time $t = 120$ shifted by $(x - c_{A_i} t)$ m (blue) and initial condition (red) for the solitary wave of amplitude $A = 0.2$ and $\gamma = 2$. $\mathbb{P}_2$ scheme.

5.1.3. Numerical convergence studies

To check the accuracy of the numerical scheme we propose a numerical test consisting in the propagation of a solitary wave of amplitude $A = 0.01$ m over a constant still-water depth $H = 1$ m in a channel of length 200.997 m along the x direction. Periodic boundary conditions are used. The final simulation time is set to $t = 63.855$ s, which is the propagation time needed to make a complete cycle through the domain. The artificial non-hydrostatic pressure wave speed c is set to $c = \alpha\sqrt{gH}$ with $\alpha = 5$. Although the *a posteriori* limiter strategy is active, it doe not detect any troubled elements, hence during the entire simulation of the solitary wave the pure unlimited ADER-DG scheme is used. Table 1 shows the computed L^2 errors for the hydrodynamic variables h and hu, for different meshes and different polynomial approximation degrees of the DG scheme. The numerical test evinces that the theoretical convergence order is properly achieved by our scheme.

5.2. Solitary wave run-up on a plane beach

Synolakis [123] carried out laboratory experiments for incident solitary waves, to study propagation, breaking and run-up over a planar beach with a slope 1 : 19.85. Many researchers have used this data to validate numerical models. With this test case we assess the ability of the model to describe shoreline motions and wave breaking, when it occurs. Experimental data for the surface elevation are available in [123] at different times. The bathymetry of the problem is described in Fig. 8.

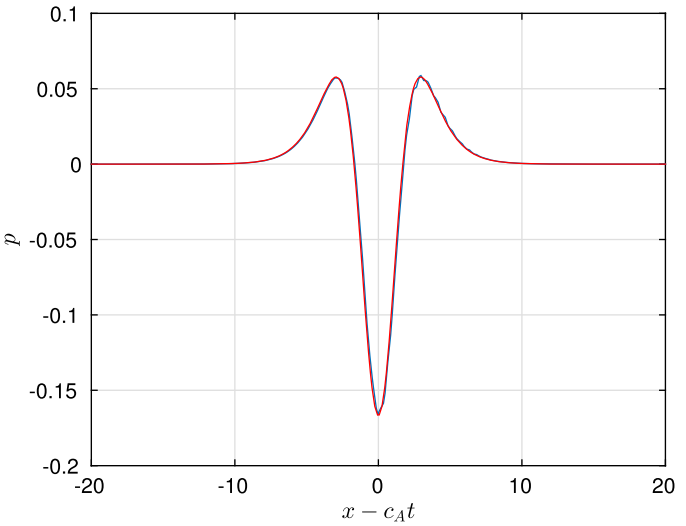

Fig. 7. Comparison of non-hydrostatic pressure p at time $t = 120$ shifted by $(x - c_{A_i} t)$ m (blue) and initial condition (red) for the solitary wave of amplitude $A = 0.2$ and $\gamma = 2$. $\mathbb{P}_2$ scheme.

Table 1
L_2 errors and convergence rates for the Solitary wave problem for the ADER-DG-$\mathbb{P}_N$ scheme. The errors have been computed for the variables h and hu and $\alpha = 5$.

Solitary wave problem — $\gamma = 2$ — ADER-DG-$\mathbb{P}_N$						
	N_x	L_2 error h	L_2 error hu	L_2 order h	L_2 order hu	Theor.
DG-$\mathbb{P}_2$	30	1.12E-3	5.20E-5	–	–	3
	40	4.99E-4	2.38E-5	2.80	2.72	
	50	2.38E-4	1.09E-5	3.31	3.49	
	60	1.27E-4	5.44E-6	3.48	3.82	
DG-$\mathbb{P}_3$	30	1.23E-4	4.94E-6	–	–	4
	40	2.84E-5	9.77E-7	5.09	5.63	
	50	9.63E-6	3.12E-7	4.85	5.12	
	60	4.20E-6	1.35E-7	4.55	4.59	
DG-$\mathbb{P}_4$	15	5.00E-4	2.26E-5	–	–	5
	20	1.02E-4	3.99E-6	5.53	6.02	
	25	2.49E-5	8.25E-7	6.32	7.06	
	30	7.59E-6	2.19E-7	6.50	7.27	
DG-$\mathbb{P}_5$	10	8.70E-4	3.52E-5	–	–	6
	15	8.61E-5	3.18E-6	5.71	5.93	
	20	7.97E-6	2.60E-7	8.27	8.71	
	25	1.83E-6	5.42E-8	6.60	7.02	
DG-$\mathbb{P}_6$	10	2.54E-4	9.85E-6	–	–	7
	15	1.35E-5	3.94E-7	7.24	7.94	
	25	3.91E-7	1.09E-8	6.93	7.02	
	30	1.07E-7	3.06E-9	7.12	6.96	

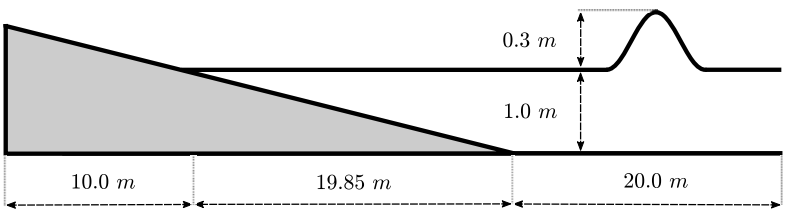

Fig. 8. Sketch of the bathymetry used for the solitary wave run-up onto a beach test problem.

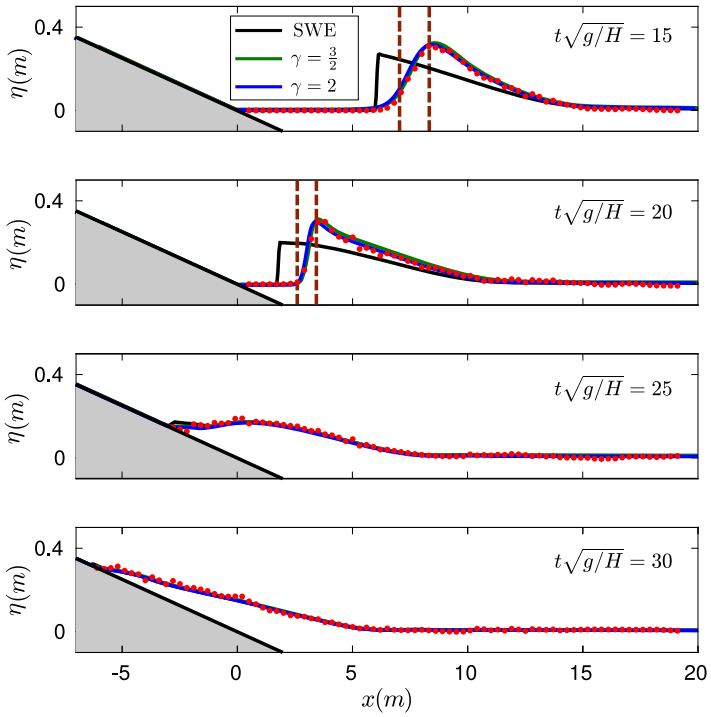

Fig. 9. Comparison of experimental data (red) and numerical simulation with the proposed hyperbolic system (4) including friction and wave breaking mechanism for $\gamma = 2$ (blue), $\gamma = 3/2$ (green) and the standard shallow water equations (black) at different times during the run-up. Between the vertical bars, the region where the breaking mechanism is active is shown.

Two solitary waves of amplitude $A = 0.3$, given by the exact solution of the original system (1) for $\gamma \in \{2, 3/2\}$, are placed at the location $x = 20$. This serves as initial condition for the free-surface elevation and all other flow quantities. A Manning coefficient of $n_m = 0.01$ is used in order to define the glass surface roughness used in the experiments. The computational domain $\Omega = [-10, 40]$ is divided into 500 equidistant cells. Free-outflow boundary conditions are considered.

Fig. 9 and Fig. 10 show snapshots, at different times, $t\sqrt{g/H} = t_0$ where $H = 1$, comparing experimental and simulated data for two different simulations with $\gamma = 2$, $\gamma = 3/2$, and for $\alpha = 0$ which stands for hydrostatic simulations (SWE). Fig. 9 also shows where the breaking mechanism is active (region between the bars), and demonstrates the efficacy of the criterion used in this paper. Numerical results evince only very small discrepancies for both choices of $\gamma = 2$ and $\gamma = 3/2$. As expected, observing the dispersive relation (see Fig. 2) for the celerity, the system with $\gamma = 3/2$ tends to decelerate waves, and thus the slightly later arrival time of the wave. Simulations highlight the importance of dispersive effects when compared with hydrostatic simulations (SWE).

The breaking mechanism also works properly in terms of grid convergence, see Fig. 12, where the snapshots at times $t\sqrt{g/H} = 15$ (run-up) and $t\sqrt{g/H} = 55$ (run-down) are shown for different mesh sizes.

Finally, Fig. 11 represents the obtained numerical results when the breaking mechanism is not considered. In this case, a spurious overshoot of the wave amplitude appears, which underlines the importance to consider wave breaking in the context of dispersive non-linear shallow water models.

In addition, good results are obtained for the maximum wave run-up, where the friction terms play an important role. Note that no additional wet-dry treatment for the non-hydrostatic pressure is needed. This test shows that the proposed *a posteriori* limiting strategy, the chosen breaking mechanism, as well as the standard SWE friction term perform adequately for the proposed hyperbolic system. Moreover, the corresponding discretization is robust and can deal with the presence of wet-dry fronts correctly.

5.3. Periodic waves over a submerged bar

The experiment of plunging breaking periodic waves over a submerged bar by Beji and Battjes [124] is considered here. The numerical test is performed in a one-dimensional channel with a trapezoidal obstacle submerged. Waves in the free surface are measured in seven point stations $S_0, S_1, \ldots, S_6$ (see Fig. 13).

The one-dimensional domain $[0, 25]$ is discretized with $\Delta x = 0.05$ m, and the bathymetry is defined in the Fig. 13.

The velocity u and surface elevation η are set initially to 0. The boundary conditions are: free outflow at $x = 25$ m and free surface is imposed at $x = 0$ m using the data provided by the experiment at S_0. The data provided at S_0 by the

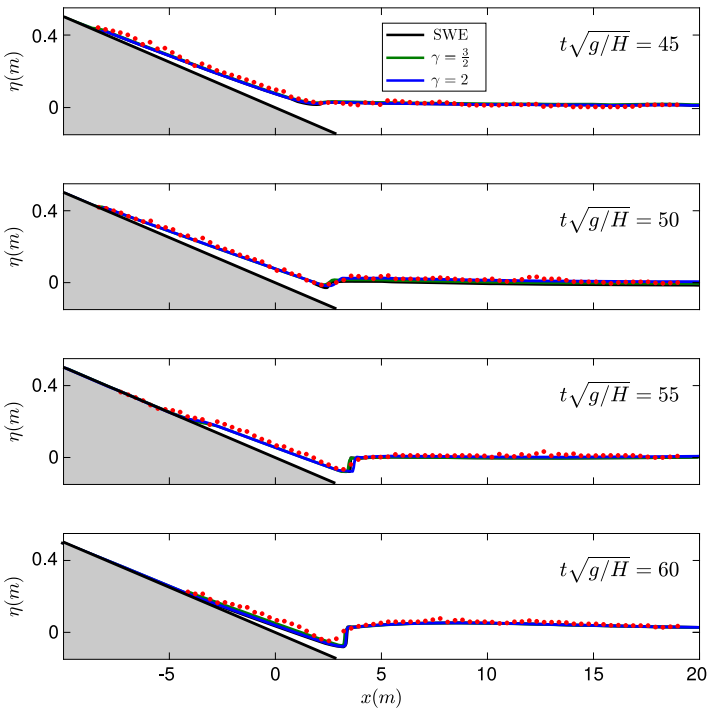

Fig. 10. Comparison of experimental data (red) and numerical simulation with the proposed hyperbolic system (4) for $\gamma = 2$ (blue), $\gamma = 3/2$ (green) and the standard shallow water equations (black) at different times during the wave run-down.

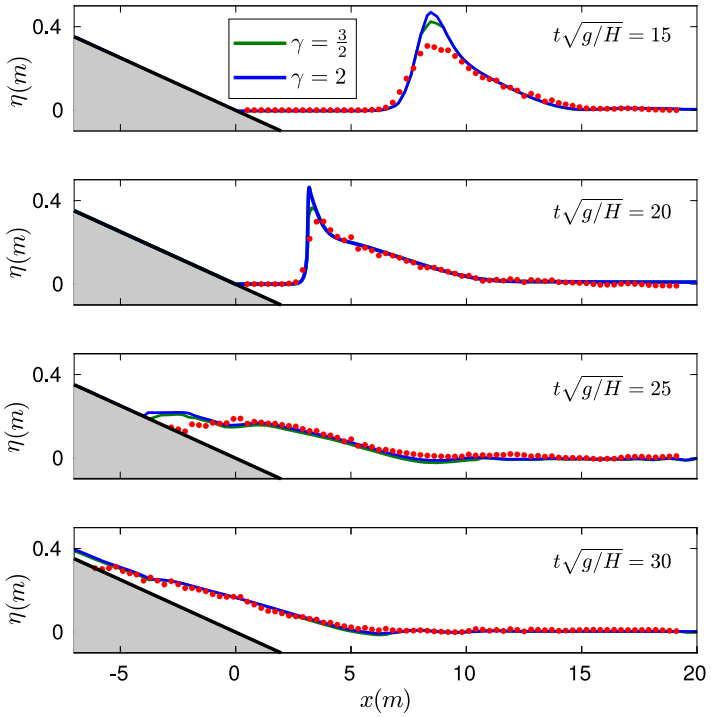

Fig. 11. Comparison of experimental data (red) and numerical simulation with the proposed hyperbolic system (4) for $\gamma = 2$ (blue) and $\gamma = 3/2$ (green) at different times during the run-up, but *without* using a wave breaking mechanism.

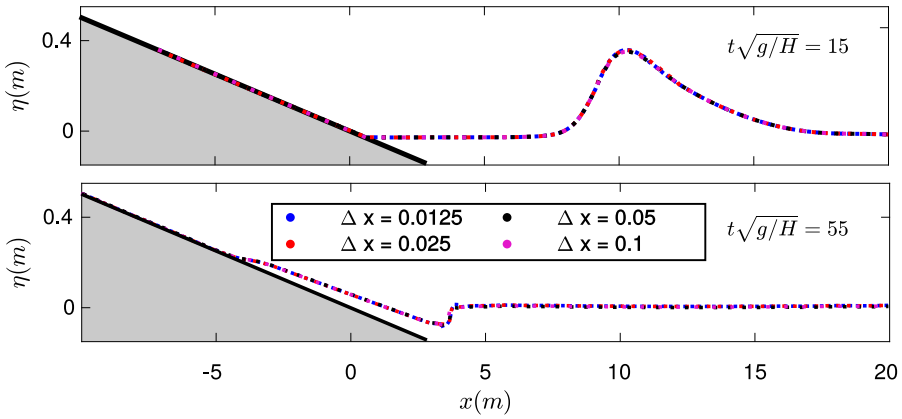

Fig. 12. Comparison of the free-surface location at times $t\sqrt{g/H} = 15$ and $t\sqrt{g/H} = 55$ for different mesh sizes using $\gamma = 2$.

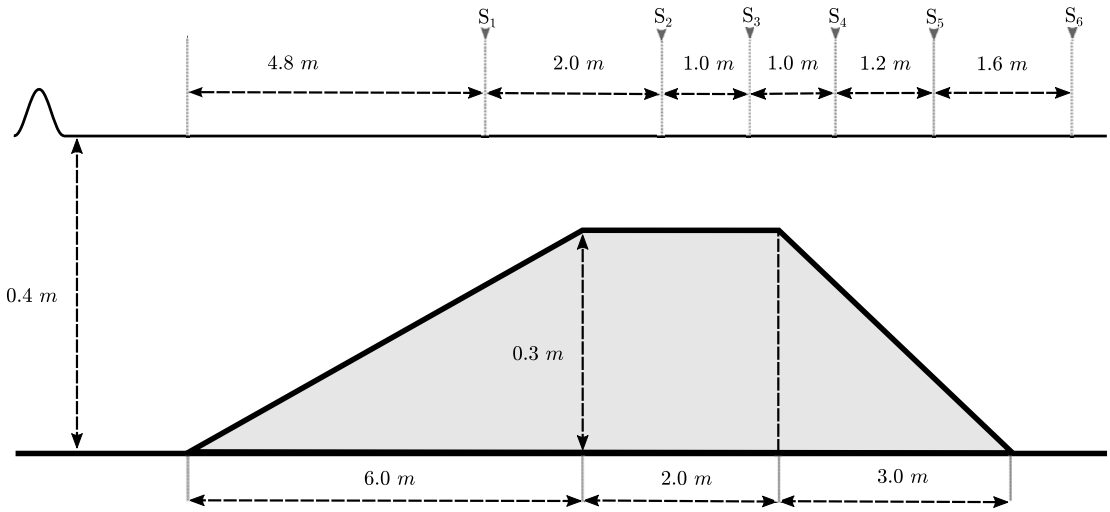

Fig. 13. Periodic waves over a submerged bar. Sketch of the topography and layout of the wave gauges.

experiment is the free-surface $\eta_{S_0}(t)$ and the velocity $u_{S_0}(t)$. Thus, we use as a target solution for the generating boundary condition (see Section 3.4)

$$h^*(t) = 0.4 + \eta_{S_0}(t),\ u^*(t) = (t)u_{S_0}(t),\ w^*(t) = 0,\ p^*(t) = 0.$$

The first wave gauge S_1 shows that the imposed generating boundary conditions are well implemented, since the match is excellent.

Fig. 14 shows the time evolution of the free surface at points $S_1, \ldots, S_6$. The comparison with experimental data emphasizes the need to consider a dispersive model to faithfully capture the shape of the waves near the continental slope, according to the comparison between the hydrostatic (SWE) and non-hydrostatic results. Both amplitude and frequency of the waves are captured on all wave gauges successfully. Numerical results with the parameter $\gamma = 3/2$, that stands for the hyperbolic Serre-Green-Naghdi simulation, show a better fit with data at some wave gauges, where higher harmonics appear (e.g. in wave gauge S_5).

5.4. Favre waves

Here we consider an experiment where a fluid layer with a free surface is impacting against a vertical wall (see e.g. [31, 125,126]). Due to dispersion, the reflected wave is a wave train of waves of different lengths and amplitudes (see Fig. 15), rather than a simple shock wave as predicted by the shallow water equations (SWE).

The one-dimensional computational domain is $\Omega = [0, 180]$ and we set $H = 1$. The impact velocity u_0 is related to the relative Froude number F by the formula ([127])

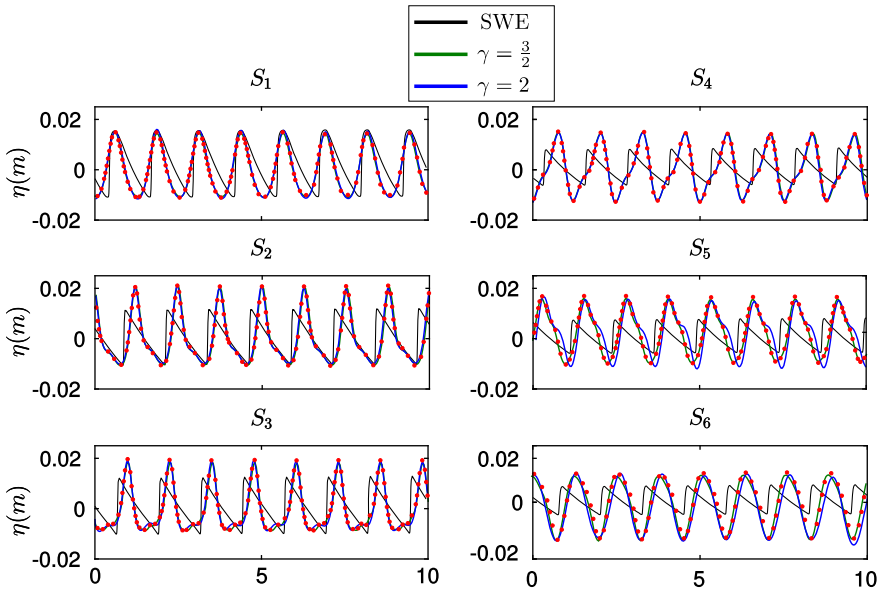

Fig. 14. Comparison of the time series of the experimental data (red) with the numerical solution of the new hyperbolic model (4) with $\gamma = 2$ (blue), $\gamma = 3/2$ (green) and the standard shallow water equations (black) at different wave gauges S_i.

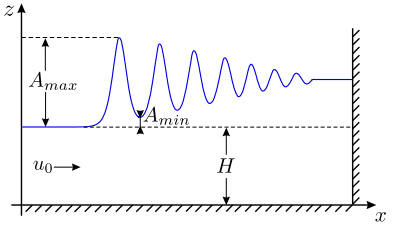

Fig. 15. Sketch of Favre waves.

$$u_0 = \left(F - \frac{1 + \sqrt{1 + 8F^2}}{4F} \right) \sqrt{gH}. \tag{33}$$

Hence, the initial condition for u is given by (33) and $w = p = 0$ as well as $h = H$. The final simulation time was $t = 54$ s.

Fig. 16 shows a comparison at time $t = 54$ for the Froude number $Fr = 1.35$ for different mesh sizes and $\gamma = 2$. One can observe that the results for 250 and 500 elements are quite similar, hence good numerical results can be obtained also on reasonably coarse meshes.

It is well-known that above the critical value $Fr = 1.35$, breaking waves arise (see [31,127]) and the model is not valid without a breaking mechanism. Fig. 17 shows the comparison between the experimental and the numerical results with a breaking mechanism included and without it. With wave breaking the obtained numerical results are in excellent agreement with the measured experimental data for both values of the parameter $\gamma \in \{2, 3/2\}$. Fig. 17 also clearly shows that the breaking mechanism only acts when and where it is really needed, since similar results with respect to the simulations without the breaking mechanism are observed for $F < 1.35$, where no wave breaking occurs in the experiments.

5.5. Plane solitary wave impinging on a conical island

The goal of this 2D numerical test is to compare the results of our new mathematical model for $\gamma = 2$, $\gamma = 3/2$ and $\alpha = 0$ (SWE) with laboratory measurements for a two-dimensional problem. The experiment was carried out at the Coastal and Hydraulic Laboratory, Engineer Research and Development Center of the U.S. Army Corps of Engineers ([128]). The laboratory experiment consists in an idealized representation of Babi Island in the Flores Sea in Indonesia. The produced data sets have been frequently used to validate run-up models ([22,129]).

A directional wave-maker is used to produce planar solitary waves of specified crest lengths and heights. The set–up consists in a 25×30 m^2 basin with a conical island situated near the centre. The still water level is $H = 0.32$ m. The island has a base diameter of 7.2 m, a top diameter of 2.2 m and it is 0.625 m high. Four wave gauges, $\{WG_1, WG_2, WG_3, WG_4\}$, are distributed around the island in order to measure the free surface elevation (see Fig. 18).

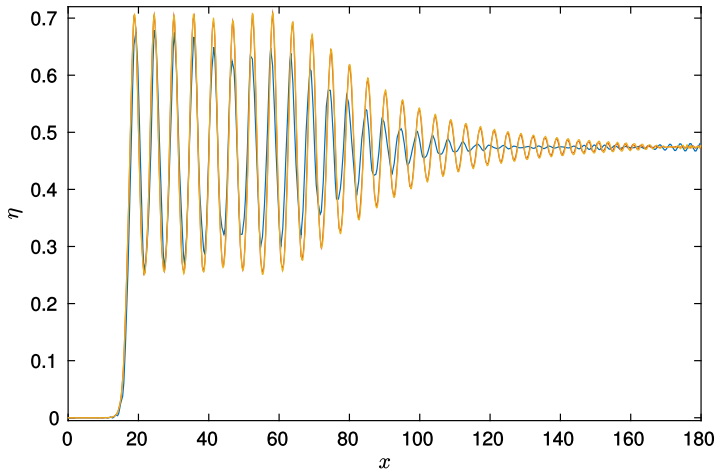

Fig. 16. Comparison of the numerical results with $\gamma = 2$ obtained at time $t = 54$ for the Froude number $Fr = 1.35$. The numerical simulations are shown for different mesh sizes: 125 (blue), 250 (red) and 500 (orange) elements.

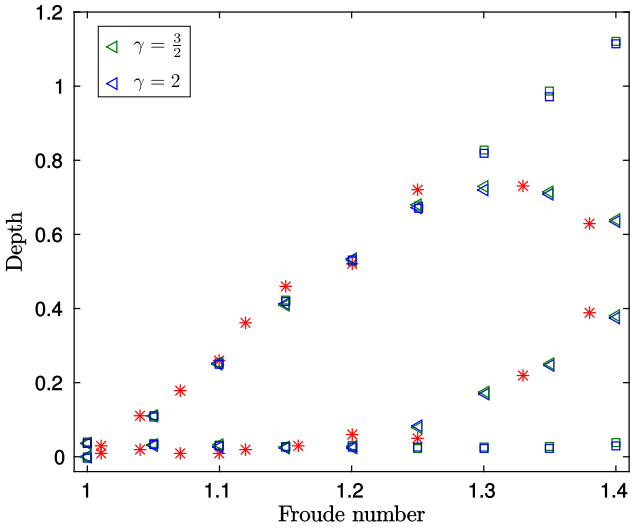

Fig. 17. Comparison between the experimental data (red asterisks) and the numerical results with a breaking mechanism (triangles) and without (squares). Blue colours stand for $\gamma = 2$ and green for $\gamma = 3/2$. The upper markers indicate the amplitude of the first wave; the lower show the amplitude of the trough after the first wave.

For the numerical simulation the computational domain is chosen as $\Omega = [-5, 23] \times [0, 28]$ with $\Delta x = 0.1$ and $\Delta y = 0.1$. Free outflow boundary conditions are imposed. As initial condition for η, u, w and p, a solitary wave of amplitude $A = 0.06$ centred at $x = 0$ is given. The wave propagates until $t = 30$ seconds and a Manning coefficient of $n_m = 0.015$ is used.

The numerical simulation shows two wave fronts splitting in front of the island and collide behind it (see Figs. 23 and 24). Comparison between measured and computed water levels at gauges WG_1, WG_2, WG_3, WG_4 show good results (Fig. 20), as well as a comparison between the computed run-up and the laboratory measurement, see Fig. 19. Comparison between hydrostatic and non-hydrostatic simulations evince the importance of dispersive non-hydrostatic effects on this type of events.

Table 2 shows the execution times on an NVIDIA Tesla P100 GPU for $\alpha = 0$ (SWE) and $\alpha = 3$. For the non-hydrostatic simulations ($\alpha = 3$), the computational results remain approximately equal for $\gamma = 2$ and $\gamma = 3/2$. In view of the obtained results, we can conclude that the non-hydrostatic code can achieve a good computational performance with an additional computational cost that is only at most 2.95 times the cost of a simple SWE simulation. This additional computational cost is similar to the one presented in [23], where the non-hydrostatic pressure system proposed in [22] was discretized with a second order hybrid finite volume–finite difference scheme, by solving a mixed hyperbolic-elliptic problem. The real highlight in this work is that the same low additional computational cost is maintained for any order of accuracy in space and time of the numerical scheme. Moreover, Fig. 21 shows that the computational time required to evolve a degree of freedom remains approximately constant when the order of the DG scheme is increased.

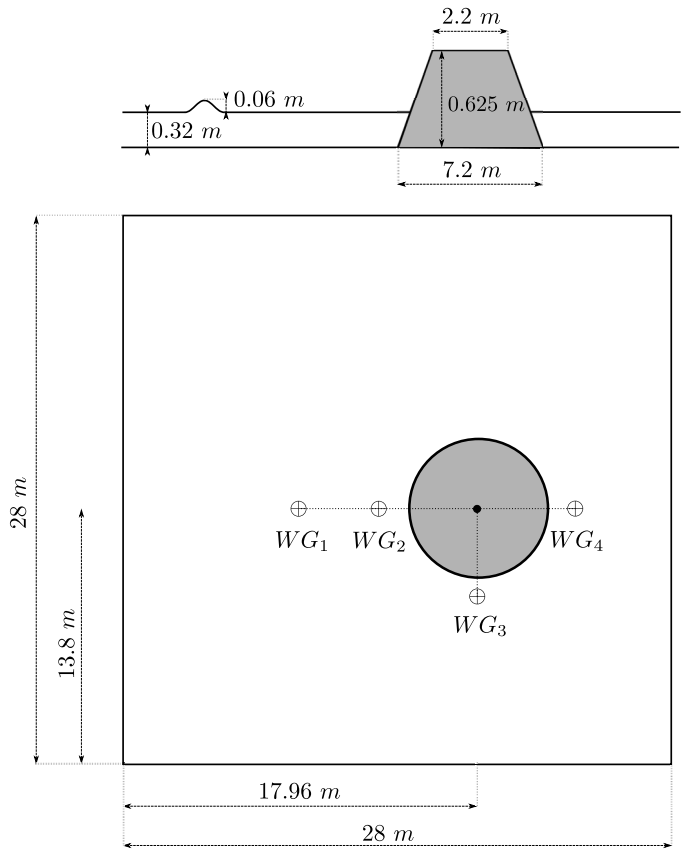

Fig. 18. Sketch of the topography for the conical island test case.

As can be expected, almost all the execution time is spent in the space-time predictor *kernel*, which is *communication-avoiding* and thus by construction ideal for parallel execution. Fig. 22 shows graphically the GFLOPS/s obtained in the CUDA implementation for the space-time predictor *kernel*. The code achieves 700 GFLOPS/s for big enough meshes. Theoretical maximum peak performance for the Tesla P100 is 4.761 TFLOPS in double precision, and therefore the code can achieve around 14% of this theoretical maximum peak performance value. Moreover, the GFLOP/s values obtained are rather independent of the order of the scheme.

It can thus be stated that the numerical scheme used here is computationally efficient and can correctly simulate dispersive water waves with only a moderate computational overhead compared to the classical SWE model.

6. Conclusion

A new first order hyperbolic model for shallow non-hydrostatic free surface flows has been proposed in order to incorporate dispersive effects in the propagation of waves in a homogeneous, inviscid and incompressible fluid. For $\gamma = 2$ the presented model corresponds to a hyperbolic approximation of the dispersive system derived by Sainte-Marie et al. in [1,21], while for $\gamma = 3/2$ a hyperbolic reformulation of the mild-slope approximation of the Serre-Green-Naghdi model is retrieved. The dispersion properties of our new hyperbolic system are close to those of the aforementioned original models. However, the big advantage of our new hyperbolic formulation is that it can be easily discretized with explicit and high order accurate numerical schemes for hyperbolic conservation laws, without requiring the solution of an elliptic problem in each time step, in contrast to the original PDE system proposed in [1,21] (see [23,93]).

The numerical scheme employed for the solution of the new model combines an explicit arbitrary high order accurate fully discrete one-step ADER-DG scheme with an *a posteriori* finite volume subcell limiter that is based on a robust path-conservative HLL Riemann solver. The method is high order accurate in smooth regions and is well-balanced for the water at rest solution; it is by construction positivity preserving and stable under a usual CFL-type stability condition.

The new model is able to simulate properly the run-up and run-down process in wet-dry areas on the shore. Moreover, no numerical truncation for the non-hydrostatic pressure is needed in wet-dry areas, where the non-hydrostatic pressure vanishes.

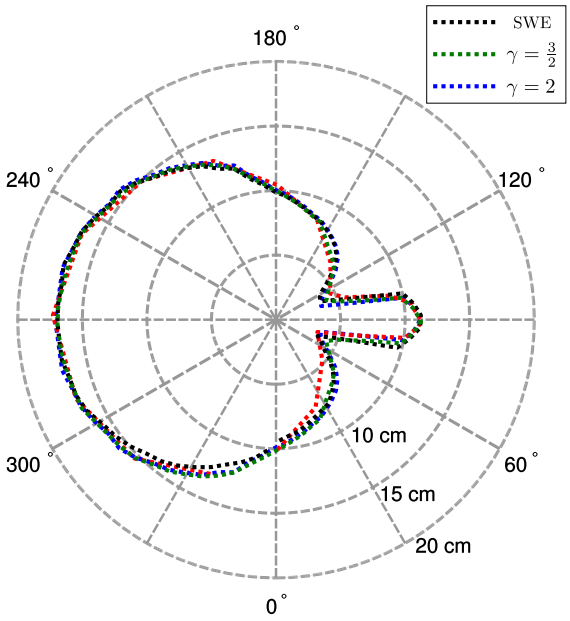

Fig. 19. Maximum run-up measured (red) and simulated with $\gamma = 2$ (blue) and $\gamma = 3/2$ (green) on a 200×200 cell mesh with the ADER-DG $\mathbb{P}_2$ scheme applied to the new hyperbolic model (4) proposed in this paper. For comparison, also the wave runup as computed by the standard shallow water equations (black) is shown.

Table 2
Execution times in seconds of the GPU implementation of the scheme for the new non-hydrostatic model (4) with $\alpha = 3$ and the standard shallow water equations (SWE).

Runtime(s) GPU — ADER-DG-$\mathbb{P}_N$				
	$N_x \times N_y$	NH-model (4), $\alpha = 3$	Shallow water	Ratio
DG-$\mathbb{P}_1$	100×100	7.29	4.85	1.50
	200×200	47.09	29.02	1.62
	300×300	147.88	70.04	2.11
DG-$\mathbb{P}_2$	100×100	89.98	40.23	2.24
	200×200	705.90	296.10	2.38
	300×300	2323.79	932.51	2.49
DG-$\mathbb{P}_3$	100×100	548.56	230.83	2.38
	200×200	4446.10	1803.18	2.47
	300×300	14496.71	5625.33	2.58
DG-$\mathbb{P}_4$	100×100	4823.15	2285.85	2.11
	200×200	19292.60	7874.53	2.45
	300×300	43311.88	15692.71	2.76
DG-$\mathbb{P}_5$	100×100	12078.85	4755.45	2.54
	200×200	48315.39	16776.18	2.88
	300×300	93600.83	36850.72	2.54
DG-$\mathbb{P}_6$	100×100	29410.50	9836.29	2.99
	200×200	117642.01	48814.11	2.41
	300×300	364882.63	123689.03	2.95
DG-$\mathbb{P}_7$	100×100	49100.24	18118.17	2.71
	200×200	196400.97	88468.91	2.22
	300×300	441902.18	206496.34	2.14

As it is well known, non-hydrostatic models need some additional dissipative mechanism for describing the effect of breaking waves, in order to accurately model their behaviours in coastal areas (see [23]). We have implemented a very simple and efficient breaking mechanism that performs properly in all numerical tests that have been carried out.

In order to allow simulations in real time or faster, an efficient GPU implementation of the numerical method has been carried out in the two-dimensional case. From a computational point of view, the non-hydrostatic code presents good computational times with respect to the standard shallow water equations discretized with the same scheme on the same GPU system. On fine meshes and with a third order accurate scheme, the wall-clock times needed for non-hydrostatic

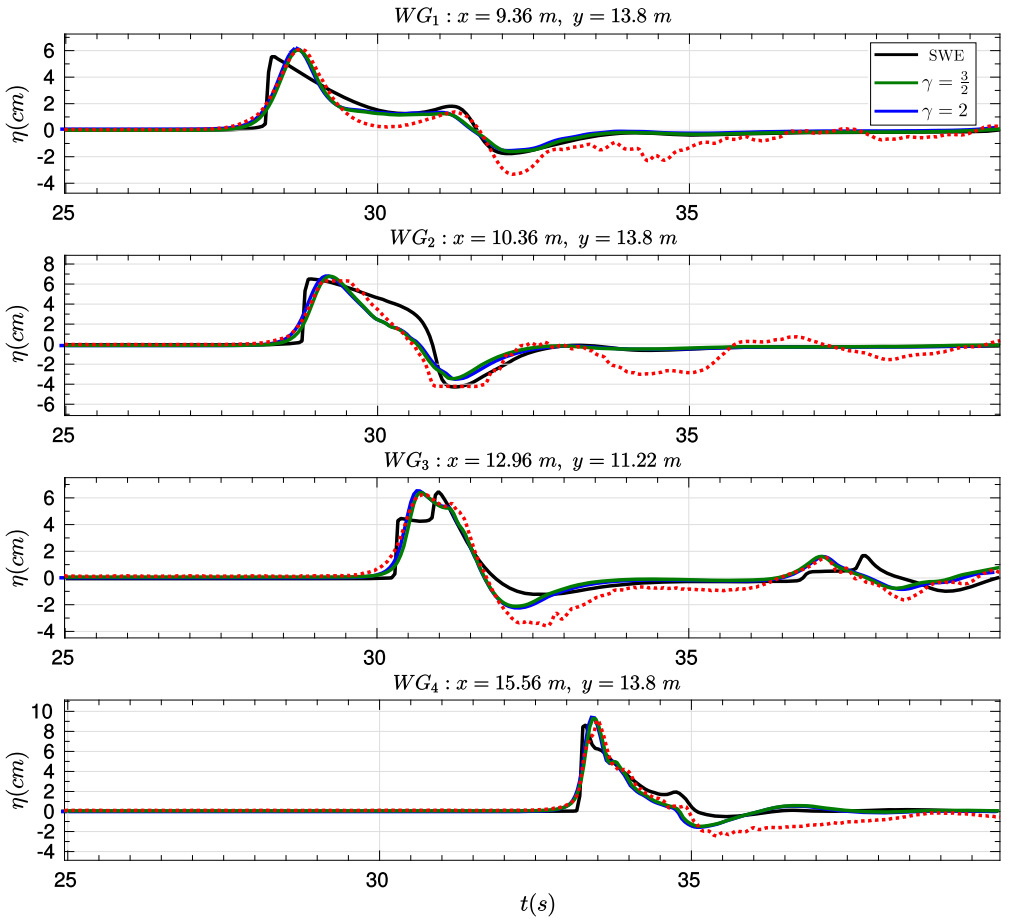

Fig. 20. Comparison of experimental data time series (red) and numerical results for $\gamma = 2$ (blue), $\gamma = 3/2$ (green) and standard SWE (black) at wave gauges WG_1, WG_2, WG_3, WG_4 on a mesh composed of 200×200 elements using the ADER-DG $\mathbb{P}_2$ scheme applied to the new hyperbolic model (4) proposed in this paper.

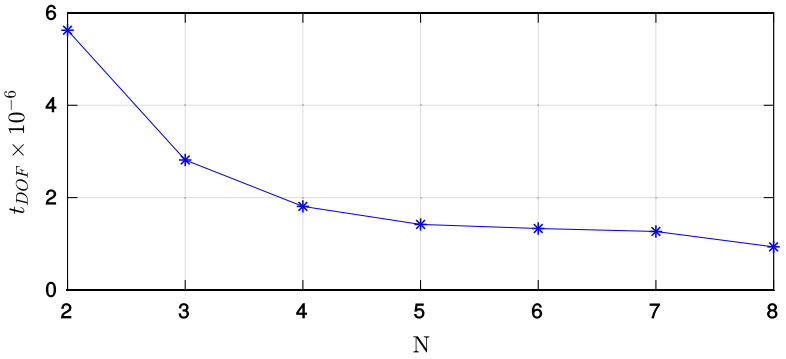

Fig. 21. Computational time ($\times 10^{-6}$) to evolve a degree of freedom for a DG-$\mathbb{P}_N$ scheme.

simulations with the new model proposed in this paper are at most a factor of 2.95 higher than the wall clock times needed for a simple shallow water model, but which is not able to capture the correct dispersion characteristics of non-hydrostatic water waves. A similar factor of the computational overhead for nonhydrostatic simulations w.r.t. SWE is obtained in [23] for a second order accurate finite volume scheme. The great asset of the high order numerical scheme proposed in this paper is that this good computational factor compared to SWE is maintained for any order of the numerical scheme. In the same manner, the number of measured floating point operations (FLOPS) show a similar feature for any order of accuracy of the numerical scheme.

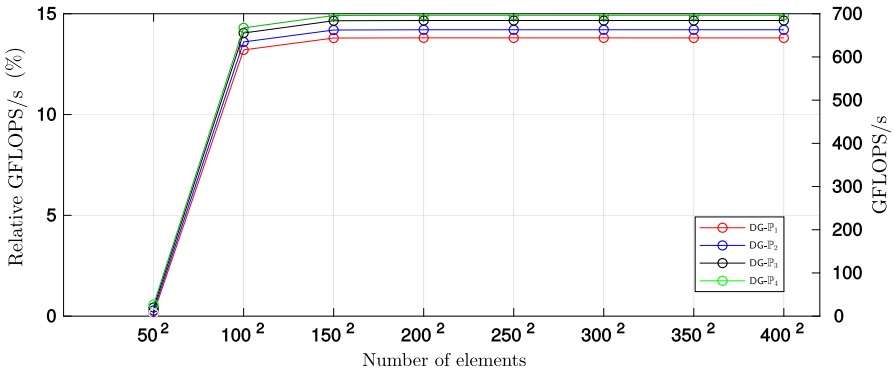

Fig. 22. GFLOPS/s obtained for the space-time predictor *kernel* for several meshes and $DG-\mathbb{P}_N$ schemes.

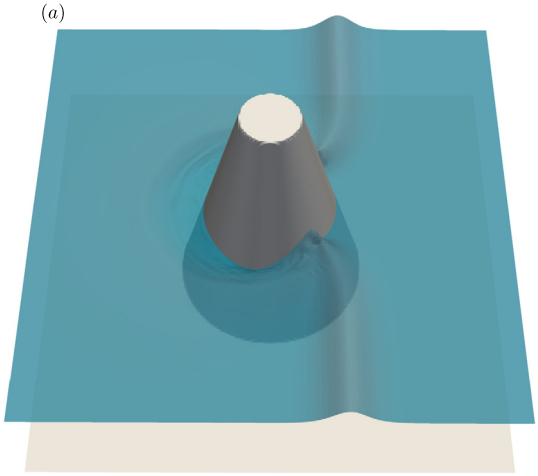

Fig. 23. Snapshot of the free surface profile at time $t = 8$.

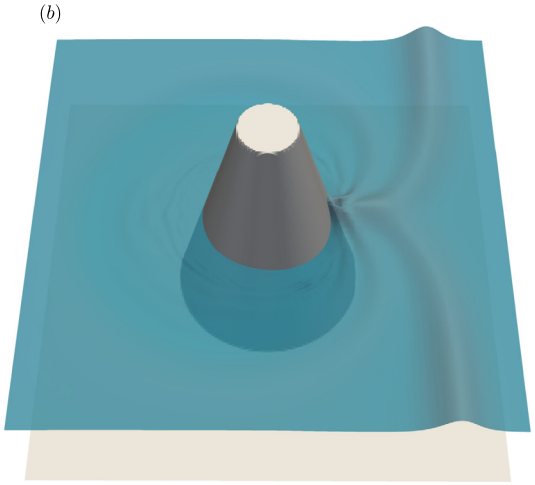

Fig. 24. Snapshot of the free surface profile at time $t = 10$.

Our numerical experiments show that the new approach presented here correctly describes the propagation of solitary waves and is able to accurately preserve their shape even for very long integration times. In particular for solitons, high order DG type schemes are ideal due to their low numerical dissipation and dispersion errors and because the undesired clipping of local extrema, which is typical for second order TVD finite volume schemes, can be completely avoided.

We have also provided numerical results and a detailed comparison with experimental data for more complex test problems in one and two space dimension. The comparison with experimental data requires to incorporate dispersive effects to capture faithfully wave propagation in the vicinity of the continental shelf, in particular involving complex processes such as wave run-up, shoaling, the appearance of higher harmonics and wet-dry areas. Numerical evidence has clearly shown that a simple shallow water model is *not* able to capture all these physical effects simultaneously, hence more sophisticated non-hydrostatic models are needed.

The proposed model and the numerical scheme presented in this work provide thus an efficient and accurate approach to model dispersive effects in the propagation of waves near coastal areas and intermediate waters.

Further research will concern a direct comparison of the new mathematical model proposed in this paper with the alternative hyperbolic reformulation of the Serre-Green-Naghdi (SGN) equations recently forwarded by Favrie and Gavrilyuk in [31].

Acknowledgements

This research has been supported by the Spanish Government and FEDER through the research project MTM2015-70490-C2-1-R. Moreover the authors have received funding from the European Union's Horizon 2020 Research and Innovation Programme under the Marie Sklodowska-Curie grant agreement no. 642768 and under the project *ExaHyPE*, grant no. 671698 (call FET-HPC-1-2014).

MD also acknowledges funding from the Italian Ministry of Education, University and Research (MIUR) via the *Progetto Premiale INDAM-INGV SIES*, via the project PRIN 2017 *Innovative numerical methods for evolutionary partial differential equations and applications* and via the Departments of Excellence Initiative 2018–2022 attributed to DICAM of the University of Trento (grant L. 232/2016). MD has also received support from the University of Trento in the frame of the Strategic Initiative *Modeling and Simulation*.

Last but not least, the authors would like to thank the two anonymous referees for their very helpful and constructive comments, which led to a substantial improvement of the overall quality and readability of this paper.

Appendix A. GPU implementation

We are also interested in the application of our algorithm to real-life problems, such as the simulation of channels, dam–break problems, ocean currents, tsunami wave propagation, etc. Simulating those phenomena requires long time simulations in big computational domains. Thus, extremely efficient implementations are needed to be able to analyse those problems in low computational time.

The numerical scheme presented here exhibits a high potential for data parallelization. This fact suggests the design of an efficient parallel implementation of the numerical scheme. NVIDIA has developed the CUDA programming toolkit [130] for modern Graphics Processor Units (GPUs). CUDA includes an extension of the C language and facilitates the programming on GPUs for general purpose applications by preventing the programmer to deal with low level language programming on GPU.

In this appendix, guidelines for the implementation of the numerical scheme presented in the previous sections are given. The general steps of the parallel implementation are shown in Fig. 25. Each step executed on the GPU is assigned to a CUDA *kernel*, which is a function executed on the GPU. Let us describe the main loop of the program.

At the beginning of the algorithm we build the *main grid*, that corresponds to the Cartesian finite element mesh $\{\boldsymbol{\Omega}_i\}$ as well as the finite-volume subcells $\{\boldsymbol{\Omega}_{i,S}\}$, that we will call the *subcell grid*. For each element $\boldsymbol{\Omega}_i$ on the *main grid* we store the degrees of freedom of the variables h, hu, hv, hw and hp, in one array $\mathbf{u}$ of type `double5`.[1] The bathymetry H is stored in another array of type `double`.

Once the data structure is created, the grids are initialized. To do so, we first compute the mean values from a given initial condition on the *subcell grid* and they are stored in a vector $\mathbf{v}_0$ of type `double5`. In a second place, the *main grid* is initialized via L2 projection of the initial condition onto the discrete approximation space of the DG scheme. Now the initial time step size Δt is computed and the main loop in time of the algorithm starts, where the numerical scheme is iterated until the final simulation time is reached. A series of CUDA *kernels* will do the following tasks:

1. **Space-time predictor and sum regular contributions:** Each thread computes for each element on the *main grid* the degrees of freedom of the predictor solution $\mathbf{q}_h$. This *kernel* also computes the local regular contribution to each element following (19), avoiding the computation of the path-integral. This regular contribution (volume integral) is stored in $\mathbf{u}$. Note that the *kernel* acts over each element $\boldsymbol{\Omega}_i$ of the *main grid* and is the most compute intensive but also the most parallelizable part of the proposed communication-avoiding numerical scheme.
2. **Solve the Riemann-problems and sum jump contributions:** In this step, the boundary extrapolated states at the left $\mathbf{q}_h^-$ and at the right $\mathbf{q}_h^+$ of the interface are computed from the degrees of freedom $\mathbf{q}_h$, obtained within the previous

[1] The `double5` data type represents structures with five double precision real components.

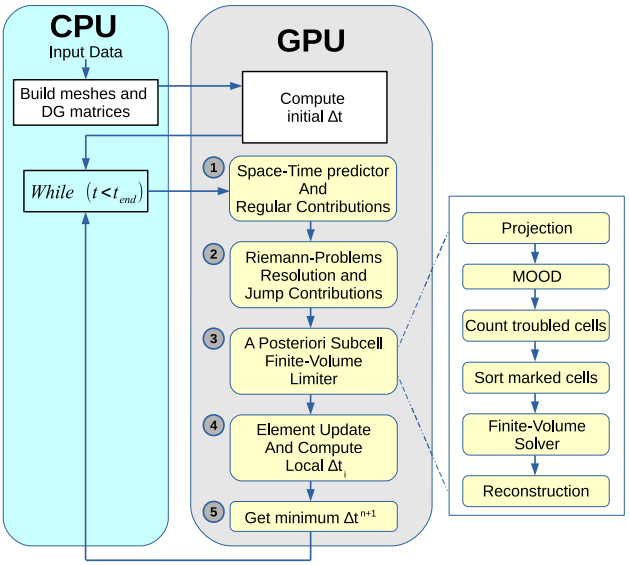

Fig. 25. Parallel CUDA implementation.

predictor *kernel* for each interface of a given element Ω_i. After that, the path-integral is computed. Then, the value of the variable $\mathbf{u}$ is updated to include also the jump contribution:

$$\mathbf{u} \leftarrow \mathbf{u} + \int_{t^n}^{t^{n+1}} \int_{\partial\Omega_i} \Phi_k \mathcal{D}^- \left(\mathbf{q}_h^-, \mathbf{q}_h^+\right) \cdot \mathbf{n} \, dS \, dt.$$

3. **A posteriori subcell finite volume limiter:**
Let us give some brief ideas employed in the implementation of the subcell finite volume solver. It is clear that a *kernel* to project the DG polynomial onto the *subcell grid* must be implemented. The same applies for the reconstruction procedure. Also a third *kernel* to check the validity of the candidate solution computed with the unlimited ADER-DG is also needed. Note that, the third *kernel* can be launched whenever the projected values onto the *subcell grid* are already computed. Additionally, this third *kernel* can not be combined with the first one, since to check the validity of the numerical solution, the corresponding Voronoi neighbours are needed. Therefore, it is justified to split this procedure into at least three parts.
Another *kernel* to evolve the numerical solution with the finite volume solver is needed. In this case, the *kernel* will compute a numerical solution for a given marked $\mathbf{\Omega}_{i,S}$ of the *subcell grid* at the previous step, stored in $\mathbf{v_0}$. We would like to remark that this procedure applied to two different elements $\mathbf{\Omega}_{i,S}$, $\mathbf{\Omega}'_{i,S}$ are completely independent. Thus each thread acts over different marked subcells independently.
Note that the finite-volume *kernel* only acts on a subset contained in the *subcell grid*, in particular the *kernel* acts over those subcells that are marked with the MOOD detector. The number of elements of the subgrid is changing along the simulation, since the number of troubled cells changes. Due to that, a *kernel* has been implemented that takes advantage of such a situation. Let us now describe the following *kernels* that define the steps during the a posteriori subcell finite volume limiter procedure:
- Projection: In this *kernel*, $\mathbf{u}$ that contains $\mathbf{u}_h(\mathbf{x}, t^{n+1})$ is projected onto the subcells and stored in an array of type `double5` $\mathbf{v_1}$.
- A posteriori MOOD detection procedure: Each thread checks the relaxed discrete maximum principle, the positivity of the water height and the presence of floating point errors (`NaN`) within each cell $\mathbf{\Omega}_i$. This kernel also uses $\mathbf{v}_0$.
To do so, an array of type `int2` is defined on the *main grid* as (fl_i, pos_i), being

$$fl_i = \begin{cases} 1, & \text{if } \mathbf{\Omega}_i \text{ is detected as a troubled cell} \\ 0, & \text{Otherwise} \end{cases}, \quad pos_i = \begin{cases} i, & \text{if } fl_i = 1 \\ -1, & \text{if } fl_i = 0 \end{cases}$$

- Counting troubled cells: In this step, a *kernel* is used to count the number of troubled cells, and it is stored in a `integer` variable $M_{troubled}$. To do so, we remark that:

$$M_{troubled} = \sum_i fl_i,$$

that is computed by applying a reduction algorithm in GPU, similarly to what is done in [131] and [132].

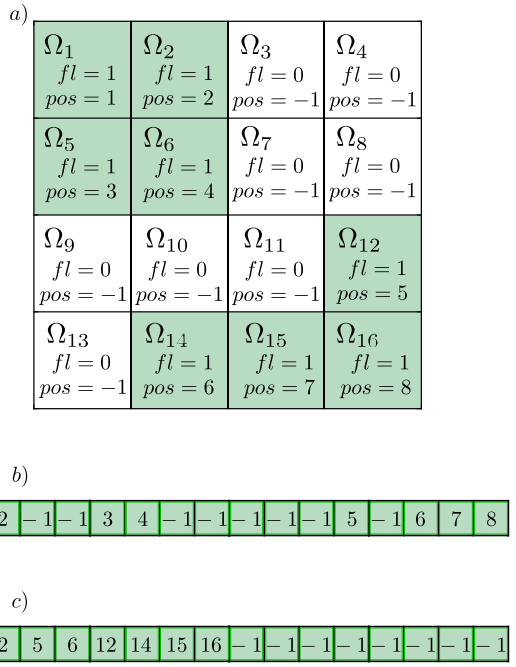

Fig. 26. Troubled cells marking process. In *a*), an example of tagging. In *b*) and *c*) the array of integers pos_i before and after the *Sorting troubled cells kernel*.

- Sorting troubled cells: A sorting algorithm is applied to the vector pos_i. The algorithm separates positive from negative values. In that way, we can separate the marked index cells from the rest (see Fig. 26 to better clarify). To do so, we use the `Thrust` c++ library for CUDA.
- Finite volume solver: This *kernel* contains a finite volume solver that computes a numerical solution on the time-dependent cluster formed by the marked subcells, whose indexes are ordered and stored in pos_i. Thanks to that, the *kernel* it is configured on the small grid that contains the troubled cells and acts in parallel over a smaller set of data. Note that with this procedure, no communication is needed for the computation of two different marked subcells. The results are stored in $\mathbf{v_1}$.
- Reconstruction: In this *kernel*, the degree of freedom of the DG polynomial, are reconstructed from the new computed mean values $\mathbf{v_1}$ that contains the solution given by the finite-volume method at the previous step. This *kernel*, as the previous one, only acts on those subcells that are marked as troubled.

4. **Element update and computing the local Δt_i^{n+1}:** In this *kernel* the mean values are updated, $\mathbf{v_1} \leftarrow \mathbf{v_0}$, and a local Δt_i^{n+1} is computed for each element of the *main grid*.
5. **Computing the minimum of all local Δ_i^{n+1}:** Similarly to what is done in [131] and [132], the minimum of all the local Δt_i^{n+1} values is obtained by applying a reduction algorithm in GPU. This value shall be used as precomputed Δt^{n+1} for the next step of the loop.

References

[1] M.-O. Bristeau, A. Mangeney, J. Sainte-Marie, N. Seguin, An energy-consistent depth-averaged Euler system: derivation and properties, Discrete Contin. Dyn. Syst., Ser. B 20 (4) (2015) 961–988.
[2] M. Abbott, A. McCowan, I. Warren, Accuracy of short wave numerical models, J. Hydraul. Eng. 110 (10) (1984) 1287–1301.
[3] J. Boussinesq, Théorie des ondes et des remous qui se propagent le long dun canal rectangulaire horizontal, en communiquant au liquide contenu dans ce canal des vitesses sensiblement pareilles de la surface au fond, J. Math. Pures Appl. 17 (1872) 55–108.
[4] A. Green, P. Naghdi, A derivation of equations for wave propagation in water of variable depth, Fluid Mech. 78 (1976) 237–246.
[5] P. Madsen, O. Sorensen, A new form of the Boussinesq equations with improved linear dispersion characteristics. Part 2: A slowing varying bathymetry, Coast. Eng. 18 (1992) 183–204.
[6] O. Nwogu, An alternative form of the Boussinesq equations for nearshore wave propagation, J. Waterw. Port Coast. Ocean Eng. 119 (1994) 618–638.
[7] D. Peregrine, Long waves on a beach, Fluid Mech. 27 (4) (1967) 815–827.
[8] G. Wei, J. Kirby, S. Grilli, R. Subramanya, A fully nonlinear Boussinesq model for surface waves. Part 1. Highly nonlinear unsteady waves, J. Fluid Mech. 294 (1) (1995) 71.
[9] G. Whitham Wiley, Linear and nonlinear waves, Earthq. Eng. Struct. Dyn. 4 (5) (1976) 518.
[10] J. Witting, A unified model for the evolution nonlinear water waves, J. Comput. Phys. 56 (2) (1984) 203–236.
[11] V. Casulli, A semi-implicit finite difference method for non-hydrostatic free-surface flows, Int. J. Numer. Methods Fluids 30 (1999) 425–440.
[12] V. Casulli, P. Zanolli, Semi-implicit numerical modeling of nonhydrostatic free-surface flows for environmental problems, Math. Comput. Model. 36 (2002) 1131–1149.
[13] V. Casulli, A high-resolution wetting and drying algorithm for free-surface hydrodynamics, Int. J. Numer. Methods Fluids 60 (2009) 391–408.

[14] V. Casulli, G.S. Stelling, Semi-implicit subgrid modelling of three-dimensional free-surface flows, Int. J. Numer. Methods Fluids 67 (2011) 441–449.
[15] L. Brugnano, V. Casulli, Iterative solution of piecewise linear systems, SIAM J. Sci. Comput. 30 (2007) 463–472.
[16] V. Casulli, A semi-implicit numerical method for the free-surface Navier-Stokes equations, Int. J. Numer. Methods Fluids 74 (2014) 605–622.
[17] A. Jeschke, G.K. Pedersen, S. Vater, J. Behrens, Depth-averaged non-hydrostatic extension for shallow water equations with quadratic vertical pressure profile: equivalence to Boussinesq-type equations, Int. J. Numer. Methods Fluids 84 (10) (2017) 569–583.
[18] P. Lynett, P.L. Liu, A two-layer approach to wave modelling, Proc. R. Soc. Lond., Ser. A, Math. Phys. Eng. Sci. 460 (2049) (2004) 2637–2669.
[19] P. Lynett, P.-F. Liu, Linear analysis of the multi-layer model, Coast. Eng. 51 (2004) 439–454.
[20] C.H. Wu, C.-C. Young, Q. Chen, P.J. Lynett, Efficient nonhydrostatic modeling of surface waves from deep to shallow water, J. Waterw. Port Coast. Ocean Eng. 136 (2) (2010) 104–118, https://doi.org/10.1061/(ASCE)WW.1943-5460.0000032.
[21] E. Fernández-Nieto, M. Parisot, Y. Penel, J. Sainte-Marie, A hierarchy of non-hydrostatic layer-averaged approximation of Euler equations for free surface flows, https://hal.archives-ouvertes.fr/hal-01324012v1, 2017.
[22] Y. Yamazaki, Z. Kowalik, K. Cheung, Depth-integrated, non-hydrostatic model for wave breaking and run-up, Int. J. Numer. Methods Fluids 61 (2008) 473–497.
[23] C. Escalante, T. Morales, M. Castro, Non-hydrostatic pressure shallow flows: GPU implementation using finite volume and finite difference scheme, Appl. Math. Comput. 338 (2018) 631–659.
[24] M.J. Castro, C. Escalante, T. Morales de Luna, Modelling and simulation of non-hydrostatic shallow flows, in: C. Cancès, P. Omnes (Eds.), Finite Volumes for Complex Applications VIII - Hyperbolic, Elliptic and Parabolic Problems, Springer International Publishing, Cham, 2017, pp. 119–126.
[25] C. Escalante, M. Castro, E. Fernández-Nieto, T. Morales, An efficient two-layer non-hydrostatic approach for dispersive water waves, J. Sci. Comput. 79 (2019) 273–320.
[26] M. Kazolea, A. Delis, C. Synolakis, Numerical treatment of wave breaking on unstructured finite volume approximations for extended Boussinesq-type equations, J. Comput. Phys. 271 (2014) 281–305.
[27] M. Tonelli, M. Petti, Hybrid finite volume – finite difference scheme for 2dh improved Boussinesq equations, Coast. Eng. 56 (5–6) (2009) 609–620.
[28] M. Ricchiuto, A. Filippini, Upwind residual discretization of enhanced Boussinesq equations for wave propagation over complex bathymetries, J. Comput. Phys. 271 (2014) 306–341.
[29] D. Lannes, F. Marche, A new class of fully nonlinear and weakly dispersive Green–Naghdi models for efficient 2d simulations, J. Comput. Phys. 282 (2015) 238–268.
[30] C. Cattaneo, Sur une forme de l'équation de la chaleur éliminant le paradoxe d'une propagation instantanée, C. R. Acad. Sci. 247 (1958) 431–433.
[31] N. Favrie, S. Gavrilyuk, A rapid numerical method for solving Serre–Green–Naghdi equations describing long free surface gravity waves, Nonlinearity 30 (7) (2017) 2718.
[32] A. Mazaheri, M. Ricchiuto, H. Nishikawa, A first-order hyperbolic system approach for dispersion, J. Comput. Phys. 321 (Supplement C) (2016) 593–605.
[33] G. Grosso, M. Antuono, M. Brocchini, Dispersive nonlinear shallow-water equations: some preliminary numerical results, J. Eng. Math. 67 (1) (2010) 71–84.
[34] I. Peshkov, E. Romenski, A hyperbolic model for viscous Newtonian flows, Contin. Mech. Thermodyn. 28 (2016) 85–104.
[35] M. Dumbser, I. Peshkov, E. Romenski, O. Zanotti, High order ADER schemes for a unified first order hyperbolic formulation of continuum mechanics: viscous heat–conducting fluids and elastic solids, J. Comput. Phys. 314 (2016) 824–862.
[36] M. Dumbser, I. Peshkov, E. Romenski, O. Zanotti, High order ADER schemes for a unified first order hyperbolic formulation of Newtonian continuum mechanics coupled with electro-dynamics, J. Comput. Phys. 348 (2017) 298–342.
[37] C. Munz, P. Omnes, R. Schneider, E. Sonnendrücker, U. Voss, Divergence correction techniques for Maxwell solvers based on a hyperbolic model, J. Comput. Phys. 161 (2000) 484–511.
[38] A. Dedner, F. Kemm, D. Kröner, C.-D. Munz, T. Schnitzer, M. Wesenberg, Hyperbolic divergence cleaning for the MHD equations, J. Comput. Phys. 175 (2002) 645–673.
[39] W.H. Reed, T.R. Hill, Triangular Mesh Methods for the Neutron Transport Equation, Tech. rep., University of California, Los Alamos Scientific Laboratory, 1973.
[40] B. Cockburn, C. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws II: general framework, Math. Comput. 52 (186) (1989) 411–435.
[41] B. Cockburn, S. Lin, C. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: one-dimensional systems, J. Comput. Phys. 84 (1) (1989) 90–113.
[42] B. Cockburn, S. Hou, C. Shu, The Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: the multidimensional case, Math. Comput. 54 (190) (1990) 545–581.
[43] B. Cockburn, C. Shu, The Runge-Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems, J. Comput. Phys. 141 (2) (1998) 199–224.
[44] B. Cockburn, C. Shu, The local discontinuous Galerkin method for time-dependent convection-diffusion systems, SIAM J. Numer. Anal. 35 (6) (1998) 2440–2463.
[45] F. Bassi, S. Rebay, A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier-Stokes equations, J. Comput. Phys. 131 (1997) 267–279.
[46] G. Gassner, F. Lörcher, C. Munz, A contribution to the construction of diffusion fluxes for finite volume and discontinuous Galerkin schemes, J. Comput. Phys. 224 (2007) 1049–1063.
[47] G. Gassner, F. Lörcher, C.D. Munz, A discontinuous Galerkin scheme based on a space-time expansion II. Viscous flow equations in multi dimensions, J. Sci. Comput. 34 (2008) 260–286.
[48] R. Hartmann, P. Houston, An optimal order interior penalty discontinuous Galerkin discretization of the compressible Navier–Tokes equations, J. Comput. Phys. 227 (2008) 9670–9685.
[49] J. Yan, C. Shu, A local discontinuous Galerkin method for KdV type equations, SIAM J. Numer. Anal. 40 (2) (2002) 769–791.
[50] J. Yan, C. Shu, Local discontinuous Galerkin methods for partial differential equations with higher order derivatives, J. Sci. Comput. 17 (1–4) (2002) 27–47.
[51] D. Levy, C. Shu, J. Yan, Local discontinuous Galerkin methods for nonlinear dispersive equations, J. Comput. Phys. 196 (2) (2004) 751–772.
[52] C. Eskilsson, S.J. Sherwin, Spectral/hp discontinuous Galerkin methods for modelling 2D Boussinesq equations, J. Comput. Phys. 212 (2) (2006) 566–589.
[53] C. Eskilsson, S.J. Sherwin, L. Bergdahl, An unstructured spectral/hp element model for enhanced Boussinesq-type equations, Coast. Eng. 53 (11) (2006) 947–963.
[54] A. Engsig-Karup, J. Hesthaven, H. Bingham, T. Warburton, DG-FEM solution for nonlinear wave-structure interaction using Boussinesq-type equations, Coast. Eng. 55 (2008) 197–208.
[55] S. Kontos, H. Bingham, O. Lindberg, P. Engsig-Karup, A robust WENO scheme for nonlinear waves in a moving reference frame, J. Hydrodynam. Ser. B 28 (2016) 482–488.
[56] M. Dumbser, M. Facchini, A local space-time discontinuous Galerkin method for Boussinesq-type equations, Appl. Math. Comput. 272 (2016) 336–346.

[57] J. van der Vegt, H. van der Ven, Space–time discontinuous Galerkin finite element method with dynamic grid motion for inviscid compressible flows: I. General formulation, J. Comput. Phys. 182 (2) (2002) 546–585.
[58] H. van der Ven, J.J.W. van der Vegt, Space–time discontinuous Galerkin finite element method with dynamic grid motion for inviscid compressible flows: II. Efficient flux quadrature, Comput. Methods Appl. Mech. Eng. 191 (41–42) (2002) 4747–4780.
[59] C.M. Klaij, J.J.W. van der Vegt, H. van der Ven, Space–time discontinuous Galerkin method for the compressible Navier–Stokes equations, J. Comput. Phys. 217 (2) (2006) 589–611.
[60] M. Feistauer, V. Kucera, K. Najzar, J. Prokopová, Analysis of space–time discontinuous Galerkin method for nonlinear convection–diffusion problems, Numer. Math. 117 (2) (2011) 251–288.
[61] J. Cesenek, M. Feistauer, Theory of the space-time discontinuous Galerkin method for nonstationary parabolic problems with nonlinear convection and diffusion, SIAM J. Numer. Anal. 50 (3) (2012) 1181–1206.
[62] M. Dumbser, C. Munz, Building blocks for arbitrary high order discontinuous Galerkin schemes, J. Sci. Comput. 27 (2006) 215–230.
[63] M. Dumbser, C. Enaux, E.F. Toro, Finite volume schemes of very high order of accuracy for stiff hyperbolic balance laws, J. Comput. Phys. 227 (8) (2008) 3971–4001.
[64] M. Dumbser, D. Balsara, E. Toro, C. Munz, A unified framework for the construction of one-step finite volume and discontinuous Galerkin schemes on unstructured meshes, J. Comput. Phys. 227 (18) (2008) 8209–8253.
[65] V. Titarev, E. Toro, ADER: arbitrary high order Godunov approach, J. Sci. Comput. 17 (1–4) (2002) 609–618.
[66] E. Toro, V. Titarev, Solution of the generalized Riemann problem for advection-reaction equations, Proc. R. Soc. Lond. (2002) 271–281.
[67] V. Titarev, E. Toro, ADER schemes for three-dimensional nonlinear hyperbolic systems, J. Comput. Phys. 204 (2005) 715–736.
[68] E.F. Toro, V.A. Titarev, Derivative Riemann solvers for systems of conservation laws and ADER methods, J. Comput. Phys. 212 (1) (2006) 150–165.
[69] M. Dumbser, O. Zanotti, R. Loubère, S. Diot, A posteriori subcell limiting of the discontinuous Galerkin finite element method for hyperbolic conservation laws, J. Comput. Phys. 278 (2014) 47–75.
[70] O. Zanotti, F. Fambri, M. Dumbser, A. Hidalgo, Space-time adaptive ADER discontinuous Galerkin finite element schemes with a posteriori sub-cell finite volume limiting, Comput. Fluids 118 (2015) 204–224.
[71] C. Parés, M.J. Castro, On the well-balance property of Roe's method for nonconservative hyperbolic systems. Applications to shallow-water systems, ESAIM: Math. Model. Numer. Anal. 38 (5) (2004) 821–852.
[72] M. Castro, E. Fernández-Nieto, A class of computationally fast first order finite volume solvers: PVM methods, SIAM J. Sci. Comput. 34 (4) (2012) 173–196.
[73] M.J. Castro, J.M. Gallardo, C. Parés, High-order finite volume schemes based on reconstruction of states for solving hyperbolic systems with nonconservative products. Applications to shallow-water systems, Math. Comput. 75 (2006) 1103–1134.
[74] C. Parés, Numerical methods for nonconservative hyperbolic systems: a theoretical framework, SIAM J. Numer. Anal. 44 (2006) 300–321.
[75] M.J. Castro Díaz, E.D. Fernández-Nieto, A.M. Ferreiro, J.A. García-Rodríguez, C. Parés, High order extensions of Roe schemes for two-dimensional nonconservative hyperbolic systems, J. Sci. Comput. 39 (1) (2009) 67–114.
[76] M. Castro, A. Pardo, C. Parés, E. Toro, On some fast well-balanced first order solvers for nonconservative systems 79 (271) (2010) 1427–1472.
[77] A. Harten, P.D. Lax, B. van Leer, On upstream differencing and Godunov-type schemes for hyperbolic conservation laws, SIAM Rev. 25 (1) (1983) 35–61.
[78] M. Dumbser, D. Balsara, A new, efficient formulation of the HLLEM Riemann solver for general conservative and non-conservative hyperbolic systems, J. Comput. Phys. 304 (2016) 275–319.
[79] S. Clain, S. Diot, R. Loubère, A high–order finite volume method for systems of conservation laws – multi – dimensional optimal order detection (MOOD), J. Comput. Phys. 230 (10) (2011) 4028–4050.
[80] S. Diot, S. Clain, R. Loubère, Improved detection criteria for the multi-dimensional optimal order detection (MOOD) on unstructured meshes with very high-order polynomials, Comput. Fluids 64 (2012) 43–63.
[81] S. Diot, R. Loubère, S. Clain, The MOOD method in the three-dimensional case: very-high-order finite volume method for hyperbolic systems, Int. J. Numer. Methods Fluids 73 (2013) 362–392.
[82] R. Loubère, M. Dumbser, S. Diot, A new family of high order unstructured MOOD and ADER finite volume schemes for multidimensional systems of hyperbolic conservation laws, Commun. Comput. Phys. 16 (2014) 718–763.
[83] V. Roeber, K.F. Cheung, M.H. Kobayashi, Shock-capturing Boussinesq-type model for nearshore wave processes, Coast. Eng. 57 (2010) 407–423.
[84] R. Manning, On the flow of water in open channels and pipes, Trans. Inst. Civ. Eng. Irel. 20 (1891) 161–207.
[85] J. Monaghan, Simulating free surface flows with SPH, J. Comput. Phys. 110 (1994) 399–406.
[86] A. Ferrari, M. Dumbser, E. Toro, A. Armanini, A new 3D parallel SPH scheme for free surface flows, Comput. Fluids 38 (2009) 1203–1217.
[87] M. Dumbser, A simple two-phase method for the simulation of complex free surface flows, Comput. Methods Appl. Mech. Eng. 200 (2011) 1204–1219.
[88] S. Godunov, An interesting class of quasilinear systems, Dokl. Akad. Nauk SSSR 139 (3) (1961) 521–523.
[89] S. Godunov, E. Romenski, Nonstationary equations of nonlinear elasticity theory in eulerian coordinates, J. Appl. Mech. Tech. Phys. 13 (6) (1972) 868–884.
[90] E. Romenski, Hyperbolic systems of thermodynamically compatible conservation laws in continuum mechanics, Math. Comput. Model. 28 (10) (1998) 115–130.
[91] S. Godunov, E. Romenski, Elements of Continuum Mechanics and Conservation Laws, Kluwer Academic/Plenum Publishers, 2003.
[92] H.A. Schäffer, P.A. Madsen, Further enhancements of Boussinesq-type equations, Coast. Eng. 26 (1) (1995) 1–14.
[93] N. Aïssiouene, M.-O. Bristeau, E. Godlewski, A. Mangeney, C. Parés, J. Sainte-Marie, Application of a combined finite element—finite volume method to a 2d non-hydrostatic shallow water problem, in: C. Cancès, P. Omnes (Eds.), Finite Volumes for Complex Applications VIII - Hyperbolic, Elliptic and Parabolic Problems, Springer International Publishing, Cham, 2017, pp. 219–226.
[94] S. Rhebergen, O. Bokhove, J. van der Vegt, Discontinuous Galerkin finite element methods for hyperbolic nonconservative partial differential equations, J. Comput. Phys. 227 (2008) 1887–1922.
[95] M. Dumbser, M. Castro, C. Parés, E. Toro, ADER schemes on unstructured meshes for non-conservative hyperbolic systems: applications to geophysical flows, Comput. Fluids 38 (2009) 1731–1748.
[96] M. Dumbser, A. Hidalgo, M. Castro, C. Parés, E. Toro, FORCE schemes on unstructured meshes II: non–conservative hyperbolic systems, Comput. Methods Appl. Mech. Eng. 199 (2010) 625–647.
[97] L.O. Müller, C. Parés, E.F. Toro, Well-balanced high-order numerical schemes for one-dimensional blood flow in vessels with varying mechanical properties, J. Comput. Phys. 242 (2013) 53–85.
[98] L. Müller, E. Toro, Well-balanced high-order solver for blood flow in networks of vessels with variable properties, Int. J. Numer. Methods Biomed. Eng. 29 (12) (2013) 1388–1411.
[99] E. Gaburro, M. Dumbser, M. Castro, Direct arbitrary-Lagrangian-Eulerian finite volume schemes on moving nonconforming unstructured meshes, Comput. Fluids 159 (2017) 254–275.
[100] E. Gaburro, M. Castro, M. Dumbser, Well balanced arbitrary-Lagrangian-Eulerian finite volume schemes on moving nonconforming meshes for the Euler equations of gas dynamics with gravity, Mon. Not. R. Astron. Soc. 477 (2018) 2251–2275.

[101] M. Dumbser, E.F. Toro, A simple extension of the Osher Riemann solver to non-conservative hyperbolic systems, J. Sci. Comput. 48 (2011) 70–88.
[102] M. Castro, J. Gallardo, A. Marquina, Approximate Osher–Solomon schemes for hyperbolic systems, Appl. Math. Comput. 272 (2016) 347–368.
[103] B. Einfeldt, P.L. Roe, C.D. Munz, B. Sjogreen, On Godunov-type methods near low densities, J. Comput. Phys. 92 (1991) 273–295.
[104] A. Harten, B. Engquist, S. Osher, S. Chakravarthy, Uniformly high order essentially non-oscillatory schemes, III, J. Comput. Phys. 71 (1987) 231–303.
[105] A. Hidalgo, M. Dumbser, ADER schemes for nonlinear systems of stiff advection-diffusion-reaction equations, J. Sci. Comput. 48 (2011) 173–189.
[106] O. Zanotti, M. Dumbser, Efficient conservative ADER schemes based on WENO reconstruction and space-time predictor in primitive variables, Comput. Astrophys. Cosmol. 3 (1) (2016) 1.
[107] F. Fambri, M. Dumbser, S. Köppel, L. Rezzolla, O. Zanotti, ADER discontinuous Galerkin schemes for general-relativistic ideal magnetohydrodynamics, Mon. Not. R. Astron. Soc. 477 (2018) 4543–4564.
[108] B. Owren, M. Zennaro, Derivation of efficient, continuous, explicit Runge–Kutta methods, SIAM J. Sci. Stat. Comput. 13 (1992) 1488–1501.
[109] G. Gassner, M. Dumbser, F. Hindenlang, C. Munz, Explicit one–step time discretizations for discontinuous Galerkin and finite volume schemes based on local predictors, J. Comput. Phys. 230 (11) (2011) 4232–4247.
[110] D. Charrier, T. Weinzierl, Stop talking to me – a communication-avoiding ADER-DG realisation, https://arxiv.org/abs/1801.08682, 2018.
[111] M. Dumbser, R. Loubère, A simple robust and accurate a posteriori sub–cell finite volume limiter for the discontinuous Galerkin method on unstructured meshes, J. Comput. Phys. 319 (2016) 163–199.
[112] W. Boscheri, M. Dumbser, Arbitrary–Lagrangian–Eulerian discontinuous Galerkin schemes with a posteriori subcell finite volume limiting on moving unstructured meshes, J. Comput. Phys. 346 (2017) 449–479.
[113] B.V. Leer, Towards the ultimate conservative difference scheme. V. A second order sequel to Godunov's method, J. Comput. Phys. 32 (1979) 101–136.
[114] S. Gottlieb, C.-W. Shu, Total variation diminishing Runge-Kutta schemes, Math. Comput. 67 (221) (1998) 73–85.
[115] P.A. Madsen, H.B. Bingham, H.A. Schäffer, Boussinesq-type formulations for fully nonlinear and extremely dispersive water waves: derivation and analysis, Proc., Math. Phys. Eng. Sci. 459 (2033) (2003) 1075–1104.
[116] M. Castro, A. Ferreiro, J. García, J. González, J. Macías, C. Parés, M. Vázquez-Cendón, The numerical treatment of wet/dry fronts in shallow flows: applications to one-layer and two-layer systems, Math. Comput. Model. 42 (3–4) (2005) 419–439.
[117] J. Macías, M.J. Castro, S. Ortega, C. Escalante, J.M. González-Vida, Performance benchmarking of Tsunami-HySEA model for NTHMP's inundation mapping activities, Pure Appl. Geophys. 174 (8) (2017) 3147–3183.
[118] P.J. Lynett, K. Gately, R. Wilson, L. Montoya, D. Arcas, B. Aytore, Y. Bai, J.D. Bricker, M.J. Castro, K.F. Cheung, C.G. David, G.G. Dogan, C. Escalante, J.M. González-Vida, S.T. Grilli, T.W. Heitmann, J. Horrillo, U. Kânoğlu, R. Kian, J.T. Kirby, W. Li, J. Macías, D.J. Nicolsky, S. Ortega, A. Pampell-Manis, Y.S. Park, V. Roeber, N. Sharghivand, M. Shelby, F. Shi, B. Tehranirad, E. Tolkova, H.K. Thio, D. Velioğlu, A.C. Yalçıner, Y. Yamazaki, A. Zaytsev, Y. Zhang, Inter-model analysis of tsunami-induced coastal currents, Ocean Model. 114 (2017) 14–32.
[119] F. Enet, S. Grilli, Experimental study of tsunami generation by three-dimensional rigid underwater landslides, J. Waterw. Port Coast. Ocean Eng. 133 (6) (2007) 442–454.
[120] G. Ma, F. Shi, J.T. Kirby, Shock-capturing non-hydrostatic model for fully dispersive surface wave processes, Ocean Model. 43 (2012) 22–35.
[121] G. Stelling, M. Zijlema, An accurate and efficient finite-difference algorithm for non-hydrostatic free-surface flow with application to wave propagation, Int. J. Numer. Methods Fluids 43 (1) (2003) 1–23.
[122] M. Dumbser, Arbitrary high order PNPM schemes on unstructured meshes for the compressible Navier–Stokes equations, Comput. Fluids 39 (2010) 60–76.
[123] C. Synolakis, The runup of solitary waves, Fluid Mech. 185 (1987) 523–545.
[124] S. Beji, J. Battjes, Numerical simulation of nonlinear wave propagation over a bar, Coast. Eng. 23 (1994) 1–16.
[125] H. Favre, Ondes de translation dans les canaux découverts, Dunod, Paris, 1935.
[126] A. Treske, Undular bores (Favre-waves) in open channels - experimental studies, J. Hydraul. Res. 32 (3) (1994) 355–370.
[127] S.L. Gavrilyuk, V.Y. Liapidevskii, A.A. Chesnokov, Spilling breakers in shallow water: applications to Favre waves and to the shoaling and breaking of solitary waves, J. Fluid Mech. 808 (2016) 441–468.
[128] M. Briggs, C. Synolakis, G. Harkins, D. Green, Laboratory experiments of tsunami runup on a circular island, Pure Appl. Geophys. 144 (3) (1995) 569–593.
[129] P. Lynett, T. Wu, P.-F. Liu, Modeling wave runup with depth-integrated equations, Coast. Eng. 46 (2) (2002) 89–107.
[130] NVIDIA, Cuda home page, http://www.nvidia.com/object/cuda_home_new.html.
[131] M. de la Asunción, J.-M. Mantas, M.J. Castro Díaz, Programming CUDA-Based GPUs to Simulate Two-Layer Shallow Water Flows, Lecture Notes in Computer Science, vol. 6272, Springer, Berlin / Heidelberg, 2010, pp. 353–364.
[132] M. de la Asunción, J.-M. Mantas, M.J. Castro Díaz, Simulation of one-layer shallow water systems on multicore and CUDA architecture, J. Supercomput. 58 (2) (2011) 206–214.