A well-balanced, positive, entropy-stable, and multi-dimensional-aware finite volume scheme for 2D shallow-water equations with unstructured grids

Alessia Del Grosso, Manuel J. Castro, Agnes Chan, Gérard Gallice, Raphaël Loubère, Pierre-Henri Maire

A well-balanced, positive, entropy-stable, and multi-dimensional-aware finite volume scheme for 2D shallow-water equations with unstructured grids

Alessia Del Grosso*[a,b], Manuel J. Castro[d], Agnes Chan[a,b], Gérard Gallice[c], Raphaël Loubère[a], Pierre-Henri Maire[b]

[a] *Institut de Mathématiques de Bordeaux (IMB), Université de Bordeaux, CNRS, Bordeaux INP, F33400, Talence, France*
[b] *CEA Cesta, 15 avenue des sablières, Le Barp, France*
[c] *Retired fellow from CEA Cesta, France*
[d] *Universidad de Málaga, Dpto. Análisis Matemático, Campus de Teatinos s/n, 29071 Málaga, Spain*

Abstract

In this article, we present a multi-dimensional-aware Eulerian Riemann solver (RS) and its associated finite volume (FV) scheme for the 2D shallow-water equations. This RS, appropriately derived from its associated Lagrangian version, presents the specific feature of coupling all cells in the vicinity of the current one. Consequently, this solver is no longer a 1D RS across one edge. Contrarily, it encounters for genuine multidimensional effects and for the presence of the source term of the SW equations. The associated first order FV numerical scheme ensures well-balanced for lake at rest steady states, positivity preservation and entropy stability properties. Moreover, a second-order accurate extension is proposed based on Runge-Kutta time discretization and piecewise linear limited reconstructions, that preserve the well-balanced character of the first order scheme. We present several 2D tests assessing the good behaviors of the obtained numerical scheme on unstructured mesh. The numerical scheme seems insensitive to spurious numerical instabilities such as the carbuncle effect.

Key words: Lagrangian Riemann solver, Eulerian Riemann solver, Shallow-water equations, Balance laws, Well-balanced scheme, Finite volume schemes.

1. Introduction

Mathematical modeling and computational simulation associated with experimental observations of physical phenomena are the key ingredients for a deeper understanding of human activities under fast environmental changes. The modelling part requires the development and analysis of physically meaningful mathematical models fed by an increasing amount of data. The computational part relies on the development of efficient and reliable computational numerical methods, which are the engine within a simulation code, further operating on super-computers.

*Corresponding author
Email addresses: alessia.del-grosso@math.u-bordeaux.fr (Alessia Del Grosso*), castro@anamat.cie.uma.es (Manuel J. Castro), agnes.chan@math.u-bordeaux.fr (Agnes Chan), gerard.gallice@cea.fr (Gérard Gallice), raphael.loubere@math.u-bordeaux.fr (Raphaël Loubère), pierre-henri.maire@cea.fr (Pierre-Henri Maire)

This paper concerns the simulation of shallow water flows in channels, rivers, lakes and oceans. Such flows can be described by the so-called Shallow Water (SW) model of balance laws. More precisely, we aim to develop a new numerical method to solve the 1D/2D time dependent SW model. A plethora of finite difference, finite volume or finite element, centered or upwind biased numerical methods already exist for such a model, see for instance the textbooks [62, 9, 39] and the reviews [63, 13]. We can not cite exhaustively the contributors of this field as it would be extremely challenging nowadays. Hence, we simply refer the interested readers to these references and those contained within.

One characteristic of systems of balance laws is the presence of source terms. They demand a special treatment to ensure that some steady state solutions of the model are preserved, see for instance [4, 52, 6, 11]. Numerical schemes able to preserve all the steady states of SW model are called *fully well-balanced* [52, 6, 13], while we talk about *well-balanced* schemes when the methods preserve only the steady states with zero-velocity [1, 19]. This work only focuses on the latter notion. In particular, in the case of SW equations, a geometric source term involving the bottom topography is present. As such, we are concerned with the preservation of the so called *lake at rest* steady state. Bermudez and Vázquez [5] were the first to present the issue of well-balancing in the framework of the shallow water system, and since then, many works have been published on the subject, for instance Audusse *et al.* [1] introduced the well-known hydrostatic reconstruction. Other references for first-order well-balanced schemes include [7, 38, 21], to name but a few. Extensions to second (or higher) order well-balanced methods have also been investigated, refer for instance to [8, 10, 13, 14].

To numerically simulate the 2D shallow water system, we design a Finite Volume (FV) scheme of Godunov type [37, 63], which heavily relies on the classical "upwinding" notion for the numerical flux introduced by Roe, also in the presence of source term [58]. In this FV context, the 1D Riemann Problem (RP) and its solver play a central role. It consists of the "simpler" problem of the evolution of an initial condition constituted by a discontinuity separating two constant states after the physical waves emanate from the discontinuity. A Riemann Solver (RS) is a central tool to compute the numerical flux at the interface between two computational cells [64] in a FV scheme.

In this context, following the work accomplished for the gas dynamics model in [15, 36], we rely on the derivation of a RS in the Lagrangian system of coordinates. This specificity allows to design a so-called "simple" RS [32, 33, 35, 34], which takes into account the source term. The simple Lagrangian RS is further expressed in Eulerian coordinates. The link between Lagrangian and Eulerian system of coordinates allows for a clear definition of ordered Eulerian wave speeds. Moreover this allows for the preservation of positivity and entropy stability properties under a well defined CFL like condition, but also an embedded well-balanced property. The simple Eulerian numerical RS is therefore explicitly defined.

However, when 2D is invoked, the specificity of the RS is highlighted. If the RS was to be classically employed in the 1D edge normal direction between two cells, then the obtained FV scheme would also be a classical one (called 'two-point scheme'). In fact, low order classical schemes do not take into account the multi-directional structure of the solutions and may destroy their isotropy. In the literature, one could find different techniques that introduce multidimensional features in the numerical schemes, see for example [22, 23, 24, 25, 46, 47, 56, 40, 3, 41, 60]. In [3], Wendroff proposed a multidimensional Harten-Lax-van Leer (HLL) Riemann solver for the Euler equations, but its extension to high orders or to other systems is not straightforward. In [22, 23], the authors introduced a genuinely two-dimensional Riemann solver based on the HLL and HLLC [64] Riemann solvers, which provides closed-form expressions of the fluxes and allowed an easy extension to high orders. In [40], another version of Wendroff's scheme has been proposed. Extensions to unstructured meshes have been also proposed in [27, 26]. Recently, a new class of genuinely

two-dimensional incomplete Riemann solvers based on AVM-type (Approximate Viscosity Matrix) solvers was introduced in [41] and its extension to balance laws, like the shallow-water system has been considered in [60].

In this work, we illustrate that the simple Eulerian RS can involve a nodal solver, developed when designing Lagrangian FV scheme for gas dynamics [20, 50]. This nodal solver considers all cell states surrounding a given node to determine the value of a nodal vectorial quantity, which can be identified as being the 'material velocity' of a node. This nodal velocity is further used in the definition of the Eulerian face-based numerical flux. As such, the notion of conservation is no more based on the cancellation of the numerical flux across one edge, rather the conservation is retrieved around each node. This allows to couple all cells in the vicinity of the current FV cell, hence the name 'multi-point scheme'. This first-order FV scheme involving source terms has good properties such as the local conservation, positivity and entropy stability under a well-defined CFL condition and well-balanced properties by construction. A classical extension to second order is also presented and further tested on academical and demanding test cases.

At last, in this work, we are also interested in studying the sensitivity of the two-point and multi-point numerical schemes to troublesome tumor-growth-like instabilities such as the Carbuncle [55, 57]. Such instabilities usually appear in low-diffusion methods, for instance based on complete Riemann solver. Hence, many works have focused on adding diffusion to the schemes to remove the spurious oscillations from the numerical results [53, 57], blaming the lack of diffusion of the methods. However, it is still not really clear what causes the Carbuncle, which also explains why it has been defined as "the greatest unresolved problem of classical finite-volume schemes" by Van Leer [65]. One of the hypothesis is that the Carbuncle has a multidimensional nature [29]. As such, methods based on a dimensional flux-splitting (i.e. schemes that apply a 1D solver in each direction) would be prone to this kind of instabilities. Hence, the idea is that, thanks to its multi-dimensional-aware character, the multi-point scheme should be insensitive to the Carbuncle phenomenon and related problems. Indeed, many works have proposed strategies to embed a multi-dimensional character to numerical methods, see for instance [45, 59]. Finally, we highlight that such instabilities have been extensively studied in the gas dynamics framework, but much less in the shallow-water context, refer for instance to [43, 31, 42, 30]. We will also illustrate that Carbuncle-like instability can develop in the SW context.

The paper is structured as follows. First of all, we introduce the model in both Eulerian and Lagrangian coordinate systems in section 2. Next, section 3 provides a detailed description of the 2D FV scheme and the underlying approximate Riemann solver. Then, we summarize the properties that the FV scheme inherits from the simple Riemann solver in section 4, along with presenting the second-order extension of the method. Section 5 gathers the numerical results in 2D. At last, conclusions and perspectives are drawn in the ultimate section.

2. Shallow-water equations

In this work we are concerned by the so-called 2D Shallow-Water (SW) equations, also referred to as the Saint-Venant equations in 1D [18]. Shallow water flows are typical of flows for which the vertical dimension is much smaller than the typical horizontal one, such in ocean, rivers, lakes, close to coasts for example.

One considers the free surface flow of water in a channel assuming that the water is incompressible, non-viscous, non-heat conducting, without bottom friction and subject to gravitational forces. The horizontal plan is given by coordinate x and y. The body force vector is denoted by $\mathbf{g} = (0, -g)$ where g is the acceleration due to gravity, assumed to be constant in this work. The

2D computational domain is denoted by $\mathcal{D}$. The bottom of the channel $B(x, y)$, also called bottom topography, is assumed fixed in time but maybe irregular, while the water depth is denoted by $h(x, y, t)$, see figure 1. At last, the velocity vector of the fluid is referred to as $\mathbf{u} = (u, v)$.

The SW equations are given by the following non-linear system of balance laws:

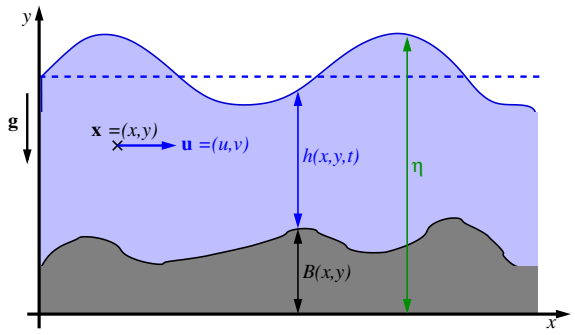


Figure 1: Sketch of shallow-water context.

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathbb{F}(\mathbf{U}) = \mathbf{S}(\mathbf{U}). \tag{1}$$

Here, $\mathbf{U} = \mathbf{U}(\mathbf{x}, t)$, for $\mathbf{x} = (x, y) \in \mathbb{R}^2$ and $t \geq 0$, is the vector of conservative variables which takes values in $\mathbb{R}^3$ and $\mathbb{F} = \mathbb{F}(\mathbf{U})$ is the flux tensor in $\mathbb{R}^3 \times \mathbb{R}^2$, while $\mathbf{S}(\mathbf{U}) \in \mathbb{R}^3$ is the source term involving the bottom topography. The positive integers d = 2 and q = 3 denote respectively the space dimension and the size of the foregoing hyperbolic system. Let $\mathbf{e}_x$, $\mathbf{e}_y$ be the vectors of the Cartesian basis of $\mathbb{R}^2$, then $\mathbf{F}_1 = \mathbb{F}\mathbf{e}_x \in \mathbb{R}^3$ and $\mathbf{F}_2 = \mathbb{F}\mathbf{e}_y \in \mathbb{R}^3$ are the components of the tensor flux. Its divergence then reads

$$\nabla \cdot \mathbb{F}(\mathbf{U}) = \frac{\partial}{\partial x}\mathbf{F}_1(\mathbf{U}) + \frac{\partial}{\partial y}\mathbf{F}_2(\mathbf{U}),$$

so that we can reformulate system (1) in the more classical form

$$\frac{\partial}{\partial t}\mathbf{U} + \frac{\partial}{\partial x}\mathbf{F}_1(\mathbf{U}) + \frac{\partial}{\partial y}\mathbf{F}_2(\mathbf{U}) = \mathbf{S}(\mathbf{U}), \tag{2}$$

where the vector of conserved variables $\mathbf{U}$, the flux vectors $\mathbf{F}_1(\mathbf{U})$ and $\mathbf{F}_2(\mathbf{U})$, and the source term vector $\mathbf{S}(\mathbf{U})$ are explicitly given by

$$\mathbf{U} = \begin{pmatrix} h \\ hu \\ hv \end{pmatrix}, \quad \mathbf{F}_1(\mathbf{U}) = \begin{pmatrix} hu \\ hu^2 + p \\ huv \end{pmatrix}, \quad \mathbf{F}_2(\mathbf{U}) = \begin{pmatrix} hv \\ huv \\ hv^2 + p \end{pmatrix}, \quad \mathbf{S}(\mathbf{U}) = \begin{pmatrix} 0 \\ -gh\partial_x B \\ -gh\partial_y B \end{pmatrix}. \tag{3}$$

The variable p can be seen as the pressure and, in the case of SW equations, it takes the simple form: $p \equiv p(h) = \frac{1}{2}gh^2$. If the source term is identically null, $\mathbf{S}(\mathbf{U}) = \mathbf{0}$, then system (2) becomes a hyperbolic system of conservation laws. Given a unit vector $\mathbf{e} = (e_x, e_y)$, the eigenvalues of the Jacobian matrix $\mathbf{J}_\mathbf{e} = \frac{\partial \mathbf{F}_1}{\partial \mathbf{U}} e_x + \frac{\partial \mathbf{F}_2}{\partial \mathbf{U}} e_y$ are given by

$$\lambda^- = \mathbf{ue} - a, \quad \lambda^- = \mathbf{ue}, \quad \lambda^- = \mathbf{ue} + a,$$

where $a = \sqrt{gh}$ plays the role of the sound-speed for gas dynamics. Moreover, we can also interpret the homogeneous version of system (2) as the isentropic Euler system if $h \equiv \rho$ and a polytropic pressure law $p(h) = Kh^\gamma$ is considered, with $K = \dfrac{g}{2}$ and $\gamma = 2$. With such an analogy, we define the internal energy $\varepsilon = \varepsilon(h)$ with $\dfrac{d\varepsilon}{dh} = \dfrac{p}{h^2}$ and the energy as $E = \varepsilon + \frac{1}{2}\|\mathbf{u}\|^2$, such that the entropy inequality is given by

$$\mathcal{E} = \frac{\partial}{\partial t}(hE) + \frac{\partial}{\partial x}(uhE + pu) + \frac{\partial}{\partial y}(vhE + pv) \leq -gh\left(u\frac{\partial}{\partial x}B + v\frac{\partial}{\partial y}B\right). \tag{4}$$

We can reformulate this inequality in a more compact form as

$$\frac{\partial}{\partial t}(\Sigma) + \nabla \cdot \Theta \leq \Phi, \tag{5}$$

where

$$\Sigma = hE, \qquad \Theta = (uhE + pu, vhE + pv)^t, \qquad \Phi = -gh\left(u\frac{\partial}{\partial x}B + v\frac{\partial}{\partial y}B\right).$$

We remark that in the case of a regular topography, the foregoing equation is equivalent to

$$\mathcal{E}_B = \frac{\partial}{\partial t}(hE_B) + \frac{\partial}{\partial x}(uhE_B + pu) + \frac{\partial}{\partial y}(vhE_B + pv) \leq 0,$$

where $E_B = E + gB$. Indeed the equivalence is easily retrieved [32, 33]:

$$\begin{aligned}
\mathcal{E}_B &= \mathcal{E} + \frac{\partial}{\partial t}(hgB) + \frac{\partial}{\partial x}(uhgB) + \frac{\partial}{\partial y}(vhgB) = \mathcal{E} + gB\left(\frac{\partial}{\partial t}h\right) + \frac{\partial}{\partial x}(uhgB) + \frac{\partial}{\partial y}(vhgB) \\
&= \mathcal{E} + gB\left(-\frac{\partial}{\partial x}(hu) - \frac{\partial}{\partial y}(hv)\right) + \frac{\partial}{\partial x}(uhgB) + \frac{\partial}{\partial y}(vhgB) \\
&= \mathcal{E} + guh\frac{\partial}{\partial x}B + gvh\frac{\partial}{\partial y}B = \mathcal{E} + gh\left(u\frac{\partial}{\partial x}B + v\frac{\partial}{\partial y}B\right),
\end{aligned}$$

therefore $\mathcal{E}_B \leq 0$ is equivalent to (4).

The exact solutions of system (2) are usually not known, generally tremendously complicated, in particular due to the presence of irregularities such as shock waves, rarefaction corners, or contact discontinuities. The set of admissible solutions of system (2) is characterized by the positivity of the water depth h:

$$\mathcal{D} = \{\mathbf{U},\ \text{s. t. } h > 0\}.$$

Numerical methods which are able to maintain numerical solutions within $\mathcal{D}$ are referred to as being *positivity preserving*. Moreover, there exist large sets of stationary solutions for the SW equations when the bottom topography is non zero. Of particular importance are the so-called *Lake at Rest* (LaR) solutions, which are parameterized by a constant η:

$$\forall (x, y) \in \mathcal{D},\ \forall t > 0, \quad u(x, y, t) = 0, \quad h(x, y, t) + B(x, y) = \eta.$$

LaR steady states are important solutions that should be captured and maintained exactly when resorting to numerical methods. In such a case, the numerical method is said to be *well-balanced* (WB) [9, 13].

To conclude, we highlight that, in this work, we consider so-called wet situations, that is states for which $h > 0$. We postpone the issue of dealing with wet/dry transition area for future developments.

2.1. Lagrange-to-Euler mapping

With the aim of designing numerical methods, we briefly describe here the Lagrange-to-Euler mapping, already presented in [36, 15, 34]. Specifically, this mapping will be needed to define the numerical flux of the associated FV method.

To simplify the task, we directly present the Lagrange-to-Euler mapping for the system formulated in a normal direction to a given cell interface, represented by the unit vector $\mathbf{n}$. Since we only use this mapping in this subcase, we refrain from presenting the more complex general description of the model in Lagrangian coordinates. For interested readers, the general description can be found in [26, 50, 49].

First of all, we present the SW system in the normal direction $\mathbf{n}$. Let $\mathbf{t}$ be the unit vector such that $(\mathbf{n}, \mathbf{t})$ is an orthonormal basis attached to a generic interface. The normal and tangential components of the velocity vector $\mathbf{u}$ respectively read $u_\mathbf{n} = \mathbf{u}\cdot\mathbf{n}$ and $u_\mathbf{t} = \mathbf{u}\cdot\mathbf{t}$, with $\mathbf{u} = u_\mathbf{n}\mathbf{n} + u_\mathbf{t}\mathbf{t}$. Then, the variables vector, the flux and the source term projected onto the normal direction $\mathbf{n}$ read

$$\mathbf{U}_\mathbf{n} = \begin{pmatrix} h \\ hu_\mathbf{n} \\ hu_\mathbf{t} \end{pmatrix}, \qquad \mathbf{F}_\mathbf{n} = \mathbb{F}\mathbf{n} = \begin{pmatrix} hu_\mathbf{n} \\ hu_\mathbf{n}^2 + p(h) \\ hu_\mathbf{n}u_\mathbf{t} \end{pmatrix}, \qquad \mathbf{S}_\mathbf{n} = \begin{pmatrix} 0 \\ -gh(\nabla B)_\mathbf{n} \\ 0 \end{pmatrix},$$

with $(\nabla B)_\mathbf{n} = \nabla B \cdot \mathbf{n}$. Therefore, the one-dimensional Eulerian system associated to the SW system in the $\mathbf{n}$ direction, where $x_\mathbf{n} = \mathbf{x}\cdot\mathbf{n}$, reads

$$\frac{\partial \mathbf{U}_\mathbf{n}}{\partial t} + \frac{\partial \mathbf{F}_\mathbf{n}(\mathbf{U})}{\partial x_\mathbf{n}} = \mathbf{S}_\mathbf{n}(\mathbf{U}). \tag{6}$$

At this stage, we can introduce the Lagrange-to-Euler mapping $m \longmapsto x_\mathbf{n}(m,t)$ such that

$$\mathrm{d}x_\mathbf{n} = \frac{1}{h}\mathrm{d}m + u_\mathbf{n}\mathrm{d}t$$

is an exact differential and where m denotes the Lagrangian mass coordinate. Hence, it follows

$$\frac{\partial x_\mathbf{n}}{\partial m}(m,t) = \tau, \text{ and } \frac{\partial x_\mathbf{n}}{\partial t}(m,t) = u_\mathbf{n},$$

with $\tau = \dfrac{1}{h}$. It should be noted that we use the same notation for both Lagrangian and Eulerian time. As we know that $x_\mathbf{n} = x_\mathbf{n}(m,t)$, the physical quantities can be expressed in either Lagrangian coordinates (m,t) or Eulerian ones $(x_\mathbf{n}, t)$:

$$\mathbf{U}_\mathbf{n}(m,t) = \mathbf{U}_\mathbf{n}(x_\mathbf{n}(m,t), t).$$

By taking the time derivative of this identity while keeping m fixed, and applying the chain rule, the Lagrangian time derivative is obtained:

$$\frac{\partial \mathbf{U}_\mathbf{n}}{\partial t}(m,t)\Big|_m = \frac{\partial \mathbf{U}_\mathbf{n}}{\partial t}(m,t)\Big|_{x_\mathbf{n}} + u_\mathbf{n}\frac{\partial \mathbf{U}_\mathbf{n}}{\partial x_\mathbf{n}}(x_\mathbf{n},t)\Big|_t.$$

Then, few computations lead to the following identity,

$$h\frac{\partial}{\partial t}(\tau\mathbf{U}_\mathbf{n})(m,t)\Big|_m = \frac{\partial \mathbf{U}_\mathbf{n}}{\partial t}(m,t)\Big|_{x_\mathbf{n}} + \frac{\partial (u_\mathbf{n}\mathbf{U}_\mathbf{n})}{\partial x_\mathbf{n}}(x_\mathbf{n},t)\Big|_t,$$

which, substituted into system (6), leads to

$$\frac{\partial \mathbf{V}_\mathbf{n}}{\partial t} + \frac{\partial \mathbf{G}_\mathbf{n}(\mathbf{V})}{\partial m} = \mathbf{P}_\mathbf{n}(\mathbf{V}_\mathbf{n}).$$

where, in general,

$$\mathbf{V_n} = \tau(\mathbf{U_n} - h\mathbf{e}_1) + \tau\mathbf{e}_1, \text{ where } \mathbf{e}_1 = (1, 0, \ldots, 0)^t,$$
$$\mathbf{G_n} = \mathbf{F_n} - u_\mathbf{n}\mathbf{U_n} - u_\mathbf{n}\mathbf{e}_1.$$

In the particular case of the SW system, we have

$$\mathbf{V_n} = \begin{pmatrix} \tau \\ u_\mathbf{n} \\ u_\mathbf{t} \end{pmatrix}, \qquad \mathbf{G_n} = \begin{pmatrix} -u_\mathbf{n} \\ p \\ 0 \end{pmatrix}, \qquad \mathbf{P_n} = \begin{pmatrix} 0 \\ -gh\partial_m B \\ 0 \end{pmatrix}.$$

Refer also to [33, 19] for the 1D shallow water system in Lagrangian coordinates. Neglecting the source term, the Lagrangian system is hyperbolic and admits three eigenvalues $-\frac{a}{\tau}$, 0 and $\frac{a}{\tau}$, which are automatically ordered as $a > 0$. We also remark that the formulation of the SW equations in Lagrangian coordinates is simpler than the Eulerian one, as the advection part (material waves) of the system is no longer present.

Finally, the selection of physically admissible weak solutions is ensured by supplementing this system of conservation laws with the entropy inequality:

$$\frac{\partial}{\partial t}E_\mathbf{n} + \frac{\partial}{\partial m}(pu_\mathbf{n}) \leq -ghu_\mathbf{n}\,\partial_m B.$$

We can reformulate this inequality in a more compact form as

$$\frac{\partial}{\partial t}(\sigma_\mathbf{n}) + \frac{\partial}{\partial m}\theta_\mathbf{n} \leq \phi_\mathbf{n},$$

where

$$\sigma_\mathbf{n} = E, \qquad \theta_\mathbf{n} = pu_\mathbf{n}, \qquad \phi_\mathbf{n} = -ghu_\mathbf{n}\,\partial_m B.$$

3. 2D finite volume scheme: Riemann solver

We now aim to design the two-dimensional finite volume scheme for solving the SW system (2)-(3). The computational domain is a polygonal portion of $\mathbb{R}^2$, paved with a set of non overlapping polygonal cells ω_c. c is the generic label of the cell. Let $\mathcal{P}(c)$ be the set of vertices (points) of ω_c. The generic label of a point is p and $\mathbf{x}_p$ denotes its vector position. The points of cell ω_c are counter-clockwise ordered, and p^- and p^+ are respectively the previous and the next points with respect to p, refer to figure 2. The subcell ω_{pc} is the quadrangle formed by joining the cell centroid, $\mathbf{x}_c$, to the midpoints of $[\mathbf{x}_{p^-}, \mathbf{x}_p]$, $[\mathbf{x}_p, \mathbf{x}_{p^+}]$ and to $\mathbf{x}_p$. The set of subcells ω_{pc} for $p \in \mathcal{P}(c)$ constitutes a partition of the cell ω_c, that is,

$$\omega_c = \bigcup_{p\in\mathcal{P}(c)} \omega_{pc}.$$

We also introduce the set of faces of cell ω_c and denote it $\mathcal{F}(c)$. Each face f of cell c is decomposed into subfaces by means of the partition of c induced by the subcells pc for $p \in \mathcal{P}(c)$. As such we define $\mathcal{SF}(pc)$ the set of subfaces attached to the corner pc, which is nothing but the set of faces of subcell ω_{pc} impinging at point p. We denote by l_{pcf} and $\mathbf{n}_{pcf} = (n_x, n_y)_{pcf}$ the measure and the unit outward normal of the subface f respectively. We observe that the set of subfaces $\mathcal{SF}(pc)$ for $p \in \mathcal{P}(c)$ constitutes a partition of the set of faces of ω_c, that is,

$$\mathcal{F}(c) = \bigcup_{p\in\mathcal{P}(c)} \mathcal{SF}(pc).$$

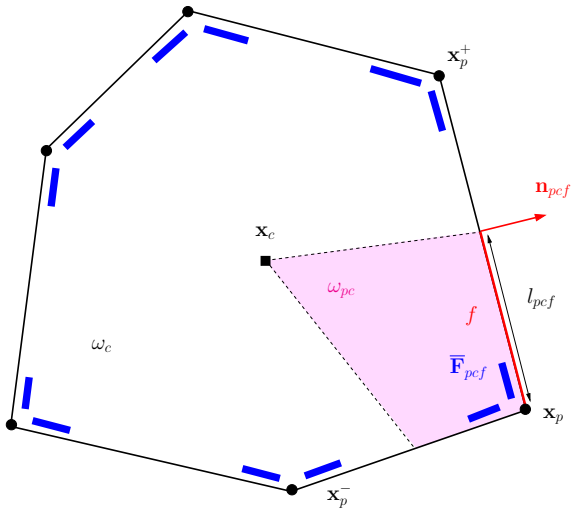


Figure 2: Geometrical entities attached to the polygonal cell ω_c.

Given a cell c and one of its face f, a unique 'neighbor' cell can be associated and we refer to it as $d(c,f)$ or d to shorten the notation. Obviously c is the neighbor cell of d through edge f as well. The set of neighbors of cell c is denoted by $\mathcal{N}(c)$. Shortly, we will define it more appropriately, either face-based or node-based.

Let us recall the generic subface-based finite volume discretization. We integrate the system of balance laws (1) over ω_c and employ the Green formula, obtaining

$$|\omega_c|\frac{\mathrm{d}\mathbf{U}_c}{\mathrm{d}t} + \int_{\partial\omega_c} \mathbb{F}(\mathbf{U})\mathbf{n}\,\mathrm{d}s = \int_{\omega_c} \mathbf{S}(\mathbf{U})\,\mathrm{d}v, \tag{7}$$

where $\mathbf{U}_c(t) = \dfrac{1}{|\omega_c|}\displaystyle\int_{\omega_c} \mathbf{U}(\mathbf{x},t)\,\mathrm{d}v$ is the cell-averaged value of $\mathbf{U}$ over ω_c and $\mathbf{U}_c^n \equiv \mathbf{U}_c(t^n)$ for any discrete time t^n. A classical first-order explicit time integration turns (7) into

$$\mathbf{U}_c^{n+1} - \mathbf{U}_c^n + \frac{\Delta t}{|\omega_c|}\int_{\partial\omega_c} \mathbb{F}(\mathbf{U}^n)\mathbf{n}\,\mathrm{d}s = \frac{\Delta t}{|\omega_c|}\int_{\omega_c} \mathbf{S}(\mathbf{U}^n)\,\mathrm{d}v. \tag{8}$$

At this stage, we need to define an approximation of the normal flux and source term integrals. Following [36], we design an original node-based approximation of theses integral terms relying on the partition of ω_c into subcells. In particular, we state

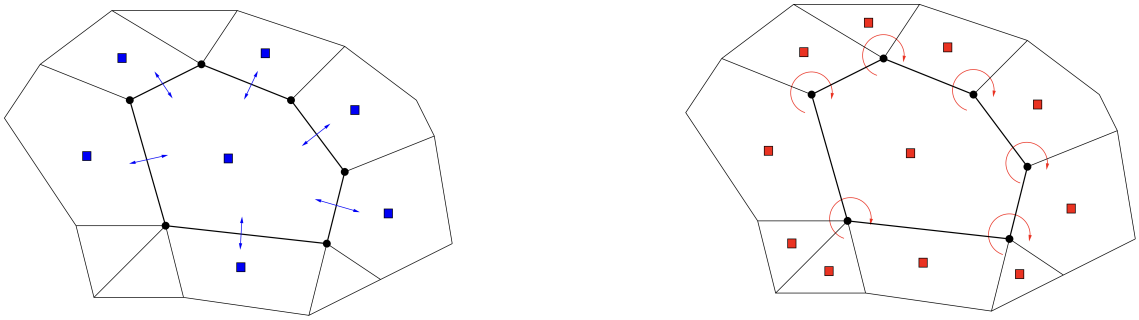
$$\int_{\partial\omega_c} \mathbb{F}(\mathbf{U}^n)\mathbf{n}\,\mathrm{d}s = \sum_{p\in\mathcal{P}(c)} \int_{\partial\omega_{pc}\cap\partial\omega_c} \mathbb{F}(\mathbf{U}^n)\mathbf{n}\,\mathrm{d}s,$$

and we approximate the surface integral term in the right-hand side along the subfaces by means of subface fluxes related to the subface f attached to the corner pc, denoted $\overline{\mathbf{F}}_{pcf}$. Namely, we impose

$$\int_{\partial\omega_{pc}\cap\partial\omega_c} \mathbb{F}(\mathbf{U}^n)\mathbf{n}\,\mathrm{d}s = \sum_{f\in\mathcal{SF}(pc)} l_{pcf}\overline{\mathbf{F}}_{pcf}.$$

We assume that we can similarly discretize the source term using subface sources,

$$\int_{\omega_c} \mathbf{S}(\mathbf{U}^n)\,\mathrm{d}v \approx \sum_{p\in\mathcal{P}(c)}\sum_{f\in\mathcal{SF}(pc)} w_{pcf}\overline{\mathbf{S}}_{pcf},$$



(a) Face-based stencil of the classical two-point flux.

(b) Node-based stencil of the unclassical multi-point flux.

Figure 3: Images taken from [36].

where we used a quadrature formula with w_{pcf} weights that will be later specified. By substituting the above formulas into (8), we obtain the subface-based generic finite volume scheme:

$$\begin{aligned}\mathbf{U}_c^{n+1} &= \mathbf{U}_c^n - \frac{\Delta t}{|\omega_c|} \sum_{p\in\mathcal{P}(c)} \sum_{f\in\mathcal{SF}(pc)} \left[l_{pcf}\overline{\mathbf{F}}_{pcf} - w_{pcf}\overline{\mathbf{S}}_{pcf} \right] \\ &= \mathbf{U}_c^n - \frac{\Delta t}{|\omega_c|} \sum_{p\in\mathcal{P}(c)} \sum_{f\in\mathcal{SF}(pc)} l_{pcf}\widetilde{\mathbf{F}}_{pcf},\end{aligned} \tag{9}$$

where

$$\widetilde{\mathbf{F}}_{pcf} = \overline{\mathbf{F}}_{pcf} - \frac{w_{pcf}}{l_{pcf}}\overline{\mathbf{S}}_{pcf}.$$

Thus, the contribution of both flux and source terms is included in $\widetilde{\mathbf{F}}_{pcf}$, for which we state

$$\widetilde{\mathbf{F}}_{pcf} = \widetilde{\mathbf{F}}_{pcf}(\mathbf{U}_c^n, \mathbf{U}_d^n, B_c, B_d, \mathbf{n}_{pcf}, \mathbf{u}_p),$$

where $d \equiv d(c,f)$ and $\mathbf{u}_p$ is a nodal parameter we will later discuss. Specifically, we will see that $\mathbf{u}_p$ approximates the nodal velocity at the node p, which is commonly used in the Lagrangian framework to move the mesh in a compatible manner. In this work, we will exploit this notion and include it in our Eulerian FV scheme. The objective is to consider the entire neighborhood of a given cell c when updating its state value, including information coming from cells that only share one vertex with c, see also figure 3. Hence, the numerical flux $\widetilde{\mathbf{F}}_{pcf}$ will not only depend on the two (left and right) states attached to a given face f as for a classical "two-point" scheme. Indeed, thanks to the presence of $\mathbf{u}_p$, the contribution of all (multiple) cells around a given node p will be used to compute $\widetilde{\mathbf{F}}_{pcf}$, hence the name "multi-point" method.

3.1. Riemann problem in the normal direction

In this section, we introduce the concept of Riemann problem and solver in the normal direction to a given cell interface, which will be used to define the fluxes $\widetilde{\mathbf{F}}_{pcf}$. In particular, in this subsection and the following one, we provide the description of the Riemann problem, the associated consistency conditions and flux definition. Then, from subsection 3.3, we discuss the solution of the Riemann problem for the SW system, first in Lagrangian coordinates and, then, in Eulerian coordinates. Specifically, we will also see how to include the nodal parameter $\mathbf{u}_p$ in the

approximate Riemann solver.

3.1.1. The Eulerian case

Consider a given cell interface f impinging at a point p, and $\mathbf{U}_{lf}$, $\mathbf{U}_{rf}$ the left and right state values to f. Then, we define $\mathbf{n}_{pf}$ as the normal vector to the face f and pointing towards the right state. With this formalism, the Riemann problem in the normal direction $\mathbf{n}_{pf}$ reads

$$(\mathcal{RP}) : \begin{cases} \dfrac{\partial \mathbf{U}_{\mathbf{n}_{pf}}}{\partial t} + \dfrac{\partial [\mathbf{F}_{\mathbf{n}_{pf}}(\mathbf{U})]}{\partial x_{\mathbf{n}_{pf}}} = \mathbf{S}_{\mathbf{n}_{pf}}, \\ \mathbf{U}_{\mathbf{n}_{pf}}(x_{\mathbf{n}_{pf}}, 0) = \begin{cases} \mathbf{U}_{lf} \text{ if } x_{\mathbf{n}_{pf}} < 0, \\ \mathbf{U}_{rf} \text{ if } x_{\mathbf{n}_{pf}} \geq 0. \end{cases} \end{cases}$$

Observe that, for the sake of clarity, we simplified the notation regarding the considered cell c. Furthermore, for the vector variables $\mathbf{U}$ (and $\mathbf{V}$ later), we now drop the notation regarding the normal direction $\mathbf{n}_{pf}$.

It is then clear that the resulting one-dimensional approximate Riemann solver depends on $\mathbf{U}_{lf}$, $\mathbf{U}_{rf}$, but also on the self-similar variable $\xi = \dfrac{x_{\mathbf{n}_{pf}}}{t}$ and on the nodal parameter $\mathbf{u}_p$. With these arguments, the Riemann solver reads

$$\mathbf{W}_{pf,\mathcal{E}} = \mathbf{W}_{pf,\mathcal{E}}(\mathbf{U}_{lf}, \mathbf{U}_{rf}, B_{lf}, B_{rf}, \mathbf{n}_{pf}, \xi, \mathbf{u}_p),$$

where $\mathcal{E}$ stands for Eulerian. Then, for all $\mathbf{U}_{lf}$, $\mathbf{U}_{rf}$, B_{lf}, B_{rf}, $\mathbf{n}_{pf}$, ξ and $\mathbf{u}_p$, we assume the Riemann solver to satisfy the following classical properties:

- $\mathbf{W}_{pf,\mathcal{E}}(\mathbf{U}_{lf}, \mathbf{U}_{rf}, B_{lf}, B_{rf}, \mathbf{n}_{pf}, \xi, \mathbf{u}_p) = \mathbf{U}_{lf}$ for $-\xi$ large enough;
- $\mathbf{W}_{pf,\mathcal{E}}(\mathbf{U}_{lf}, \mathbf{U}_{rf}, B_{lf}, B_{rf}, \mathbf{n}_{pf}, \xi, \mathbf{u}_p) = \mathbf{U}_{rf}$ for ξ large enough;
- $\mathbf{W}_{pf,\mathcal{E}}(\mathbf{U}_{lf}, \mathbf{U}_{rf}, B_{lf}, B_{rf}, \mathbf{n}_{pf}, \xi, \mathbf{u}_p) = \mathbf{U}$.

Finally, $\mathbf{W}_{pf,\mathcal{E}}$ is assumed to be symmetric with respect to the cell interface.

However, as the exact solution of the Riemann problem is generally not known, in this work we are concerned with simple (approximate) Riemann solvers. A simple Riemann solver is constituted of $J+1$ constant states separated by J waves of speeds Λ_k, $k = 1, \ldots, J$:

$$\mathbf{W}_{pf,\mathcal{E}}(\mathbf{U}_{lf}, \mathbf{U}_{rf}, B_{lf}, B_{rf}, \mathbf{n}_{pf}, \xi, \mathbf{u}_p) = \begin{cases} \mathbf{U}_1 = \mathbf{U}_{lf} & \text{if} \quad \xi < \Lambda_1, \\ \vdots & \\ \mathbf{U}_k & \text{if} \quad \Lambda_{k-1} < \xi \leq \Lambda_k, \ k = 2, \ldots, J, \\ \vdots & \\ \mathbf{U}_{J+1} = \mathbf{U}_{rf} & \text{if} \quad \Lambda_J \leq \xi. \end{cases} \tag{10}$$

Given the space and time steps Δx_{lf}, Δx_{rf} and Δt, the approximate solution (10) of $(\mathcal{RP})$ is classically defined as being consistent with the integral form of $(\mathcal{RP})$ if the integration of $(\mathcal{RP})$ in the interval $[-\Delta x_{lf}, \Delta x_{rf}] \times [0, \Delta t]$ leads to

$$\begin{aligned} &- \left[\sum_{k=1}^{J} \Lambda_k \left(\mathbf{U}_{k+1} - \mathbf{U}_k \right) \right]_{lf,rf} + \left(\mathbb{F}(\mathbf{U}_{rf}) - \mathbb{F}(\mathbf{U}_{lf}) \right) \mathbf{n}_{pf} \\ &- (\Delta x_{lf} + \Delta x_{rf}) \mathbf{S}_{\mathbf{n}_{pf}}(\Delta x_{lf}, \Delta x_{rf}, \mathbf{U}_{lf}, \mathbf{U}_{Lrf}, B_{lf}, B_{rf}) = \mathbf{0}, \end{aligned} \tag{11}$$

see [34] for more details. However, for the multi-point scheme, we do not require the classical consistency conditions across each cell interface. Since we aim to take into account the entire neighborhood of a cell, we propose to impose that the sum of the consistency conditions across each face around the given node should be zero. This can also be seen as an approximation of the integral of $(\mathcal{RP})$ around the node (integral over $\bigcup_{c\in\mathcal{C}(p)} \omega_{pc} \times [0, \Delta t]$). Hence, we ask for the following generalized consistency conditions around the node p:

$$\sum_{f\in\mathcal{SF}(p)} l_{pf}\Bigg[-\left[\sum_{k=1}^{J} \Lambda_k \left(\mathbf{U}_{k+1} - \mathbf{U}_k\right)\right]_{lf,rf} + \left(\mathbb{F}(\mathbf{U}_{rf}) - \mathbb{F}(\mathbf{U}_{lf})\right)\mathbf{n}_{pf} \\ - (\Delta x_{lf} + \Delta x_{rf})\mathbf{S}_{\mathbf{n}_{pf}}(\Delta x_{lf}, \Delta x_{rf}, \mathbf{U}_{lf}, \mathbf{U}_{rf}, B_{lf}, B_{rf})\Bigg] = \mathbf{0}, \tag{12}$$

where $\mathcal{SF}(p)$ denotes the set of subfaces impinging at point p and, for $f \in \mathcal{SF}(p)$, l_{pf} is the length of subface f. In the rest of the paper, we will distinguish between classical (11) and generalized (12) consistency conditions, referring to them as *face-based* and *node-based* consistency conditions respectively.

We highlight that each node p must fulfill its own local conditions, and (12) will de facto provide a way to determine the nodal parameter $\mathbf{u}_{pf}$. Appropriate considerations of this choice will be given in the following sections. In particular, we will see that fundamental properties as the conservation one are still verified in spite of not using the classical consistency conditions (11). Namely, we will show that conditions (12) are sufficient to prove the conservation property.

To conclude the discussion about the Eulerian Riemann problem, we emphasise that for the entropy we invoke the same argument to that proposed for the consistency conditions. Namely, we do not simply impose the following inequality (see also [34]),

$$-\left[\sum_{k=1}^{h} \Lambda_k \left(\Sigma_{k+1} - \Sigma_k\right)\right]_{lf,rf} + \left(\Theta(\mathbf{U}_{rf}) - \Theta(\mathbf{U}_{lf})\right)\mathbf{n}_{pf} \\ - (\Delta x_{lf} + \Delta x_{rf})\Phi(\Delta x_{lf}, \Delta x_{rf}, \mathbf{U}_{lf}, \mathbf{U}_{rf}, B_{lf}, B_{rf})\mathbf{n}_{pf} \leq \mathbf{0},$$

but we ask for the more general node-based version around the node p,

$$\sum_{f\in\mathcal{SF}(p)} l_{pf}\Bigg[-\left[\sum_{k=1}^{J} \Lambda_k \left(\Sigma_{k+1} - \Sigma_k\right)\right]_{lf,rf} + \left(\Theta(\mathbf{U}_{rf}) - \Theta(\mathbf{U}_{lf})\right)\mathbf{n}_{pf} \\ - (\Delta x_{lf} + \Delta x_{rf})\Phi_{\mathbf{n}_{pf}}(\Delta x_{lf}, \Delta x_{rf}, \mathbf{U}_{lf}, \mathbf{U}_{rf}, B_{lf}, B_{rf})\Bigg] \leq \mathbf{0}. \tag{13}$$

We will see that this is sufficient to ensure the decrease of the total entropy in the entire domain. In particular, further explanations about the entropy stability will be given in sections 3.5.3 and 4.4.

3.1.2. The Lagrangian case

Let us now apply the same reasoning to the Lagrangian counterpart of the equations. We recall that we also present the problem step by step in Lagrangian coordinates because they will be useful for defining the solution of the (Eulerian) Riemann problem and thus the numerical flux. Indeed, this is due both to the fact that the idea of the nodal parameter comes from the Lagrangian framework (nodal velocity used to move the mesh in a compatible manner), and also because the formulation of the 2D SW equations in Lagrangian coordinates is easier to manipulate.

Hence, thanks to the Lagrange-to-Euler mapping we introduced in the previous section, we can write the Lagrangian Riemann problem [34]:

$$(\mathcal{RP}_{\mathcal{L}}): \begin{cases} \dfrac{\partial \mathbf{V}_{\mathbf{n}_{pf}}}{\partial t} + \dfrac{\partial [\mathbf{G}_{\mathbf{n}_{pf}}(\mathbf{V})]}{\partial m_{\mathbf{n}_{pf}}} = \mathbf{P}_{\mathbf{n}_{pf}}, \\ \mathbf{V}_{\mathbf{n}_{pf}}(m_{\mathbf{n}_{pf}}, 0) = \begin{cases} \mathbf{V}_{lf} \text{ if } m_{\mathbf{n}_{pf}} < 0, \\ \mathbf{V}_{rf} \text{ if } m_{\mathbf{n}_{pf}} \geq 0. \end{cases} \end{cases}$$

Assumptions analogous to those made for the Eulerian solver also apply to the Lagrangian solver $\mathbf{W}_{pf,\mathcal{L}} = \mathbf{W}_{pf,\mathcal{L}}(\mathbf{V}_{lf}, \mathbf{V}_{rf}, B_{lf}, B_{rf}, \mathbf{n}_{pf}, \xi_{\mathcal{L}}, \mathbf{u}_p)$ with $\xi_{\mathcal{L}} = \frac{m}{t}$ and where $\mathcal{L}$ stands for Lagrangian. Then, once again, the approximate solution is given by piece-wise constant data separated by J discontinuities with speeds λ_k, $k = 1, \ldots, J$,

$$\mathbf{W}_{pf,\mathcal{L}}(\mathbf{V}_{lf}, \mathbf{V}_{rf}, B_{lf}, B_{rf}, \mathbf{n}_{pf}, \xi_L, \mathbf{u}_p) = \begin{cases} \mathbf{V}_1 = \mathbf{V}_{lf} & \text{if} \quad \xi_{\mathcal{L}} < \lambda_1, \\ \vdots & \\ \mathbf{V}_k & \text{if} \quad \lambda_{k-1} < \xi_{\mathcal{L}} \leq \lambda_k, \ k = 2, \ldots, J, \\ \vdots & \\ \mathbf{V}_{J+1} = \mathbf{V}_{rf} & \text{if} \quad \lambda_J \leq \xi_{\mathcal{L}}. \end{cases} \tag{14}$$

As for the Eulerian simple Riemann solver, for a two-point scheme, we would impose the classical face-based Lagrangian consistency condition in the interval $[-\Delta m_{lf}, \Delta m_{rf}] \times [0, \Delta t]$,

$$\begin{aligned} &- \left[\sum_{k=1}^{J} \lambda_k \left(\mathbf{V}_{k+1} - \mathbf{V}_k \right) \right]_{lf,rf} + \left(\mathbb{G}(\mathbf{V}_{rf}) - \mathbb{G}(\mathbf{V}_{lf}) \right) \mathbf{n}_{pf} \\ &- (\Delta m_{lf} + \Delta m_{rf}) \mathbf{P}_{\mathbf{n}_{pf}}(\Delta m_{lf}, \Delta m_{rf}, \mathbf{V}_{lf}, \mathbf{V}_{rf}, B_{lf}, B_{rf}) = \mathbf{0}, \end{aligned} \tag{15}$$

with space steps Δm_{lf}, Δm_{rf} and $\mathbb{G} = \mathbb{G}(\mathbf{V})$ the Lagrangian flux tensor in $\mathbb{R}^3 \times \mathbb{R}^2$. Once again, for the multi-point scheme, we only ask for the generalized node-based version of these conditions, namely

$$\begin{aligned} \sum_{f \in \mathcal{SF}(p)} l_{pf} \Bigg[&- \left[\sum_{k=1}^{J} \lambda_k \left(\mathbf{V}_{k+1} - \mathbf{V}_k \right) \right]_{lf,rf} + \left(\mathbb{G}(\mathbf{V}_{rf}) - \mathbb{G}(\mathbf{V}_{lf}) \right) \mathbf{n}_{pf} \\ &- (\Delta m_{lf} + \Delta m_{rf}) \mathbf{P}_{\mathbf{n}_{pf}}(\Delta m_{lf}, \Delta m_{rf}, \mathbf{V}_{lf}, \mathbf{V}_{rf}, B_{lf}, B_{rf}) \Bigg] = \mathbf{0}. \end{aligned} \tag{16}$$

Finally, to ensure the entropy decrease, the Lagrangian Riemann solver will satisfy the following node-based conditions:

$$\begin{aligned} \sum_{f \in \mathcal{SF}(p)} l_{pf} \Bigg[&- \left[\sum_{k=1}^{J} \lambda_k \left(\sigma_{k+1} - \sigma_k \right) \right]_{lf,rf} + \left(\theta(\mathbf{V}_{rf}) - \theta(\mathbf{V}_{lf}) \right) \mathbf{n}_{pf} \\ &- (\Delta m_{lf} + \Delta m_{rf}) \phi_{\mathbf{n}_{pf}}(\Delta m_{lf}, \Delta m_{rf}, \mathbf{V}_{lf}, \mathbf{V}_{rf}, B_{lf}, B_{rf}) \Bigg] \leq \mathbf{0}. \end{aligned} \tag{17}$$

3.2. Subface flux approximation by means of a Riemann solver

In this section, we exploit the Riemann problems and solvers we previously described to propose a possible approximation for the subface fluxes.

3.2.1. The Eulerian case

Given the space and time increments, Δx_{lf} and Δt respectively, we integrate the Eulerian Riemann problem ($\mathcal{RP}$) over the space-time domain $[-\Delta x_{lf}, 0] \times [0, \Delta t]$ to obtain the left-sided subface flux $\widetilde{\mathbf{F}}^{-}_{pf} = \widetilde{\mathbf{F}}^{-}_{pf}(\mathbf{U}_{lf}, \mathbf{U}_{rf}, B_{lf}, B_{rf}, \mathbf{n}_{pf}, \mathbf{u}_p)$:

$$\widetilde{\mathbf{F}}_{pf} \equiv \widetilde{\mathbf{F}}^{-}_{pf} = \mathbb{F}(\mathbf{U}_{lf})\mathbf{n}_{pf} - \int_{-\infty}^{0} [\mathbf{W}_{pf,\mathcal{E}}(\mathbf{U}_{lf}, \mathbf{U}_{rf}, B_{lf}, B_{rf}, \mathbf{n}_{pf}, \xi, \mathbf{u}_p) - \mathbf{U}_{lf}] \, \mathrm{d}\xi, \tag{18}$$

or, equivalently, $\overline{\mathbf{F}}^{-}_{pf} = \widetilde{\mathbf{F}}^{-}_{pf} + \Delta x_L \overline{\mathbf{S}}_{\mathbf{n}_{pf}}$. A similar formula for the right-sided subface flux $\widetilde{\mathbf{F}}_{pf} \equiv \widetilde{\mathbf{F}}^{+}_{pf} = \widetilde{\mathbf{F}}^{+}_{pf}(\mathbf{U}_{lf}, \mathbf{U}_{rf}, B_{lf}, B_{rf}, -\mathbf{n}_{pf}, \mathbf{u}_p)$ can be found by integrating over $[0, \Delta x_{rf}] \times [0, \Delta t]$:

$$\widetilde{\mathbf{F}}^{+}_{pf} = \mathbb{F}(\mathbf{U}_{rf})(-\mathbf{n}_{pf}) + \int_{0}^{+\infty} [\mathbf{W}_{pf,\mathcal{E}}(\mathbf{U}_{rf}, \mathbf{U}_{lf}, B_{rf}, B_{lf}, \mathbf{n}_{pf}, \xi, \mathbf{u}_p) - \mathbf{U}_{rf}] \, \mathrm{d}\xi. \tag{19}$$

However and for the sake of clarity, remark that in the finite volume formula (9), we will choose to use the left-sided flux, namely $\widetilde{\mathbf{F}}_{pf} \equiv \widetilde{\mathbf{F}}^{-}_{pf}$.

Then, thanks to the simple Riemann solver (10) we defined in the previous section, we are able to provide the following approximations for the subface fluxes (18) and (19),

$$\widetilde{\mathbf{F}}^{-}_{pf} = \mathbf{F}_{\mathbf{n}_{pf}}(\mathbf{U}_{lf}) - \left[\sum_{k=1}^{J} \Lambda_k^{(-)} (\mathbf{U}_{k+1} - \mathbf{U}_k) \right]_{lf,rf}, \tag{20}$$

and

$$\widetilde{\mathbf{F}}^{+}_{pf} = \mathbf{F}_{\mathbf{n}_{pf}}(\mathbf{U}_{rf}) - \left[\sum_{k=1}^{J} \Lambda_k^{(+)} (\mathbf{U}_{k+1} - \mathbf{U}_k) \right]_{lf,rf}, \tag{21}$$

where $\Lambda_k^{(-)} = \frac{1}{2}(|\Lambda_k| - \Lambda_k)$ and $\Lambda_k^{(+)} = \frac{1}{2}(|\Lambda_k| + \Lambda_k)$. At this stage, in a two-point flux-splitting method, in order to find the consistency conditions, we would integrate over $[-\Delta x_{lf}, \Delta x_{rf}] \times [0, \Delta t]$, obtaining formula (15). This would be equivalent to impose the following face-based jump condition across 0,

$$\widetilde{\mathbf{F}}^{+}_{pf} - \widetilde{\mathbf{F}}^{-}_{pf} = (\Delta x_{lf} + \Delta x_{rf}) \mathbf{S}_{\mathbf{n}_{pf}}(\Delta x_{lf}, \Delta x_{rf}, \mathbf{U}_{lf}, \mathbf{U}_{rf}, B_{lf}, B_{rf}).$$

However, as we previously said, we will not use the face-based consistency conditions (15) but the more general node-based ones (16), which would be equivalent to impose a general node-based jump condition:

$$\sum_{f \in \mathcal{SF}(p)} l_{pf} \left[\widetilde{\mathbf{F}}^{+}_{pf} - \widetilde{\mathbf{F}}^{-}_{pf} \right] = \sum_{f \in \mathcal{SF}(p)} l_{pf} (\Delta x_{lf} + \Delta x_{rf}) \mathbf{S}_{\mathbf{n}_{pf}}(\Delta x_{lf}, \Delta x_{rf}, \mathbf{U}_{lf}, \mathbf{U}_{rf}, B_{lf}, B_{rf}). \tag{22}$$

Namely, we are imposing that the sum of the difference of the fluxes over the subfaces around the node p is compensated by the sum of the source term approximations across each interface impinging on the node p.

3.2.2. The Lagrangian case

Finally, once again we provide the subface fluxes definitions for the Lagrangian counterpart of the equations. The left-sided Lagrangian subface flux is given by

$$\widetilde{\mathbf{G}}_{pf} \equiv \widetilde{\mathbf{G}}^{-}_{pf} = \mathbb{G}(\mathbf{V}_{lf})\mathbf{n}_{pf} - \int_{-\infty}^{0} [\mathbf{W}_{pf,\mathcal{L}}(\mathbf{V}_{lf}, \mathbf{V}_{rf}, B_{lf}, B_{rf}, \mathbf{n}_{pf}, \xi_{\mathcal{L}}, \mathbf{u}_p) - \mathbf{V}_{lf}] \, \mathrm{d}\xi_{\mathcal{L}},$$

and using the simple Riemann solver (14) we get,

$$\widetilde{\mathbf{G}}^{-}_{pf} = \mathbf{G}_{\mathbf{n}_{pf}}(\mathbf{V}_{lf}) - \left[\sum_{k=1}^{J} \lambda_k^{(-)}(\mathbf{V}_{k+1} - \mathbf{U}_k)\right]_{lf,rf},$$

and

$$\widetilde{\mathbf{G}}^{+}_{pf} = \mathbf{G}_{\mathbf{n}_{pf}}(\mathbf{V}_{rf}) - \left[\sum_{k=1}^{J} \lambda_k^{(+)}(\mathbf{V}_{k+1} - \mathbf{U}_k)\right]_{lf,rf}.$$

Once again, the generalized node-based consistency conditions (16) would be equivalent to impose

$$\sum_{f\in\mathcal{SF}(p)} l_{pf}\left[\widetilde{\mathbf{G}}^{+}_{pf} - \widetilde{\mathbf{G}}^{-}_{pf}\right] = \sum_{f\in\mathcal{SF}(p)} l_{pf}(\Delta m_{lf} + \Delta m_{rf})\mathbf{P}_{\mathbf{n}_{pf}}(\Delta m_{lf}, \Delta m_{rf}, \mathbf{V}_{lf}, \mathbf{V}_{rf}, B_{lf}, B_{rf}). \tag{23}$$

To conclude this section, we highlight that the difference between the right and left-sided Eulerian fluxes coincides with the difference between the right and the left-sided Lagrangian fluxes:

$$\widetilde{\mathbf{F}}^{+}_{pf} - \widetilde{\mathbf{F}}^{-}_{pf} = \widetilde{\mathbf{G}}^{+}_{pf} - \widetilde{\mathbf{G}}^{-}_{pf}.$$

We do not show here all the computations for such a formula but we simply refer to [36] for it. Hence, the generalized node-based consistency conditions (12) (or equivalently (22)) in Eulerian coordinates revealed themselves to be equivalent to the ones in Lagrangian coordinates: either (16) or (23).

3.3. Lagrangian simple Riemann solver for SW equations

Let us now enter the details of the simple Riemann solver we use for the SW system. We make the restrictive assumption to consider the left and right waves as shock discontinuities: this allows us to apply the jump conditions across each wave. We start by the Lagrangian version of the solver, as the simpler formulation of the equations and the ordered eigenvalues make the discussion easier. Moreover, here we drop the subscript pf to lighten the notations.

As the system admits three distinct eigenvalues, the Lagrangian solver (14) is naturally composed of four states $\mathbf{V}_l$, $\mathbf{V}_l^\star$, $\mathbf{V}_r^\star$ and $\mathbf{V}_r$, respectively separated by discontinuities of speeds $-\lambda_l$, 0 and λ_r in the (m,t) plane:

$$\mathbf{W}_{\mathcal{L}}\left(\mathbf{V}_l, \mathbf{V}_r, \frac{m}{t}\right) = \begin{cases} \mathbf{V}_l & \text{if} \quad \frac{m}{t} \le -\lambda_l, \\ \mathbf{V}_l^\star & \text{if} \quad -\lambda_l < \frac{m}{t} \le 0, \\ \mathbf{V}_r^\star & \text{if} \quad 0 < \frac{m}{t} \le \lambda_r, \\ \mathbf{V}_r & \text{if} \quad \lambda_r < \frac{m}{t}. \end{cases}$$

λ_l and λ_r are positive real parameters that should respectively approximate $\dfrac{a_l}{\tau_l}$ and $\dfrac{a_r}{\tau_r}$. Then, λ_l and λ_r are also further constrained to ensure the positivity and entropy stability properties of the Riemann solver following the methodology introduced in [32, 34, 33, 35] and revised in [15, 36]. We will discuss them in sections 3.5.2 and 3.5.3. Observe that the three waves $-\lambda_l, 0, \lambda_r$ are automatically ordered, hence no ambiguity arises. The left and right state components read $\mathbf{V}_S = (\tau_S, u_{\mathbf{n},S}, u_{\mathbf{t},S})^t$ and the intermediate states components are $\mathbf{V}_S^\star = (\tau_S^\star, u_{\mathbf{n},S}^\star, u_{\mathbf{t},S}^\star)^t$ for $S = l, r$. The jump condition for τ across the second wave (of null velocity) yields

$$0(\tau_r^\star - \tau_l^\star) - (u_{\mathbf{n},r}^\star - u_{\mathbf{n},l}^\star) = 0 \quad \text{and thus} \quad u_{\mathbf{n},r}^\star = u_{\mathbf{n},l}^\star \equiv u_{\mathbf{n}}^\star.$$

Therefore, the jump conditions for τ boils down to

$$\lambda_l(\tau_l^\star - \tau_l) - (u_{\mathbf{n}}^\star - u_{\mathbf{n},l}) = 0, \tag{24a}$$
$$\lambda_r(\tau_r^\star - \tau_r) + (u_{\mathbf{n}}^\star - u_{\mathbf{n},r}) = 0. \tag{24b}$$

Now, following [15, 36], we complete the Lagrangian Riemann solver characterization introducing the intermediate fluxes for $S = l, r$

$$\widetilde{\mathbf{G}}_{\mathbf{n},S} = (-\widetilde{u}_{\mathbf{n},S},\ \widetilde{p}_S,\ 0)^t,$$

which satisfy the system

$$\lambda_l(\mathbf{V}_l^\star - \mathbf{V}_l) + \widetilde{\mathbf{G}}_{\mathbf{n},l} - \mathbf{G}_{\mathbf{n},l} = 0, \tag{25a}$$
$$-\lambda_r(\mathbf{V}_r - \mathbf{V}_r^\star) + \mathbf{G}_{\mathbf{n},r} - \widetilde{\mathbf{G}}_{\mathbf{n},r} = 0, \tag{25b}$$

where $\mathbf{G}_{\mathbf{n},S} = \mathbf{G}_{\mathbf{n}}(\mathbf{V}_S)$. Combining the first components of (25a), (25b) with (24a), (24b) leads to $\widetilde{u}_{\mathbf{n},l} = u_{\mathbf{n}}^\star = \widetilde{u}_{\mathbf{n},r}$. We arrive at the following expressions of the intermediate states and fluxes for $S = l, r$,

$$\mathbf{V}_S^\star = \begin{pmatrix} \tau_S^\star \\ u_{\mathbf{n}}^\star \\ u_{\mathbf{t},S}^\star \end{pmatrix}, \qquad \widetilde{\mathbf{G}}_{\mathbf{n},S} = \begin{pmatrix} -u_{\mathbf{n}}^\star \\ \widetilde{p}_S \\ 0 \end{pmatrix}.$$

As such the system of 6 equations for 7 unknowns is given by (25a) and (25b), that is

$$(\mathcal{S}_l)\begin{cases} \lambda_l(\tau_l^\star - \tau_l) - (u_{\mathbf{n}}^\star - u_{\mathbf{n},l}) = 0, \\ \lambda_l(u_{\mathbf{n}}^\star - u_{\mathbf{n},l}) + \widetilde{p}_l - p_l = 0, \\ \lambda_l(u_{\mathbf{t},l}^\star - u_{\mathbf{t},l}) = 0, \end{cases} \qquad (\mathcal{S}_r)\begin{cases} \lambda_r(\tau_r^\star - \tau_r) + u_{\mathbf{n}}^\star - u_{\mathbf{n},r} = 0, \\ \lambda_r(u_{\mathbf{n}}^\star - u_{\mathbf{n},r}) - (\widetilde{p}_r - p_r) = 0, \\ \lambda_r(u_{\mathbf{t},r}^\star - u_{\mathbf{t},r}) = 0. \end{cases}$$

The left and right tangential velocities are simply given by the third equations: $u_{\mathbf{t},S}^\star = u_{\mathbf{t},S}$ as $\lambda_S \neq 0$ for $S = l, r$. Next, the intermediate normal velocity $u_{\mathbf{n}}^\star$ is viewed as a parameter to express the 4 remaining unknowns, $\tau_{l/r}^\star$ and $\widetilde{p}_{l/r}$. In other words, given $\lambda_S > 0$ and $u_{\mathbf{n}}^\star$, we compute

$$\begin{cases} \tau_l^\star = \tau_l + \dfrac{(u_{\mathbf{n}}^\star - u_{\mathbf{n},l})}{\lambda_l}, \\ \widetilde{p}_l = p_l - \lambda_l(u_{\mathbf{n}}^\star - u_{\mathbf{n},l}) \end{cases} \qquad \begin{cases} \tau_r^\star = \tau_r - \dfrac{(u_{\mathbf{n}}^\star - u_{\mathbf{n},r})}{\lambda_r}, \\ \widetilde{p}_r = p_r + \lambda_r(u_{\mathbf{n}}^\star - u_{\mathbf{n},r}). \end{cases}$$

The last unknown is the velocity $u_{\mathbf{n}}^\star$, that we shall determine in the next subsection 3.4 by using the generalized consistency conditions (16). As already specified, the latter would be equivalent to impose the node-based jump condition across the 0-wave of each subface around a given node (23). Hence, to conclude this section and since it will be shortly necessary, let us compute the jump across the 0-wave,

$$\widetilde{\mathbf{G}}_{\mathbf{n},r} - \widetilde{\mathbf{G}}_{\mathbf{n},l} - (\Delta m_{lf} + \Delta m_{rf})\mathbf{P}_{\mathbf{n}_{pf}} \tag{26}$$
$$= \lambda_l(\mathbf{V}_L^\star - \mathbf{V}_l) - \lambda_r(\mathbf{V}_R - \mathbf{V}_r^\star) + \mathbf{G}_{\mathbf{n},r} - \mathbf{G}_{\mathbf{n},l} - (\Delta m_{lf} + \Delta m_{rf})\mathbf{P}_{\mathbf{n}_{pf}} \tag{27}$$
$$= (\widetilde{p}_r - \widetilde{p}_l + g\overline{h\Delta B}_{lr})\mathbf{e}_2, \tag{28}$$

where, for the latter equality, we used the expression of the components of $\widetilde{\mathbf{G}}_{\mathbf{n},r}$ and $\widetilde{\mathbf{G}}_{\mathbf{n},l}$. In particular, we recall that $\mathbf{e}_2 = (0, 1, 0)^t$ and highlight that $\widetilde{\mathbf{G}}_{pf}^+ - \widetilde{\mathbf{G}}_{pf}^- = \widetilde{\mathbf{G}}_{\mathbf{n},r} - \widetilde{\mathbf{G}}_{\mathbf{n},l}$. Then, we sum the second equations of $(\mathcal{S}_L)$ and $(\mathcal{S}_R)$, obtaining

$$\widetilde{p}_r - \widetilde{p}_l = (\lambda_l + \lambda_r)\left\{u_{\mathbf{n}}^\star - \left[\frac{\lambda_l u_{\mathbf{n},l} + \lambda_r u_{\mathbf{n},r}}{\lambda_l + \lambda_r} - \frac{(p_r - p_l)}{\lambda_r + \lambda_l}\right]\right\},$$

which leads to

$$\widetilde{p}_r - \widetilde{p}_l + g\overline{h\Delta B}_{lr} = (\lambda_l + \lambda_r)\left\{ u^\star_{\mathbf{n}} - \left[\frac{\lambda_l u_{\mathbf{n},l} + \lambda_r u_{\mathbf{n},r}}{\lambda_l + \lambda_r} - \frac{(p_r - p_l + g\overline{h\Delta B})_{lr}}{\lambda_r + \lambda_l} \right] \right\} \\ = (\lambda_l + \lambda_r)(u^\star_{\mathbf{n}} - u^{God}_{\mathbf{n}}), \tag{29}$$

where

$$u^{God}_{\mathbf{n}} = \frac{\lambda_l u_{\mathbf{n},l} + \lambda_r u_{\mathbf{n},r}}{\lambda_l + \lambda_r} - \frac{(p_r - p_l + g\overline{h\Delta B}_{lr})}{\lambda_r + \lambda_l}. \tag{30}$$

Notice that this normal velocity, when no source terms is present, corresponds to the classical Godunov acoustic velocity, refer for instance to [64]. From this definition and (29), we conclude that if $u^\star_{\mathbf{n}} = u^{God}_{\mathbf{n}}$ (resp. $u^\star_{\mathbf{n}} \neq u^{God}_{\mathbf{n}}$), then the simple Lagrangian Riemann solver is (resp. not) classically consistent with its underlying balance law and induces (resp. does not induce) a classical conservative Godunov-type finite volume scheme. The case $u^\star_{\mathbf{n}} = u^{God}_{\mathbf{n}}$ is the classical case, here called two-point solver. In this work, we further investigate the second case for which in general $u^\star_{\mathbf{n}} \neq u^{God}_{\mathbf{n}}$ and we shall demonstrate how to retrieve a globally conservative finite volume scheme by means of node-based conservation conditions.

Finally, observe that the source term approximation $g\overline{h\Delta B}_{lr}$ at the interface has not yet been given. It will be defined according to the well-balanced property in section 3.5.1.

3.4. The Lagrangian nodal solver

In this section, the nodal parameter $\mathbf{u}_p$ is explained: what is its role in the Riemann solver and how to compute it by using the node-based consistency conditions (16). Thanks to these conditions, $\mathbf{u}_p$ will contain the information coming from all the cells around the node p. By feeding $\mathbf{u}_p$ to the flux $\widetilde{\mathbf{G}}_{pf}$ (or $\widetilde{\mathbf{F}}_{pf}$ in Eulerian coordinates), the state variables in the cells around p will be updated using information from all the cells around them, even those with which they only share a single vertex.

Let us start by reformulating the node-based consistency conditions (16) using equality (26):

$$\sum_{f \in \mathcal{SF}(p)} l_{pf}(\widetilde{p}_{rf} - \widetilde{p}_{lf} + g\overline{h\Delta B}_{lr,f})\mathbf{e}_2 = \mathbf{0},$$

and which implies

$$\sum_{f \in \mathcal{SF}(p)} l_{pf}(\widetilde{p}_{rf} - \widetilde{p}_{lf} + g\overline{h\Delta B}_{lr,f})\mathbf{n}_{pf} = \mathbf{0}, \tag{31}$$

where we used

$$(\Delta m_{lf} + \Delta m_{rf})\mathbf{P}_{\mathbf{n}_{pf}}(\Delta m_{lf}, \Delta m_{rf}, \mathbf{U}_{lf}, \mathbf{U}_{rf}) = \begin{pmatrix} 0 \\ -g\overline{h\Delta B}_{lr,f} \\ 0 \end{pmatrix}.$$

Remark that, at this stage, it is necessary to resume the pf (node-face) notation. We also observe that (31) can be seen as the approximation of $\nabla p + gh\nabla B$ in the dual cell $\omega_p = \bigcup_{c \in \mathcal{C}(p)} \omega_{pc}$, which we could interpret as the "lake at rest" equilibrium around the node and which will lead to the well-balancedness of the scheme.

Then, we notice that $u^\star_{\mathbf{n}_{pf}}$ is still an unknown parameter attached to each subface impinging at node p. Thus, the number of unknown parameters at node p is much greater than the number

of equations given by the nodal condition (31). To close this system of equations, we assume that $u^{\star}_{\mathbf{n}_{pf}}$ corresponds to the projection of the unknown nodal vector $\mathbf{u}_p$ onto the unit normal $\mathbf{n}_{pf}$, that is for all p and f

$$u^{\star}_{\mathbf{n}_{pf}} = \mathbf{u}_p \cdot \mathbf{n}_{pf}.$$

This fundamental assumption drastically reduces the number of unknowns to the vector unknown $\mathbf{u}_p$, which can be interpreted as an approximation of the nodal velocity. Thanks to (29) and (30), we are able to express conditions (31) as

$$\sum_{f \in \mathcal{SF}(p)} l_{pf}(\lambda_{l,f} + \lambda_{r,f})(\mathbf{u}_p \cdot \mathbf{n}_{pf} - u^{God}_{\mathbf{n}_{pf}})\mathbf{n}_{pf} = \mathbf{0}. \tag{32}$$

Finally, the node-based condition (31) (or equivalently (32)) boils down to the system

$$\mathbb{M}_p \, \mathbf{u}_p = \mathbf{w}_p, \tag{33}$$

where $\mathbf{u}_p$ is the unknown and

$$\mathbb{M}_p = \sum_{f \in \mathcal{SF}(p)} l_{pf}(\lambda_{l,f} + \lambda_{r,f})(\mathbf{n}_{pf} \otimes \mathbf{n}_{pf}), \qquad \mathbf{w}_p = \sum_{f \in \mathcal{SF}(p)} l_{pf}(\lambda_{l,f} + \lambda_{r,f}) u^{God}_{\mathbf{n}_{pf}} \mathbf{n}_{pf}.$$

This system, called a nodal solver [48, 61, 36], always admits a unique solution which provides an approximation of the nodal velocity $\mathbf{u}_p$, given physical states and wave speeds for all faces impinging at current node.

3.5. Properties of the Lagrangian Riemann solver

At this stage, it only remains to define a coherent definition of the source term $\mathcal{M}_{\mathbf{n}_{pf}} = g\overline{h\Delta B}_{lr,f}$ and of the wave speeds $\lambda_{lf}, \lambda_{rf}$. The latter will be determined so that the Riemann solver is positivity- and entropy-preserving. The source term approximation is found by asking for the well-balanced property. An obvious adaptation of the notations will be used as needed.

3.5.1. Well-balanced property

In the development of the simple Lagrangian Riemann solver we have stated that the contribution of the bottom topography $g\overline{h\Delta B}_{lr,f}$ is an appropriate mean value of the source term at the interface. To determine this term, we enforce that the simple Lagrangian Riemann solver satisfies the well-balanced condition: $\mathbf{V}^*_{lf} = \mathbf{V}_{lf}$ and $\mathbf{V}^*_{rf} = \mathbf{V}_{rf}$ if $\mathbf{V}_{lf}$ and $\mathbf{V}_{rf}$ verify the steady Lake at Rest (LaR) solution, see also [33, 19].

Hence, let us assume that the left and right states verify $\frac{1}{\tau_{Sf}} + B_{Sf} = \eta$ and $u_{\mathbf{n},Sf} = u_{\mathbf{t},Sf} = 0$ for $S = l, r$. Then, the nodal solver should produce $\mathbf{u}_p = \mathbf{0}$ to respect the LaR solution. To obtain such a solution, for all faces f impinging at point p, we must retrieve $u^{God}_{\mathbf{n}_{pf}} = 0$. Indeed, in this way, system (33) would be homogeneous, with the null solution the only possible one. Under the LaR condition, formula (30) reads

$$u^{God}_{\mathbf{n}_{pf}} = \frac{\lambda_{lf} u_{\mathbf{n},lf} + \lambda_{rf} u_{\mathbf{n},rf}}{\lambda_{lf} + \lambda_{rf}} - \frac{(p_{rf} - p_{lf} + g\overline{h\Delta B}_{lr,f})}{\lambda_{rf} + \lambda_{lf}} = -\frac{(p_{rf} - p_{lf} + g\overline{h\Delta B}_{lr,f})}{\lambda_{rf} + \lambda_{lf}}.$$

Hence, to obtain $u^{God}_{\mathbf{n}_{pf}} = 0$, we simply need to impose

$$p_{rf} - p_{lf} + g\overline{h\Delta B}_{lr,f} = 0,$$

and thus

$$g\overline{h\Delta B}_{lr,f} = g\frac{h_{rf}+h_{lf}}{2}(B_{rf}-B_{lf}), \tag{34}$$

where $\overline{\Delta B}_{lr,f} = B_{rf} - B_{lf}$, and $\overline{h}_{lr,f} = \dfrac{h_{rf}+h_{lf}}{2}$. Once that $\mathbf{u}_p = \mathbf{0}$ has been proved, it is clear that $u^\star_{\mathbf{n}_{pf}} = 0$ and that $\tau^\star_{lf} = \tau_{lf}$, $\tau^\star_{rf} = \tau_{rf}$ $\widetilde{p}_{lf} = p_{lf}$ and, finally, $\widetilde{p}_{rf} = p_{rf}$.

3.5.2. Positivity-preserving

In order to ensure the positivity of the specific volumes, we need to restrict the wave speed values $\lambda_S > 0$ from below. In particular, from the first equations of $(\mathcal{S}_L)$ and $(\mathcal{S}_R)$, we deduce the following conditions:

$$\lambda_l \geq -\frac{u^\star_{\mathbf{n}} - u_{\mathbf{n},l}}{\tau_l}, \qquad \text{and} \qquad \lambda_r \geq \frac{u^\star_{\mathbf{n}} - u_{\mathbf{n},r}}{\tau_r}, \tag{35}$$

which are still parametrized by the normal velocity $u^\star_{\mathbf{n}}$. Hence, such conditions are usually verified *a posteriori* and then, an iterative procedure is used when necessary.

Including constraints (35) along with the usual conditions $\lambda_l \geq \dfrac{a_l}{\tau_l}$ and $\lambda_r \geq \dfrac{a_r}{\tau_r}$, we observe that we could directly impose

$$\lambda_l \geq \frac{a_l}{\tau_l}\left[1 + \frac{(u^\star_{\mathbf{n}} - u_{\mathbf{n},l})^{(-)}}{a_l}\right], \qquad \text{and} \qquad \lambda_r \geq \frac{a_r}{\tau_r}\left[1 + \frac{(u^\star_{\mathbf{n}} - u_{\mathbf{n},r})^{(+)}}{a_r}\right], \tag{36}$$

where we recall $a = \sqrt{gh}$.

3.5.3. Entropy-preserving

To prove that the Riemann solver is entropy-stable, we need to verify conditions (17), which also read

$$\sum_{f\in\mathcal{SF}(p)} l_{pf}(\lambda_{lf}(E^*_{lf} - E_{lf}) - \lambda_{rf}(E_{rf} - E^*_{rf}) + p_{rf}u_{rf} - p_{lf}u_{lf} + g\overline{hu\Delta B}_{lr,f}) \leq 0. \tag{37}$$

In particular, for each face, we are looking for two sufficient conditions on λ_{lf} and λ_{rf} so that

$$\lambda_l(E^*_L - E_L) + \widetilde{p}_L u^*_{\mathbf{n}_{pf}} - p_L u_L \leq 0 \tag{38}$$

and

$$-\lambda_r(E_R - E^*_R) + p_R u_R - \widetilde{p}_R u^*_{\mathbf{n}_{pf}} \leq 0. \tag{39}$$

If conditions (38)-(39) are satisfied, we obtain

$$\begin{aligned}&\sum_{f\in\mathcal{SF}(p)} l_{pf}(\lambda_{lf}(E^*_{lf} - E_{lf}) - \lambda_{rf}(E_{rf} - E^*_{rf}) + p_{rf}u_{rf} - p_{lf}u_{lf} + g\overline{hu\Delta B}_{lr,f})\\ \leq &\sum_{f\in\mathcal{SF}(p)} l_{pf}((\widetilde{p}_R - \widetilde{p}_L)u^*_{\mathbf{n}_{pf}} + g\overline{hu\Delta B}_{lr,f}).\end{aligned} \tag{40}$$

Note that the term $g\overline{hu\Delta B}_{lr,f}$ has not been specified yet. Considering the previous section about the well-balanced property, here we propose to use $g\overline{hu\Delta B}_{lr,f} = gu^*_{\mathbf{n}_{pf}}\dfrac{h_{rf}+h_{lf}}{2}(B_{rf} - B_{lf})$.

Hence, from inequality (40) and relations (29)-(30), we get

$$
\begin{aligned}
&\sum_{f\in\mathcal{SF}(p)} l_{pf}(\lambda_{lf}(E^*_{lf}-E_{lf})-\lambda_{rf}(E_{rf}-E^*_{rf})+p_{rf}u_{rf}-p_{lf}u_{lf}+g\overline{hu\Delta B}_{lr,f})\\
\leq&\sum_{f\in\mathcal{SF}(p)} l_{pf}(\lambda_{lf}+\lambda_{rf})(u^\star_{\mathbf{n}_{pf}}-u^{God}_{\mathbf{n}_{pf}})u^*_{\mathbf{n}_{pf}}\\
=&\sum_{f\in\mathcal{SF}(p)} l_{pf}(\lambda_{lf}+\lambda_{rf})(u^\star_{\mathbf{n}_{pf}}-u^{God}_{\mathbf{n}_{pf}})\mathbf{u}_p\cdot\mathbf{n}_{pf}\\
=&\sum_{f\in\mathcal{SF}(p)} \mathbf{u}_p\cdot\Big(l_{pf}(\lambda_{lf}+\lambda_{rf})(u^\star_{\mathbf{n}_{pf}}-u^{God}_{\mathbf{n}_{pf}})\mathbf{n}_{pf}\Big)\\
=&\mathbf{u}_p\cdot\left(\sum_{f\in\mathcal{SF}(p)} l_{pf}(\lambda_{lf}+\lambda_{rf})(u^\star_{\mathbf{n}_{pf}}-u^{God}_{\mathbf{n}_{pf}})\mathbf{n}_{pf}\right)\\
=&\mathbf{u}_p\cdot\left(\sum_{f\in\mathcal{SF}(p)} l_{pf}(\lambda_{lf}+\lambda_{rf})(\mathbf{u}_p\cdot\mathbf{n}_{pf}-u^{God}_{\mathbf{n}_{pf}})\mathbf{n}_{pf}\right)=\mathbf{u}_p\cdot\mathbf{0}=0,
\end{aligned}
\tag{41}
$$

where we used $u^*_{\mathbf{n}_{pf}}=\mathbf{u}_p\cdot\mathbf{n}_{pf}$ and the second-to-last equality holds thanks to the nodal solver (32). Hence, condition (37) is proved provided that conditions (38), (39) hold true. Some algebraic computations show that these conditions are respectively satisfied if

$$
\lambda_{lf}\geq\sqrt{gh^*_{lf}}h_{lf} \tag{42}
$$

and

$$
\lambda_{rf}\geq\sqrt{gh^*_{rf}}h_{rf}. \tag{43}
$$

Remark 1. *We highlight that conditions* (36),(42),(43) *are implicit, hence an iterative procedure is required to ensure that they are satisfied. To start the iterative process, as initial guess for* λ_l *and* λ_r*, we may either use conditions* (36) *or*

$$
\begin{aligned}
\lambda_{lf}&=\max(\sqrt{gh_{lf}}h_{lf},\sqrt{\max(h_{lf}([\Delta p]+g\overline{h\Delta B}),0)},-[\Delta u]h_{lf}),\\
\lambda_{rf}&=\max(\sqrt{gh_{rf}}h_{rf},\sqrt{\max(-h_{rf}([\Delta p]+g\overline{h\Delta B}),0)},-[\Delta u]h_{rf}),
\end{aligned}
$$

with $[\Delta X]=X_r-X_l$*. The latter conditions have been obtained analyzing the* 1*D Riemann solver, see [36] for more details.*

Remark 2. *We highlight that the two-point scheme would have required the same conditions* (36),(42),(43) *to assure the positivity- and entropy-stability property.*

3.6. Eulerian simple Riemann solver

Let us deduce the corresponding Eulerian simple Riemann solver. Relying on the first relations of $(\mathcal{S}_r)$ and $(\mathcal{S}_l)$, we can deduce the Eulerian wave speeds Λ_l, Λ_0 and Λ_r from their Lagrangian counterparts λ_l, $\lambda_0=0$ and λ_r as

$$
\Lambda_l=u_{\mathbf{n},l}-\lambda_l\tau_l=u^\star_{\mathbf{n}}-\lambda_l\tau^\star_l,\qquad\Lambda_0=u^\star_{\mathbf{n}},\qquad\Lambda_r=u^\star_{\mathbf{n}}+\lambda_r\tau^\star_r=u_{\mathbf{n},r}+\lambda_r\tau_r.
$$

Provided that the Lagrangian approximate Riemann solver preserves the positivity of specific volumes, *i.e.*, $\tau_S^\star \geq 0$, then these Eulerian wave speeds are ordered: $\Lambda_l \leq \Lambda_0 \leq \Lambda_r$. The positivity holds true granted that the Lagrangian wave speeds satisfy condition (35). The Eulerian approximate Riemann solver is deduced from its Lagrangian counterpart as follows

$$\mathbf{W}_{\mathcal{E}}\left(\mathbf{U}_l, \mathbf{U}_r, \frac{x_{\mathbf{n}}}{t}\right) = \begin{cases} \mathbf{U}_l & \text{if} \quad \frac{x_{\mathbf{n}}}{t} \leq \Lambda_l, \\ \mathbf{U}_l^\star = \mathbf{U}(\mathbf{V}_l^\star) & \text{if} \quad \Lambda_l < \frac{x_{\mathbf{n}}}{t} \leq \Lambda_0, \\ \mathbf{U}_r^\star = \mathbf{U}(\mathbf{V}_r^\star) & \text{if} \quad \Lambda_0 < \frac{x_{\mathbf{n}}}{t} \leq \Lambda_r, \\ \mathbf{U}_r & \text{if} \quad \Lambda_r < \frac{x_{\mathbf{n}}}{t}. \end{cases}$$

Here, $\mathbf{V} \mapsto \mathbf{U}(\mathbf{V})$ is the Lagrange-to-Euler mapping defined by

$$\mathbf{V} = \tau(\mathbf{U} - h\mathbf{e}_1) + \tau\mathbf{e}_1, \qquad \text{where} \quad \mathbf{e}_1 = (1,0,0)^t.$$

The Eulerian intermediate states read $\mathbf{U}_S^\star = (h_S^\star, h_S^\star u_{\mathbf{n}}^\star, h_S^\star u_{\mathbf{t},S}^\star)^t$, while the Eulerian intermediate fluxes are given by $\widetilde{\mathbf{F}}_{\mathbf{n},S} = (h_S^\star u_{\mathbf{n}}^\star,\ h_S^\star (u_{\mathbf{n}}^\star)^2 + \widetilde{p}_S,\ h_S^\star u_{\mathbf{n}}^\star u_{\mathbf{t},S}^\star)^t$, knowing that $h_S^\star = (\tau_S^\star)^{-1}$ for $S = l, r$. We observe that the Eulerian approximate Riemann solver is also parametrized by the normal star-velocity. More importantly, by construction, the Eulerian approximate Riemann solver inherits the properties of its Lagrangian counterpart. Hence, the Eulerian Riemann solver is equally well-balanced, positivity- and entropy-preserving. See [36] for more details about the derivation of an Euler solver from its Lagrangian counterpart.

Remark 3. *While in Lagrangian coordinates we observed* $\widetilde{\mathbf{G}}_{pf}^+ - \widetilde{\mathbf{G}}_{pf}^- = \widetilde{\mathbf{G}}_{\mathbf{n},r} - \widetilde{\mathbf{G}}_{\mathbf{n},l}$*, the corresponding is generally not true in Eulerian coordinates:*

$$\widetilde{\mathbf{F}}_{pf}^+ - \widetilde{\mathbf{F}}_{pf}^- = \widetilde{\mathbf{F}}_{\mathbf{n},r} - \widetilde{\mathbf{F}}_{\mathbf{n},l} - \Lambda_0(\mathbf{U}_r^\star - \mathbf{U}_l^\star).$$

4. 2D finite volume scheme: properties and second-order extension

In this section, we discuss and analyze the multi-point finite volume scheme. We first highlight the difference between the multi-point and two-point formulations and how the multi-point scheme incorporates the source term. Then, we prove that the multi-point is positivity preserving, conservative and entropy-stable by following the lines of work [36]. We conclude by proving the well-balanced property and presenting the second-order version of the method, which preserves all of the properties of its first-order counterpart.

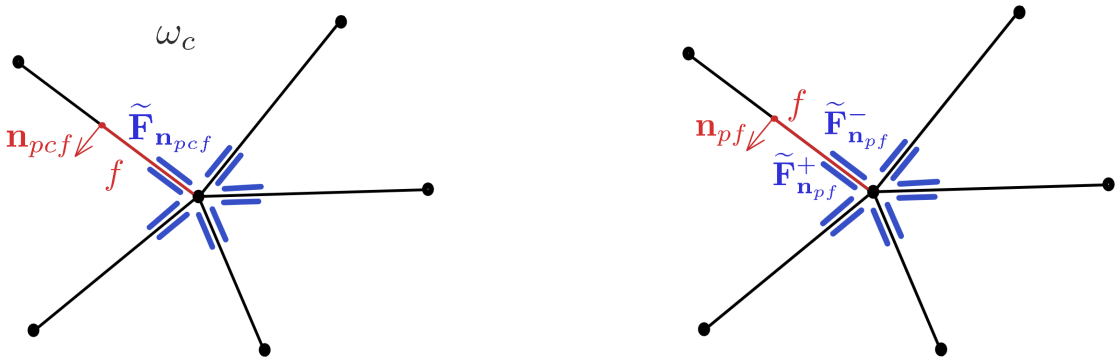
4.1. Eulerian multi-point finite volume scheme

We recall that the multi-dimensional finite volume scheme reads

$$\mathbf{U}_c^{n+1} - \mathbf{U}_c^n + \frac{\Delta t}{|\omega_c|} \sum_{p \in \mathcal{P}(c)} \sum_{f \in \mathcal{SF}(pc)} l_{pcf} \widetilde{\mathbf{F}}_{pcf} = \mathbf{0}, \tag{44}$$

where $\widetilde{\mathbf{F}}_{pcf}$ is the left-sided flux with respect to the subface f and the unit outward normal $\mathbf{n}_{pcf}$. The left- and right-sided fluxes between cells c and $d \equiv d(c,f)$ (notice that $c \equiv c(d,f)$) for a simple Eulerian Riemann solver are respectively given by (20) and (21). Taking the arithmetic average of the left and the right-sides fluxes in the normal direction $\mathbf{n}_{pcf}$ allows us to define the averaged flux on the subface f,

$$\overline{\mathbf{F}}_{\mathbf{n}_{pcf}}^{AV} = \frac{1}{2}\left[\mathbf{F}_{\mathbf{n}_{pcf}}(\mathbf{U}_c) + \mathbf{F}_{\mathbf{n}_{pcf}}(\mathbf{U}_d)\right] - \frac{1}{2}\left[\sum_{k=1}^{m} |\Lambda_k|(\mathbf{U}_{k+1} - \mathbf{U}_k)\right]_{c,d}. \tag{45}$$



(a) Cell-based notation.

(b) Face-based notation.

Figure 4: Fragment of the computational grid in the vicinity of point p: flux notation.

Since the difference between the right and left-sided Eulerian fluxes coincides with the Lagrangian one, that is

$$\widetilde{\mathbf{F}}^{+}_{\mathbf{n}_{pcf}} - \widetilde{\mathbf{F}}^{-}_{\mathbf{n}_{pcf}} = \widetilde{\mathbf{G}}^{+}_{\mathbf{n}_{pcf}} - \widetilde{\mathbf{G}}^{-}_{\mathbf{n}_{pcf}} = (\widetilde{p}_d - \widetilde{p}_c)\,\mathbf{e}_2,$$

combining the foregoing equation with formula (45), we arrive at

$$\begin{aligned}\widetilde{\mathbf{F}}^{-}_{\mathbf{n}_{pcf}} = {} & \frac{1}{2}\left[\mathbf{F}_{\mathbf{n}_{pcf}}(\mathbf{U}_c) + \mathbf{F}_{\mathbf{n}_{pcf}}(\mathbf{U}_d)\right] - \frac{1}{2}\left[\sum_{k=1}^{m} |\Lambda_k|(\mathbf{U}_{k+1} - \mathbf{U}_k)\right]_{c,d} \\ & - \frac{\lambda_c + \lambda_d}{2}\left[\mathbf{u}_p \cdot \mathbf{n}_{pcf} - u^{God,ws}_{\mathbf{n}_{pcf}}\right]\mathbf{e}_2,\end{aligned} \tag{46}$$

with $u^{God}_{\mathbf{n}_{pcf}} = u^{God,ws}_{\mathbf{n}_{pcf}} - g\overline{h\Delta B}_{lr,pf}$ (where "ws" stands for "without source"). Formula (46) can also be expressed as in the following,

$$\begin{aligned}\widetilde{\mathbf{F}}_{\mathbf{n}_{pcf}} \equiv \widetilde{\mathbf{F}}^{-}_{\mathbf{n}_{pcf}} &= \overline{\mathbf{F}}^{AV}_{\mathbf{n}_{pcf}} - \frac{1}{2}\left[\widetilde{p}_d - \widetilde{p}_c\right]\mathbf{e}_2 \\ &= \overline{\mathbf{F}}^{AV}_{\mathbf{n}_{pcf}} - \frac{\lambda_c + \lambda_d}{2}\left[\mathbf{u}_p \cdot \mathbf{n}_{pcf} - u^{God,ws}_{\mathbf{n}_{pcf}}\right]\mathbf{e}_2 \\ &= \overline{\mathbf{F}}^{AV}_{\mathbf{n}_{pcf}} - \frac{\lambda_c + \lambda_d}{2}\left[\mathbf{u}_p \cdot \mathbf{n}_{pcf} - u^{God}_{\mathbf{n}_{pcf}}\right]\mathbf{e}_2 \quad + \frac{1}{2}g\overline{h\Delta B}_{lr,pf}\mathbf{e}_2 \\ &= \overline{\mathbf{F}}_{\mathbf{n}_{pcf}} \quad + \frac{1}{2}\overline{\mathcal{M}}_{\mathbf{n}_{pcf}},\end{aligned} \tag{47}$$

where $\overline{\mathbf{F}}_{\mathbf{n}_{pcf}} = \overline{\mathbf{F}}^{AV}_{\mathbf{n}_{pcf}} - \dfrac{\lambda_c + \lambda_d}{2}\left[\mathbf{u}_p \cdot \mathbf{n}_{pcf} - u^{God}_{\mathbf{n}_{pcf}}\right]\mathbf{e}_2$, $\overline{\mathcal{M}}_{\mathbf{n}_{pcf}} = (0, g\overline{h\Delta B}_{lr,pf}, 0)^t$ with $g\overline{h\Delta B}_{lr,pf} = g\dfrac{h_{Lpf} + h_{Rpf}}{2}(B_{Rpf} - B_{Lpf})$. Refer also to figure 4 for the flux notation. Hence, the multi-point finite volume scheme (44) also reads

$$\mathbf{U}^{n+1}_c - \mathbf{U}^{n}_c + \frac{\Delta t}{|\omega_c|}\sum_{p\in\mathcal{P}(c)}\sum_{f\in\mathcal{SF}(pc)} l_{pcf}(\overline{\mathbf{F}}_{pcf} + \frac{1}{2}\overline{\mathcal{M}}_{pcf}) = \mathbf{0},$$

where $w_{pcf}\overline{\mathbf{S}}_{pcf} = -\dfrac{l_{pcf}}{2}\overline{\mathcal{M}}_{pcf}$.

Observe that, for the classical two-point scheme, we would simply have $\widetilde{\mathbf{F}}_{\mathbf{n}_{pcf}} = \overline{\mathbf{F}}^{AV}_{\mathbf{n}_{pcf}} + \frac{1}{2}\overline{\mathcal{M}}_{\mathbf{n}_{pcf}}$, or alternatively, $\overline{\mathbf{F}}^{-}_{\mathbf{n}_{pcf}} = \overline{\mathbf{F}}^{+}_{\mathbf{n}_{pcf}} = \overline{\mathbf{F}}^{AV}_{\mathbf{n}_{pcf}}$. Therefore, for the two-point scheme, the conservation is simply retrieved by cancellation when summation over the cells is invoked. Hence, the explicit expression of the numerical flux (47) shows that, in the unlikely event $\mathbf{u}_p \cdot \mathbf{n}_{pcf} = u^{God}_{\mathbf{n}_{pcf}}$, we retrieve the classical conservative face-based finite volume scheme. This happens for instance when the initial solution satisfies the LaR steady state or for trivial solutions. Otherwise, the general situation $\mathbf{u}_p \cdot \mathbf{n}_{pcf} \neq u^{God}_{\mathbf{n}_{pcf}}$ occurs. We recall that the value of $\mathbf{u}_p$ is the approximate solution of the nodal solver (32), while $u^{God}_{\mathbf{n}_{pcf}}$ is explicitly given by (30).

4.2. Preservation of the definition domain: time step condition

In this section, we introduce the concept of invariant domain [9], namely a region where, for any initial condition $\mathbf{U}^0$ inside the region, the solution of the hyperbolic system stays within the domain for all time $t > 0$. In the case of the shallow water system, the definition domain is $\mathcal{D}_{\mathcal{E}} = \{\mathbf{U} \text{ such that } h > 0\}$. For the Lagrangian counterpart, we clearly have $\mathcal{D}_{\mathcal{L}} = \{\mathbf{V} \text{ such that } \tau > 0\}$. Remark that $\mathcal{D}_{\mathcal{E}}$ and $\mathcal{D}_{\mathcal{L}}$ are convex and we shall study under which condition on the time step Δt our finite volume discretization preserves the definition domain.

Using the following geometric identity

$$\sum_{p\in\mathcal{P}(c)} \sum_{f\in\mathcal{SF}(pc)} l_{pcf}\mathbf{n}_{pcf} = \mathbf{0}, \tag{48}$$

the multi-dimensional finite volume scheme (9) also reads

$$\mathbf{U}_c^{n+1} - \mathbf{U}_c^n + \frac{\Delta t}{|\omega_c|} \sum_{p\in\mathcal{P}(c)} \sum_{f\in\mathcal{SF}(pc)} l_{pcf} \left[\widetilde{\mathbf{F}}_{pcf} - \mathbb{F}(\mathbf{U}_c^n)\mathbf{n}_{pcf} \right] = \mathbf{0}. \tag{49}$$

We already proved that the approximate Eulerian Riemann solver is $\mathcal{D}_{\mathcal{E}}$-preserving, that is, if $\mathbf{U}_c^n \in \mathcal{D}_{\mathcal{E}}$ then $\mathbf{W}_{pcf,\mathcal{E}}(\xi) \in \mathcal{D}_{\mathcal{E}}$ for all $\xi \in \mathbb{R}$. Hence, we aim to prove that the multi-point scheme (49) is itself $\mathcal{D}_{\mathcal{E}}$-preserving, that is, $\mathbf{U}_c^{n+1} \in \mathcal{D}_{\mathcal{E}}$ if $\mathbf{U}_c^n \in \mathcal{D}_{\mathcal{E}}$. To this end, we introduce $\xi^{\min}_{pcf} \geq 0$ such that

$$\mathbf{W}_{pcf,\mathcal{E}}(\xi) = \mathbf{U}_c^n, \text{ for } \xi < -\xi^{\min}_{pcf}.$$

Formulation (18) allows us to develop the subface flux expression as follows

$$\widetilde{\mathbf{F}}_{pcf} = \mathbb{F}(\mathbf{U}_c^n)\mathbf{n}_{pcf} + \xi^{\min}_{pcf}\mathbf{U}_c^n - \int_{-\xi^{\min}_{pcf}}^{0} \mathbf{W}_{pcf,\mathcal{E}}(\xi)\,\mathrm{d}\xi.$$

Substituting the subface flux into scheme (49) and collecting the terms, we arrive at

$$\mathbf{U}_c^{n+1} = \left(1 - \frac{\Delta t}{|\omega_c|} \sum_{p\in\mathcal{P}(c)} \sum_{f\in\mathcal{SF}(pc)} l_{pcf}\xi^{\min}_{pcf} \right) \mathbf{U}_c^n + \frac{\Delta t}{|\omega_c|} \sum_{p\in\mathcal{P}(c)} \sum_{f\in\mathcal{SF}(pc)} l_{pcf} \int_{-\xi^{\min}_{pcf}}^{0} \mathbf{W}_{pcf,\mathcal{E}}(\xi)\,\mathrm{d}\xi.$$

At this stage, we only need to show that $\mathbf{U}_c^{n+1}$ is a convex combination of $\mathbf{U}_c^n$ and the intermediate states of the subface-based approximate Riemann solvers. Thus, $\mathbf{U}_c^{n+1} \in \mathcal{D}_{\mathcal{E}}$ provided that the time step satisfies the following condition:

$$\Delta t \leq \frac{|\omega_c|}{\displaystyle\sum_{p\in\mathcal{P}(c)} \sum_{f\in\mathcal{SF}(pc)} l_{pcf}\xi^{\min}_{pcf}} = \Delta t_c. \tag{50}$$

Finally, to ensure that $\mathbf{U}_c^{n+1} \in \mathcal{D}_{\mathcal{E}}$ for all c, we impose the global time-step condition

$$\Delta t \leq \min_c \Delta t_c. \tag{51}$$

In practice, the latter condition is ensured provided that

$$\Delta t \leq \min_c \left[\frac{|\omega_c|}{\displaystyle\sum_{p \in \mathcal{P}(c)} \sum_{f \in \mathcal{SF}(pc)} l_{pcf} \max(|\Lambda_{pcf,L}|, |\Lambda_{pcf,R}|)} \right], \tag{52}$$

in view of the approximate Riemann solver we previously presented.

4.3. Conservation property

In this section, we show that the presented subface-based finite volume scheme is conservative when there is no source term. We do not provide all the details but we refer the reader to [36] for a similar argument for systems of conservation laws.

We have already seen that the multi-point finite volume scheme reads

$$|\omega_c|(\mathbf{U}_c^{n+1} - \mathbf{U}_c^n) + \Delta t \sum_{p \in \mathcal{P}(c)} \sum_{f \in \mathcal{SF}(pc)} l_{pcf} \widetilde{\mathbf{F}}_{pcf} = 0.$$

Neglecting the boundary conditions, such a method is conservative in the classical way (if there is no source term) or "balanced" if and only if

$$\sum_c |\omega_c|(\mathbf{U}_c^{n+1} - \mathbf{U}_c^n) = -\Delta t \sum_c \sum_{p \in \mathcal{P}(c)} \sum_{f \in \mathcal{SF}(pc)} \frac{l_{pcf}}{2} \overline{\mathcal{M}}_{pcf},$$

hence, if and only if

$$\sum_c \sum_{p \in \mathcal{P}(c)} \sum_{f \in \mathcal{SF}(pc)} l_{pcf} (\widetilde{\mathbf{F}}_{pcf} - \frac{1}{2} \overline{\mathcal{M}}_{pcf}) = \mathbf{0},$$

or equivalently

$$\sum_p \sum_{c \in \mathcal{C}(p)} \sum_{f \in \mathcal{SF}(pc)} l_{pcf} (\widetilde{\mathbf{F}}_{pcf} - \frac{1}{2} \overline{\mathcal{M}}_{pcf}) = \mathbf{0},$$

where $\mathcal{C}(p)$ is the set of cells sharing the point p. Therefore, a sufficient condition to ensure the conservation property reads

$$\sum_{c \in \mathcal{C}(p)} \sum_{f \in \mathcal{SF}(pc)} l_{pcf} (\widetilde{\mathbf{F}}_{pcf} - \frac{1}{2} \overline{\mathcal{M}}_{pcf}) = \mathbf{0} \qquad \forall p. \tag{53}$$

The latter condition can be reformulated as

$$\sum_{f \in \mathcal{SF}(p)} l_{pf} (\widetilde{\mathbf{F}}_{pf}^- + \widetilde{\mathbf{F}}_{pf}^+ - \overline{\mathcal{M}}_{pcf}) = \mathbf{0} \qquad \forall p, \tag{54}$$

where we used the fact the sum over the cells c sharing p of the fluxes attached to the subfaces impinging at p is equal to the sum over the left-sided and the right-sided fluxes attached to the subfaces impinging at p. In the previous equation, $\widetilde{\mathbf{F}}_{pf}^-$ (resp. $\widetilde{\mathbf{F}}_{pf}^+$) denotes respectively the left-sided (resp. right-sided) flux attached to the subface f. It is then clear that, using formulas as (18) of the previous section, conditions (54) lead to the generalized consistency conditions (12), which we already proved to be satisfied for each node p in section 3.4.

4.4. Entropy-stability

In section 3.5.3, we showed that the approximate Riemann solver is entropy-stable under appropriate conditions on the speeds of propagation. Thanks to this, we will now prove that the multi-point finite volume scheme (44) is entropy-stable as well.

Let us recall that $\mathbf{U}_c^{n+1}$ is a convex combination of $\mathbf{U}_c^n$ and $\frac{1}{\xi_{pcf}^{\min}}\int_{-\xi_{pcf}^{\min}}^{0}\mathbf{W}_{pcf,\mathcal{E}}(\xi)\,\mathrm{d}\xi$, provided that $\mathbf{U}_c^n \in \mathcal{D}_{\mathcal{E}}$, $\mathbf{W}_{pcf,\mathcal{E}}(\xi) \in \mathcal{D}_{\mathcal{E}}$ and Δt satisfies the time step conditions (50)-(51), see section 4.2. Exploiting this fact and the convexity of the entropy, it follows

$$\Sigma_c^{n+1} \leq \left(1 - \frac{\Delta t}{|\omega_c|}\sum_{p\in\mathcal{P}(c)}\sum_{f\in\mathcal{SF}(pc)} l_{pcf}\xi_{pcf}^{\min}\right)\Sigma_c^n + \frac{\Delta t}{|\omega_c|}\sum_{p\in\mathcal{P}(c)}\sum_{f\in\mathcal{SF}(pc)} l_{pcf}\xi_{pcf}^{\min}\Sigma\left(\frac{1}{\xi_{pcf}^{\min}}\int_{-\xi_{pcf}^{\min}}^{0}\mathbf{W}_{pcf,\mathcal{E}}(\xi)\,\mathrm{d}\xi\right),$$

and then

$$\Sigma_c^{n+1} - \Sigma_c^n \leq \frac{\Delta t}{|\omega_c|}\sum_{p\in\mathcal{P}(c)}\sum_{f\in\mathcal{SF}(pc)} l_{pcf}\int_{-\infty}^{0}\left(\Sigma(\mathbf{W}_{pcf,\mathcal{E}}(\xi)) - \Sigma_c^n\right)\,\mathrm{d}\xi,$$

where we used that, if $\xi \leq \xi_{pcf}^{\min}$, then $\mathbf{W}_{pcf,\mathcal{E}}(\xi) = \mathbf{U}_c^n$, together with the Jensen inequality. Finally, using the geometric identity (48), we get

$$\Sigma_c^{n+1} - \Sigma_c^n + \frac{\Delta t}{|\omega_c|}\sum_{p\in\mathcal{P}(c)}\sum_{f\in\mathcal{SF}(pc)} l_{pcf}\left[\Theta(\mathbf{U}_c^n)\cdot\mathbf{n}_{pcf} - \int_{-\infty}^{0}\left(\Sigma(\mathbf{W}_{pcf,\mathcal{E}}(\xi)) - \Sigma(\mathbf{U}_c^n)\right)\,\mathrm{d}\xi\right] \leq 0, \quad (55)$$

which clearly is the discrete counterpart of the continuous entropy inequality (5). As such, it is natural to define the subface entropy flux as

$$\widetilde{\Theta}_{pcf} = \Theta(\mathbf{U}_c^n)\cdot\mathbf{n}_{pcf} - \int_{-\infty}^{0}\left(\Sigma(\mathbf{W}_{pcf,\mathcal{E}}(\xi)) - \Sigma(\mathbf{U}_c^n)\right)\,\mathrm{d}\xi, \quad (56)$$

so that inequality (55) becomes

$$\Sigma_c^{n+1} - \Sigma_c^n + \frac{\Delta t}{|\omega_c|}\sum_{p\in\mathcal{P}(c)}\sum_{f\in\mathcal{SF}(pc)} l_{pcf}\widetilde{\Theta}_{pcf} \leq 0, \quad (57)$$

or, equivalently,

$$\Sigma_c^{n+1} - \Sigma_c^n + \frac{\Delta t}{|\omega_c|}\sum_{p\in\mathcal{P}(c)}\sum_{f\in\mathcal{SF}(pc)} l_{pcf}\left(\overline{\Theta}_{pcf} - \frac{w_{lpcf}}{l_{pcf}}\overline{\Phi}_{pcf}\right) \leq 0$$

with $\widetilde{\Theta}_{pcf} = \overline{\Theta}_{pcf} - \frac{w_{lpcf}}{l_{pcf}}\overline{\Phi}_{pcf}$. Similarly, in Lagrangian coordinates we get

$$\sigma_c^{n+1} - \sigma_c^n + \frac{\Delta t}{|\omega_c|}\sum_{p\in\mathcal{P}(c)}\sum_{f\in\mathcal{SF}(pc)} l_{pcf}\widetilde{\theta}_{pcf} \leq 0.$$

At this stage, we have all the ingredients to prove that the total entropy Σ_B over the the computational domain (neglecting the boundary conditions) is non increasing, that is

$$\sum_c |\omega_c|\left(\Sigma_c^{n+1} - \Sigma_c^n\right) \leq \Delta t\sum_c\sum_{p\in\mathcal{P}(c)}\sum_{f\in\mathcal{SF}(pc)} w_{lpcf}\overline{\Phi}_{pcf}. \quad (58)$$

Thanks to the local in-cell inequality (57), we can state that the global entropy inequality (58) holds true provided that

$$\sum_{c}\sum_{p\in\mathcal{P}(c)}\sum_{f\in\mathcal{SF}(p)} l_{pcf}(\widetilde{\Theta}_{pcf}+\frac{w_{lpcf}}{l_{pcf}}\overline{\Phi}_{pcf})\geq 0,$$

or equivalently

$$\sum_{p}\sum_{c\in\mathcal{C}(p)}\sum_{f\in\mathcal{SF}(p)} l_{pcf}(\widetilde{\Theta}_{pcf}+\frac{w_{lpcf}}{l_{pcf}}\overline{\Phi}_{pcf})\geq 0,$$

where $\mathcal{C}(p)$ is the set of cells sharing the point p. Therefore, to ensure that the finite volume scheme satisfies the global entropy inequality (58), it is sufficient to prove that

$$\sum_{c\in\mathcal{C}(p)}\sum_{f\in\mathcal{SF}(p)} l_{pcf}(\widetilde{\Theta}_{pcf}+\frac{w_{lpcf}}{l_{pcf}}\overline{\Phi}_{pcf})\geq 0. \tag{59}$$

Similarly to what we have done in section 4.3 (formulas (53)-(54)), condition (59) can be reformulated as

$$\sum_{f\in\mathcal{SF}(p)} l_{pf}\left(\widetilde{\Theta}^{l}_{pf}+\widetilde{\Theta}^{r}_{pf}+2\frac{w_{lpcf}}{l_{pcf}}\overline{\Phi}_{pcf}\right)\geq 0,$$

and thus,

$$\sum_{f\in\mathcal{SF}(p)} l_{pf}\left[\widetilde{\Theta}^{l}_{pf}(\mathbf{U}_{lf},\mathbf{U}_{rf},\mathbf{n}_{pf},\mathbf{v}_{p})+\widetilde{\Theta}^{r}_{pf}(\mathbf{U}_{rf},\mathbf{U}_{lf},-\mathbf{n}_{pf},\mathbf{u}_{p})+2\frac{w_{lpcf}}{l_{pcf}}\overline{\Phi}_{pcf}\right]\geq 0.$$

Finally, using the expression of the subface entropy flux (56) in terms of the simple Riemann solver and the symmetry assumption of the approximate Riemann solver $\mathbf{W}_{pf,\mathcal{E}}$, we arrive at conditions (13). The latter have already been proved in section 3.5.3 under the assumption of conditions (42)-(43) on the speeds of propagation.

4.5. Well-balanced property

Previously, we demonstrated that both the Eulerian ($\mathcal{RP}$) and Lagrangian ($\mathcal{RP}_{\mathcal{L}}$) approximate Riemann solvers are well-balanced. In this section, we briefly establish that this property extends to the multi-dimensional finite volume scheme (44), namely it preserves the LaR steady states without any further modification. To prove this, we assume that $\mathbf{u}_c=(0,0)^t$ and $h_c+B_c=\eta$ with η a constant and for each cell c. For the sake of simplicity, let us consider the flux $\widetilde{\mathbf{F}}_{\mathbf{n}_{pf}}$ in the normal direction $\mathbf{n}_{pf}$ to a given face f. It immediately follows that $u^{God}_{\mathbf{n}_{pf}}=0$ and $u^{*}_{\mathbf{n}_{pf}}=0$ thanks to definition (34) for the source term $g\overline{h\Delta B}_{lr,f}$. As a result, the first and third components of the flux $\widetilde{\mathbf{F}}_{\mathbf{n}_{pcf}}$ are clearly null. Now, considering the second component of the flux, we are left with $\widetilde{\mathbf{F}}^{(2)}_{\mathbf{n}_{pcf}}=\frac{1}{2}\left[p_{pcf}+p_{pdf}\right]+\frac{1}{2}g\overline{h\Delta B}_{lr,pcf}$. Hence, we have

$$\sum_{p\in\mathcal{P}(c)}\sum_{f\in\mathcal{SF}(pc)}\frac{l_{pcf}}{2}\left[p_{pdf}+p_c+g\overline{h\Delta B}_{lr,pcf}\right]\mathbf{n}_{pcf}=$$
$$\sum_{p\in\mathcal{P}(c)}\sum_{f\in\mathcal{SF}(pc)}\frac{l_{pcf}}{2}\left[p_{pdf}-p_c+g\overline{h\Delta B}_{lr,pcf}\right]\mathbf{n}_{pcf}=\mathbf{0},$$

where, for the first equality, we used the geometric identity (48).

4.6. Second-order extension

The second-order extension of the finite volume scheme (44) is obtained by using a classical piece-wise linear limited reconstruction technique on the variables $\mathbf{U} = (h, (hu), (hv))^t$ and also $(h + B)$. First, given any of these variables $w_c^n \in \{h_c^n, (hu)_c^n, (hv)_c^n, (h + B)_c^n\}$, the cell-centered mean values in the vicinity of cell c, that is for any $d \in \mathcal{N}(c) \equiv \bigcup_{p\in\mathcal{P}(c)} \mathcal{C}(p)$, we search for the best piece-wise linear reconstruction, $\hat{w}_c^n(\mathbf{x})$, such that

$$\int_{\omega_c} \hat{w}_c^n(\mathbf{x})\, dv = |\omega_c|\, w_c^n, \quad \text{and} \quad \arg\min \left(\sum_{d\in\mathcal{N}(c)} \left| \int_{\omega_d} \hat{w}_c^n(\mathbf{x})\, dv - |\omega_d|\, w_d^n \right|^2 \right).$$

Next, due to the presence of possible steep gradients this reconstruction is limited and we choose the classical minmod [44] or Ventakakrishnan [66] limiters. Notice that any of these limiters preserves the positivity of any reconstructed value $\hat{h}_c^n(\mathbf{x}) > 0$ for any $\mathbf{x} \in \omega_c$. Also the reconstructed value of the bathymetry is deduced from $\widehat{(h + B)}$ and $\hat{h}_c^n$ as $\hat{B}_c^n(\mathbf{x}) = \widehat{(h + B)}_c^n(\mathbf{x}) - \hat{h}_c^n(\mathbf{x})$. Remark that the mean value of the reconstructed bathymetry in the cell is correctly retrieved as being B_c^n, because

$$\int_{\omega_c} \hat{B}_c^n(\mathbf{x})dv = \int_{\omega_c} \widehat{(h + B)}_c^n(\mathbf{x}) - \hat{h}_c^n(\mathbf{x})dv = |\omega_c|(h_c^n + B_c^n) - |\omega_c|h_c^n = |\omega_c|B_c^n.$$

Furthermore, remark that, in order to preserve LaR solution with the second-order scheme, an additional term has to be added. In the normal direction $\mathbf{n}_{pcf}$, we would have

$$\mathbf{U}_c^{n+1} - \mathbf{U}_c^n + \frac{\Delta t}{|\omega_c|} \sum_{p\in\mathcal{P}(c)} \sum_{f\in\mathcal{SF}(pc)} l_{pcf}(\widehat{\overline{\mathbf{F}}}_{\mathbf{n}_{pcf}} + (\frac{1}{2}\widehat{\mathcal{M}}_{\mathbf{n}_{pcf}} + \mathcal{M}^C_{\mathbf{n}_{pcf}})\mathbf{e}_2) = \mathbf{0}$$

where

$$\mathcal{M}^C_{\mathbf{n}_{pcf}} = g\frac{\widehat{h_{pcf}} + h_{pcf}}{2}(\widehat{B}_{pcf} - B_{pcf}). \tag{60}$$

This term can be interpreted as the in-cell second-order contribution of the source term. Note that term (60) is automatically equal to zero if no reconstruction procedure is used. Finally, for the discretization in time a simple 2nd order 2 step Runge-Kutta (RK2) scheme is employed. Details are omitted as those techniques are truly classical ones.

At last, regarding the positivity- and entropy-stability properties, they are still maintained by using respectively conditions (35) and (42)-(43) on the speeds of propagation, which are now computed using the reconstructed variables.

As a summary we have developed a well-balanced 2D FV scheme based on a non-classical Lagrangian-related Riemann solver and a nodal conservation feature which is provably positivity preserving and also entropy decreasing. This FV scheme works on any type of grids and the following numerical section presents some results on structured quadrangular and unstructured triangular grids in 2D in its first and second-order accurate versions.

5. Numerical results

In this section, we present several test cases to assess the properties of both the two-point and multi-point schemes. The main purpose is not only assess the well-balanced character and the order of accuracy of the schemes, but also to study the difference between the two-point and

multi-point schemes in terms of shock-generated spurious instabilities. The CFL is set to 0.5 and 0.25 for first- and second-order schemes respectively, the gravity constant is given by $g = 9.81$, and we use MINMOD slope limiter for the second-order scheme unless otherwise stated.

In the following, we simulate the following problems:

- Subsonic regime [14]: test to check the order of convergence of the scheme.
- Flow over three mounds (and perturbation): this test presents a non-flat bottom topography to check the WB property.
- Radial dam break with flat [28] and non-flat topography: sanity check to assess the wave propagation.
- Oblique hydraulic jump [28]: to check the ability to capture the exact discontinuous solution.
- 2D Riemann problem [51]: to check the ability of the scheme in presence of complex wave structures.
- Colliding flow [31], odd-even decoupling [55], flow past a cylinder [42]: to measure the susceptibility to small perturbation and Carbuncle effect.

5.1. Subsonic regime

We start by showing that the first- and second-order methods correctly reach the corresponding order of accuracy. We consider the 1D test presented in [14] and extend it to 2D. We highlight that we use triangular unstructured grids. The domain is a rectangle of size $[0, 20\text{m}] \times [0, 5\text{m}]$ and the initial conditions read

$$h(x, y, t = 0) = 2 - 0.1e^{-(x-10)^2}, \quad u(x, y, t = 0) \equiv 0, \quad v(x, y, t = 0) \equiv 0,$$

while the topography is given by

$$B(x, y) = 0.1 - 0.01e^{-(x-10)^2}.$$

Outflow boundary conditions are used at the left ($x = 0\text{m}$) and right ($x = 20\text{m}$) sides of the boundary, while we impose classical no-slip wall boundary conditions at the top ($y = 5\text{m}$) and bottom ($y = 0\text{m}$). Since the exact solution is not known, we compute a reference one using the second-order scheme and $M = 1007616$ cells.

Tables 1 and 2 present the errors and orders of accuracy for the variables h, u and v for the first- and second-order schemes respectively. Observe that, for the latter scheme, we did not use any limiter for this test. The results are considered satisfying as the expected orders of accuracy are reached.

5.2. Flow over three mounds

This test case has a complex bottom configuration and allows to verify the well-balanced property of the schemes. The bottom topography for the domain $\Omega = [0, 40\text{m}] \times [0, 40\text{m}]$ is defined by the function:

$$B(x, y) = \max\left[0,\, 1 - \frac{1}{8}R_{10,11},\, 1 - \frac{3}{10}R_{10,31},\, 1 - \frac{4}{10}R_{27,20}\right], \qquad R_{a,b} = \sqrt{(x-a)^2 + (y-b)^2}.$$

Hence, we initialize a lake at rest solution, that is the initial conditions are $h + B = 4\text{m}$ and $u = v = 0\text{m/s}$, and the final time is $t = 200\text{s}$. The mesh is constituted of 100×100 uniform squares and outflow boundary conditions are considered.

Variable	Mesh M	$\mathbf{L}^1$	$\mathcal{O}(\mathbf{L}^1)$	$\mathbf{L}^2$	$\mathcal{O}(\mathbf{L}^2)$
h	984	2.887×10^{-3}	—	4.904×10^{-3}	—
	3936	1.656×10^{-3}	0.80	2.893×10^{-3}	0.76
	15744	9.030×10^{-4}	0.87	1.609×10^{-3}	0.84
	62976	4.740×10^{-4}	0.92	8.543×10^{-4}	0.91
u	984	6.217×10^{-3}	—	1.094×10^{-2}	—
	3936	3.621×10^{-3}	0.77	6.527×10^{-3}	0.74
	15744	1.989×10^{-3}	0.86	3.646×10^{-3}	0.84
	62976	1.049×10^{-3}	0.92	1.942×10^{-3}	0.90
v	984	1.139×10^{-3}	—	2.006×10^{-3}	—
	3936	6.779×10^{-4}	0.75	1.240×10^{-3}	0.69
	15744	3.799×10^{-4}	0.83	7.181×10^{-4}	0.78
	62976	2.049×10^{-4}	0.89	3.981×10^{-4}	0.85

Table 1: Accuracy test in subsonic regime. $\mathbf{L}^1$, $\mathbf{L}^2$ errors and associated orders of accuracy for variables h, u and v for the first-order scheme.

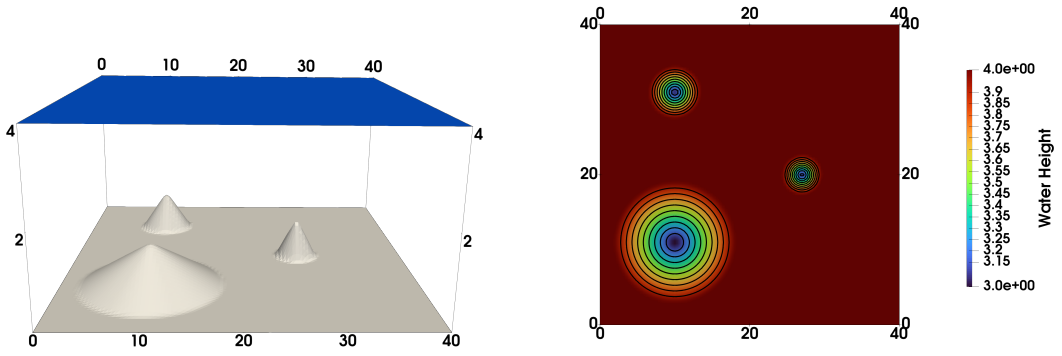
Variable	Mesh M	$\mathbf{L}^1$	$\mathcal{O}(\mathbf{L}^1)$	$\mathbf{L}^2$	$\mathcal{O}(\mathbf{L}^2)$
h	984	1.279×10^{-3}	—	2.206×10^{-3}	—
	3936	3.667×10^{-4}	1.80	6.877×10^{-4}	1.68
	15744	9.918×10^{-5}	1.88	1.913×10^{-4}	1.84
	62976	2.529×10^{-5}	1.97	4.976×10^{-5}	1.94
u	984	2.709×10^{-3}	—	4.765×10^{-3}	—
	3936	7.981×10^{-4}	1.76	1.526×10^{-3}	1.64
	15744	2.111×10^{-4}	1.91	4.183×10^{-4}	1.86
	62976	5.202×10^{-5}	2.02	1.045×10^{-4}	2.00
v	984	5.200×10^{-4}	—	1.169×10^{-3}	—
	3936	1.740×10^{-4}	1.57	4.093×10^{-4}	1.51
	15744	5.723×10^{-5}	1.60	1.279×10^{-4}	1.67
	62976	1.651×10^{-5}	1.79	3.561×10^{-5}	1.84

Table 2: Accuracy test in subsonic regime. $\mathbf{L}^1$, $\mathbf{L}^2$ errors and associated orders of accuracy for variables h, u and v for the second-order scheme.

In figure 5, we present the results obtained with the first-order multi-point scheme. In particular, on the left we insert the free surface along with the bottom topography. On the right of the figure, we show the water height with 10 isolines h from $h^{\min} = 3.15$m to $h^{\max} = 4$m. We clearly see that the initial state is preserved, specifically up to a machine error of order 10^{-15}. Notice that the WB property is obtained independently of the bottom topography B. Finally, we confirm that analogous results have been obtained with both the second-order extension of the method and the two-point scheme. We do not insert the corresponding results as they appear identical to the inserted ones.

5.2.1. Perturbation of the stationary solution

An additional test to check the well-balanced property of the scheme consists in inserting a perturbation in the stationary solution. Then, one should verify that the perturbation propagates



(a) 3D view of free surface $h + B$

(b) Iso-lines of h.

Figure 5: Flow over three mounds. Numerical solution at $t = 200s$ by the multi-point schemes on a 100×100 square grid.

away of the domain so that the stationary solution is recovered. Hence, we modified the initial condition for the water height, imposing

$$h(x, y, t = 0) = 4 - B(x, y) + \begin{cases} 10^{-3} & \text{if} \quad 19 \leq x, y \leq 21, \\ 0 & \text{otherwise,} \end{cases}$$

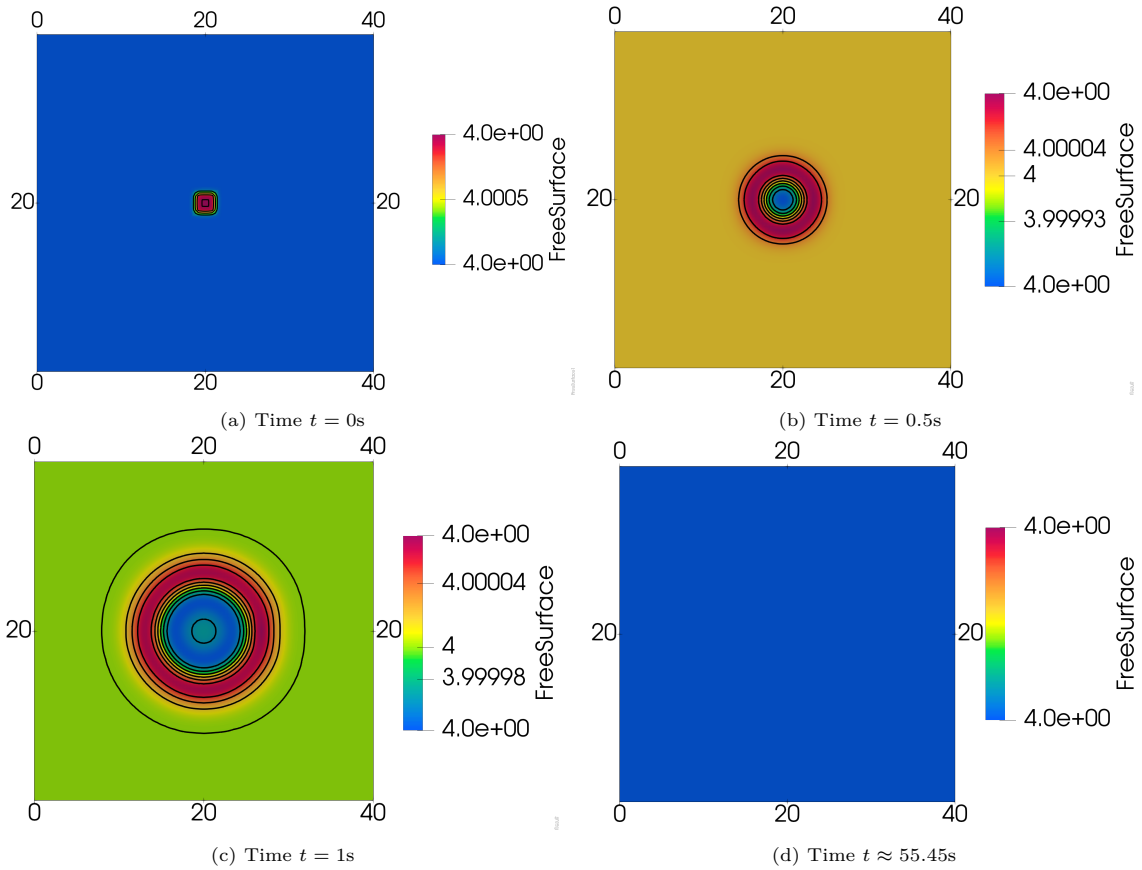
and we checked that the numerical schemes behave correctly. In particular, we show the propagation of the perturbation in the free surface $h + B$ in figure 6, for which we used the first-order multi-point scheme. Observe that we provide the solution at four differents times: $t = 0$s, $t = 0.5$s, $t = 1$s and $t \approx 55.45$s. In particular, the latter time was determined by stopping the simulation when the residuals reached a value smaller than 10^{-10}. At last, we confirm that the use of the other schemes provides similar results.

5.3. Radial dam break

The circular dam-break problem on flat bottom from [28] is a benchmark test case to assess the behavior of the schemes. This problem involves the breaking of a circular dam allowing to verify the performance of the algorithm when solving symmetry flow problems. The computational domain is a square $\omega \in [0, 50\text{m}] \times [0, 50\text{m}]$ separating two regions by a cylindrical wall of radius $r_0 = 11$m, for which the initial conditions are

$$h(x, y, t = 0) = \begin{cases} 10\text{m} & \text{if} \quad r \leq r_0, \\ 1\text{m} & \text{if} \quad \text{else,} \end{cases} \qquad u(x, y, t = 0) = v(x, y, t = 0) \equiv 0\text{m/s},$$

where $r = \sqrt{(x - 25)^2 + (y - 25)^2}$. When the water starts to drain at $t = 0$, the circular dam-break bore waves spread and propagate radially and symmetrically. The final solution is plotted at $t = 0.69s$ when the second-order accurate schemes are employed. In figures 7-8, we compare the first-order two- and multi-point schemes on a structured quadrangular 100×100 grid and on 23736 triangular cells respectively. In particular, on the right, we insert the 3D view of the water height h, while its contours from $h = 1$m to $h = 10$m are illustrated on the right. For both schemes and both meshes, the waves propagate uniformly and symmetrically. The two-point scheme appears to be slightly less diffusive than the multi-point one. This has already been observed in [36] for the Euler equations. Analogous results have been found with the second-order version of the scheme. We directly insert the comparison between the second- and first-order multi-point schemes in figure-9 using the same triangular grid. Both schemes provide correct results, and the second-order solution is much less diffusive than the first-order one, as expected.



(a) Time $t = 0$s

(b) Time $t = 0.5$s

(c) Time $t = 1$s

(d) Time $t \approx 55.45$s

Figure 6: Perturbation of stationary solution. Free surface and iso-lines computed at times $t = 0$s, $t = 0.5$s, $1t = 1$s, $t \approx 55.45$s, and with first-order multi-point scheme on a 100×100 square grid.

5.3.1. Radial dam break over non-flat topography

Let us now repeat the test from the previous section but with non-flat topography. We take inspiration from [2] for the topography definition, that now reads

$$B(x, y) = \frac{1}{2}\left(1 + \cos\left(\frac{2\pi r}{2}\right)\right),$$

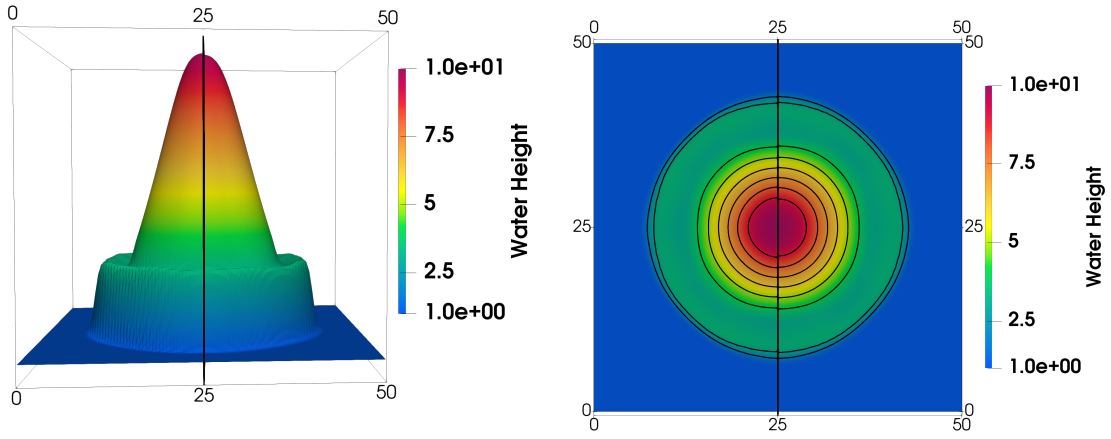
while we set the water height to

$$h(x, y, t = 0) = \begin{cases} 11 - B(x, y) & \text{if} \quad r \leq r_0, \\ 2 - B(x, y) & \text{otherwise.} \end{cases}$$

In figure 10, we directly present the results for the first-order two- and multi-point schemes with triangular mesh. We draw the same conclusions as in the previous section even if now the topography is non-flat: the numerical schemes are able to correctly reproduce the wave propagation. Once again, the two-point scheme is slightly less diffusive than the multi-point one.

5.4. Oblique hydraulic jump

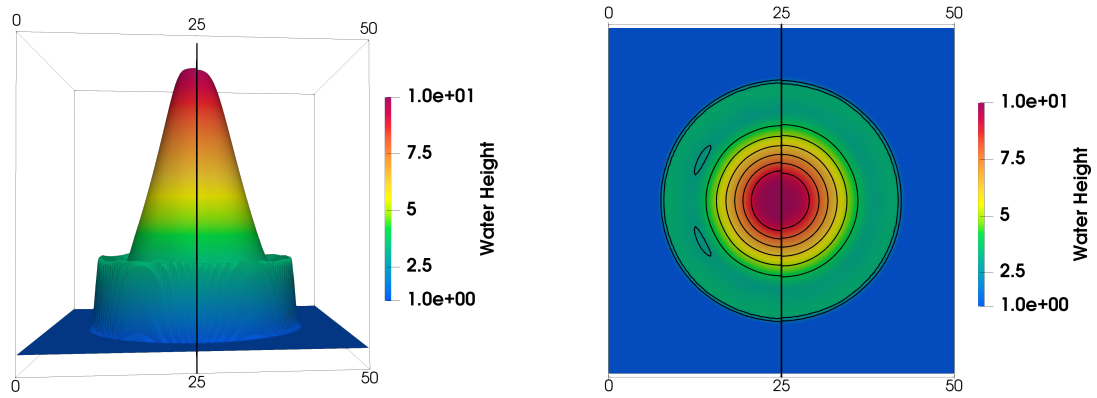
The oblique hydraulic jump is induced by means of an interaction between super-critical flow and a converging wall deflected through an angle $\theta = 8.95°$, see [28]. The computational domain



(a) 3D view of water height h.

(b) 10 iso-lines of 1m $< h <$ 10m.

Figure 7: Circular dam break. Water height at $t = 0.69s$ of the first-order two-point (left) and multi-point (right) schemes on 100×100 grid.



(a) 3D view of water height h.

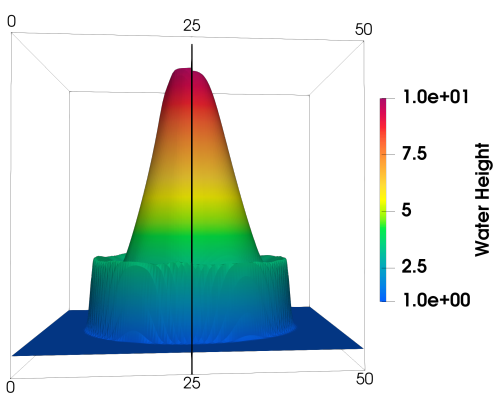
(b) 10 iso-lines of 1m $< h <$ 10m.

Figure 8: Circular dam break. Water height at $t = 0.69s$ of the first-order two-point (left) and multi-point (right) schemes on 23736 triangular cells.

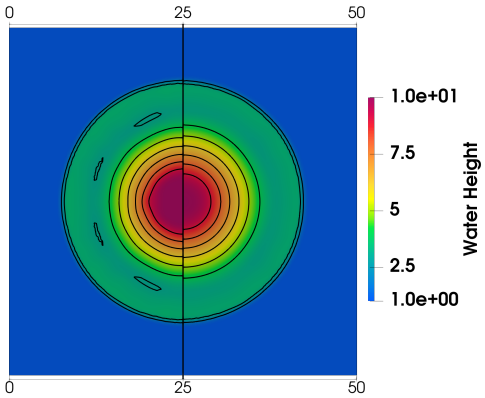
is set to $\omega \in [0, 40\text{m}] \times [0, 30\text{m}]$. The initial conditions is a uniform flow given by

$$h = 1\text{m},\ u = 8.57\text{m/s},\ v = 0\text{m/s}.$$

The boundary conditions are set super-critical flows upstream and downstream. The exact solution is given by $h = 1.5$m, $u = 7.9556$m/s for the downstream flow and an angle of 30° for the angle of the jump connecting down to upstream flow. The final time is set to $t = 10$s and the mesh is made of $N = 11892$ triangles and the results are gathered in figure 11. In figure 11a we present the numerical water height obtained with the two- and multi-point schemes (top and bottom respectively) in color and for 5 isolines (1m $< h <$ 1.5m). Next in figure 11b are compared the water height along the line $y = 40$m with respect to the exact solution showing that the correct

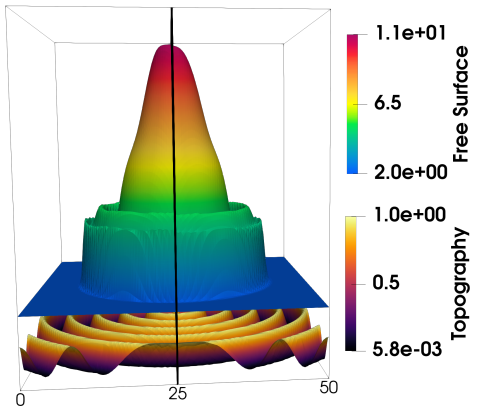


(a) 3D view of water height h.

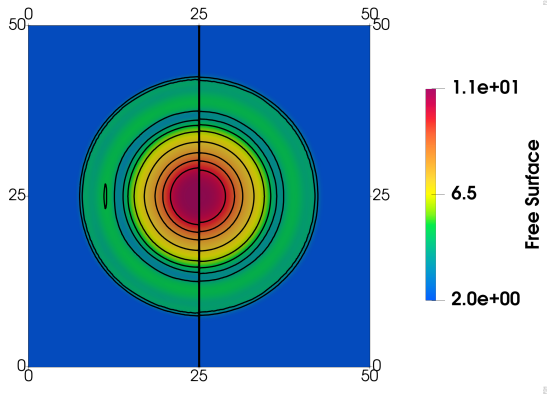


(b) 10 iso-lines of 1m $< h <$ 10m.

Figure 9: Circular dam break. Water height at $t = 0.69s$ of the second-order (left) and first-order (right) multi-point schemes on 23736 triangular cells.



(a) 3D view of free surface $h + B$ and topography B.



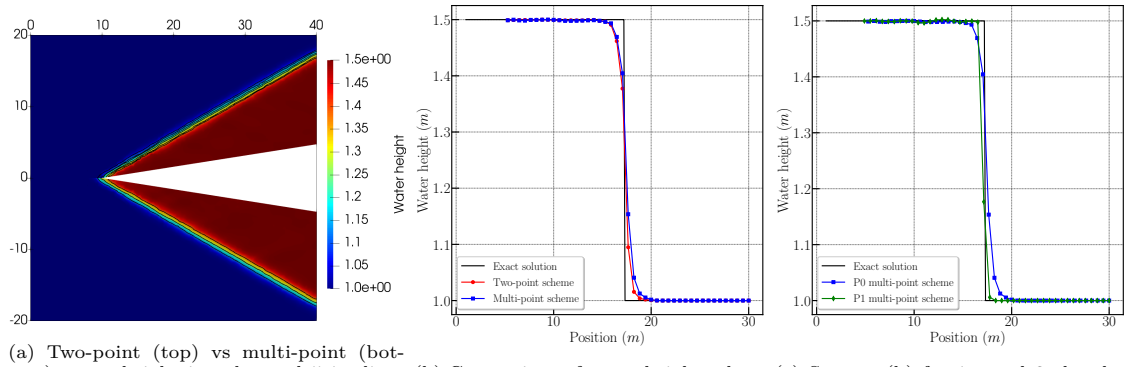
(b) 10 iso-lines of 2m $< h + B <$ 11m.

Figure 10: Circular dam break over non-flat topography. Free surface at $t = 0.69s$ and topography of the first-order two- (left) and multi-point (right) schemes on 23736 triangular cells.

solution is captured by both schemes while the two-point seems slightly less dissipative. At last, in figure 11c we present the comparison between the first and second order scheme showing an increase in the accuracy with which the discontinuity is captured.

5.5. 2D Riemann problem

In this section, we present a 2D Riemann problem with a complex wave structure. For the complete explanation of this test, we directly refer to [51]. Given a rectangular domain of size $[0, 100\text{m}] \times [0, 100\text{m}]$, we consider it as divided in four quadrants of the same size: Q_1 (up-right), Q_2 (up-left), Q_3 (bottom-left) and Q_4 (bottom-right). Then, the topography and initial conditions



(a) Two-point (top) vs multi-point (bottom) water height in color and 5 iso-lines ($1 < h < 1.5$).

(b) Comparison of water heights along $y = 40$m for both schemes.

(c) Same as(b) for 1st and 2nd order accurate multi-point schemes.

Figure 11: Oblique hydraulic jump at $t = 10s$ on 11892 unstructured grids.

read

$$h(x,y,t=0) = \begin{cases} 10 & \text{if} \quad (x,y) \in Q_1 \\ 3 & \text{if} \quad (x,y) \in Q_2 \\ 0.2 & \text{if} \quad (x,y) \in Q_3 \\ 3 & \text{if} \quad (x,y) \in Q_4 \end{cases}, \quad B(x,y) = \begin{cases} 1 & \text{if} \quad (x,y) \in Q_1 \\ 0 & \text{if} \quad (x,y) \in Q_2 \\ -0.5 & \text{if} \quad (x,y) \in Q_3 \\ 0 & \text{if} \quad (x,y) \in Q_4 \end{cases},$$

$$u(x,y,t=0) = \begin{cases} 0 & \text{if} \quad (x,y) \in Q_1 \\ 3 & \text{if} \quad (x,y) \in Q_2 \\ 3 & \text{if} \quad (x,y) \in Q_3 \\ 0 & \text{if} \quad (x,y) \in Q_4 \end{cases}, \quad v(x,y,t=0) = \begin{cases} 0 & \text{if} \quad (x,y) \in Q_1 \\ 0 & \text{if} \quad (x,y) \in Q_2 \\ 3 & \text{if} \quad (x,y) \in Q_3 \\ 3 & \text{if} \quad (x,y) \in Q_4 \end{cases},$$

The boundary conditions are outflow and the ending time is $t = 3$s.

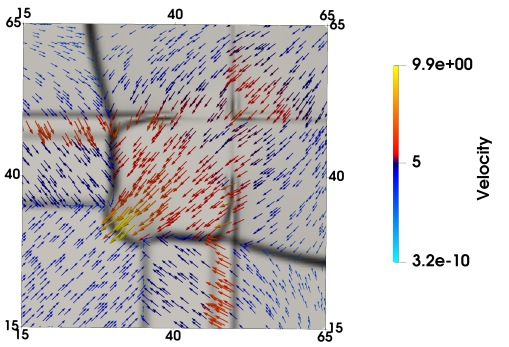
Results are given in figures 12-13. In the former, we consider a regular quadrangular 100×100 mesh and we insert the results for the first- and second-order multi-point scheme. In figure 13, we compare two- and multi-point first-order schemes, second-order multi-point scheme and a reference solution computed with the second-order method and 800×800 cells. The results are generally considered satisfying and similar conclusions to the ones of the previous section are drawn. Specifically, the methods are capable of reproducing correct solutions even in presence of a complex variety of waves and discontinuous topography.

5.6. Colliding flow

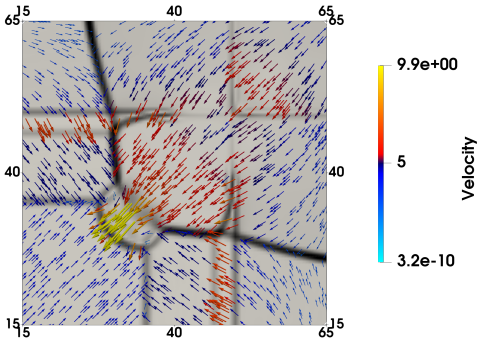
Next, we test the sensibility of the numerical schemes towards shock instabilities, i.e. carbuncle phenomenon. This test case is a colliding flow on a flat bottom, see [30]. The computational domain is $\Omega \in [-2.5\text{m}, 2.5\text{m}] \times [-2.5\text{m}, 2.5\text{m}]$, and we set the gravitational constant to $g = 1$. The initial condition is given as

$$h(x,y,t=0) \equiv 1\text{m}, \quad u(x,y,t=0) = \begin{cases} 30\text{m/s} & \text{if} \quad x \leq 0, \\ -30\text{m/s} & \text{if} \quad \text{else.} \end{cases}, \quad v(x,y,t=0) \equiv 0\text{m/s}.$$

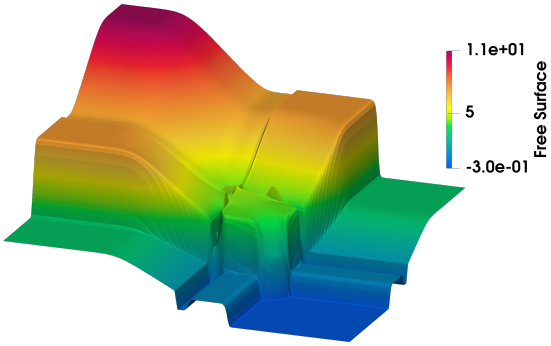
A small perturbation of the form $\delta(2z-1)$, where z is a random number with $z \in [-1,1]$, is introduced to the initial conservative state variables with $\delta = 10^{-5}$ in order to trigger instability. The domain is paved by 50×50 uniform quadrangular cells. Since the test is essentially 1D in x direction, this perturbation should be dissipated by the numerical schemes, especially the first



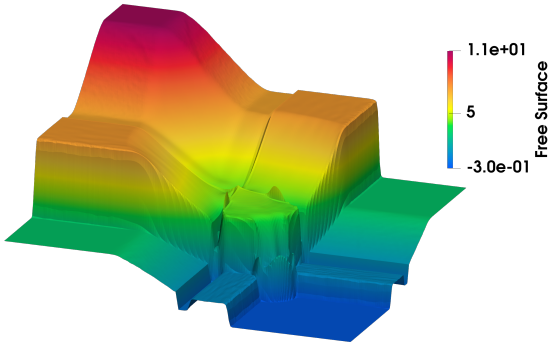
(a) First-order scheme.



(b) Second-order scheme.



(c) First-order scheme.



(d) Second-order scheme.

Figure 12: 2D Riemann problem. Gradient of the free surface $h + B$ together with the velocity vector field in the up images. Overview of the free surface elevation in the bottom images. Results obtained with first- (left) and second-order (right) multi-point method.

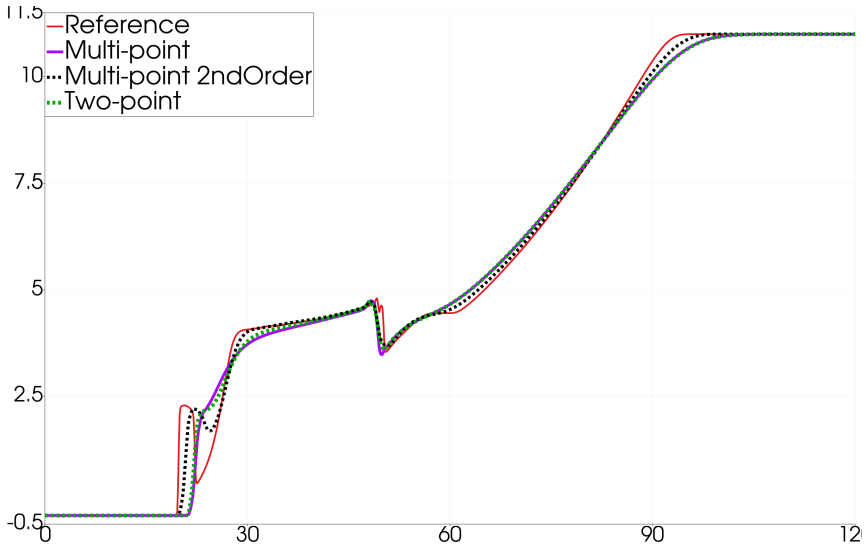


Figure 13: 2D Riemann problem. Cross-sectional free surface in the diagonal direction. Comparison among first-order two- (dashed green line) and multi- (violet line) point schemes, second-order multi-point scheme (dashed black line) and reference solution (red).

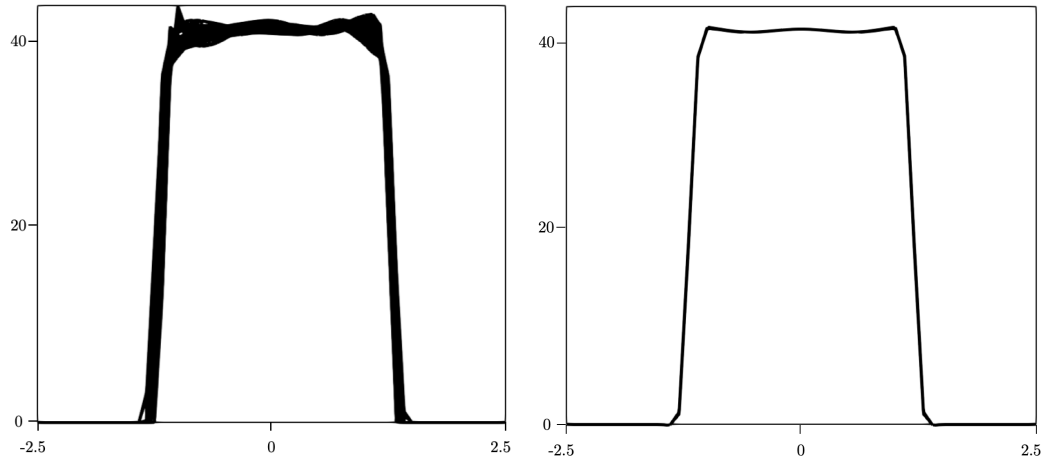
order versions.
In figure 14, we present the scatter plot of the water height h, at time $t = 2$s for both the two-point and multi-point schemes. We observe that the two-point scheme produces spurious instabilities, whereas the multi-point scheme displays a symmetrical solution, meaning that the seeded instability has been dissipated as expected from a robust numerical scheme. This spurious instability is referred to as the 'carbuncle effect' [30]. The multi-point scheme seems insensitive to this.

5.7. Odd-even decoupling

Another test case to study the carbuncle phenomenon is the moving shock problem. This problem is an analogy from the classical Quirk test [55]. The canal with a length of $L = 800$m and width of 20m is initially at rest with $h(x, y, t = 0) = 1 - B(x, y), u(x, y, t = 0) \equiv 0, v(x, y, t = 0) \equiv 0$ and

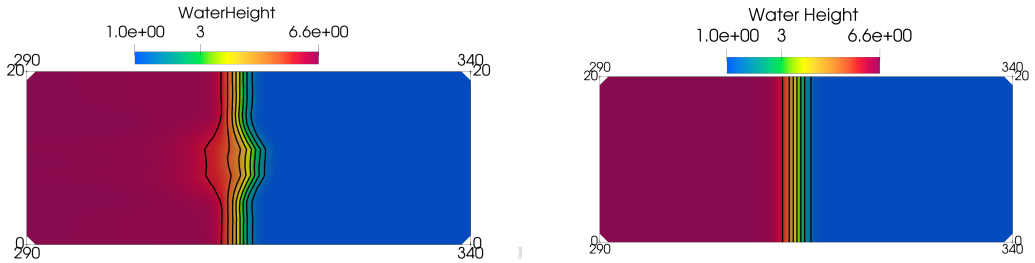
$$B(x,y) = \begin{cases} \frac{1+0.02\cos(\pi(x-400))}{2} & \text{if} \quad 350 \leq x \leq 450, \\ 0 & \text{otherwise.} \end{cases}$$

At the inlet, we impose $h_{in} = 6.588$m and $u_{in} = 13.283\text{m}^2/\text{s}$. We discretize the domain by 800×20 quadrangular cells. A small perturbation of order 10^{-3} is introduced in the horizontal central line of the mesh grid to trigger an instability, see also [36]. In figures 15-16, we show the results for the water height (together with its iso-lines) at time $t = 20$s and $t = 30$s respectively, namely before and after the shock wave has reached the topography bump. We insert the outputs of the first-order two- and multi-point schemes on the left and the right side of the figure respectively. For the former method, we observe that spurious instabilities are present at the two different times, while the multi-point scheme appears to correctly reproduce the shock wave, no Carbuncle effect is present. Furthermore, the second-order extension of the multi-point method does not present any spurious oscillations as well. We do not insert the results here as they are analogous (even if less diffusive).



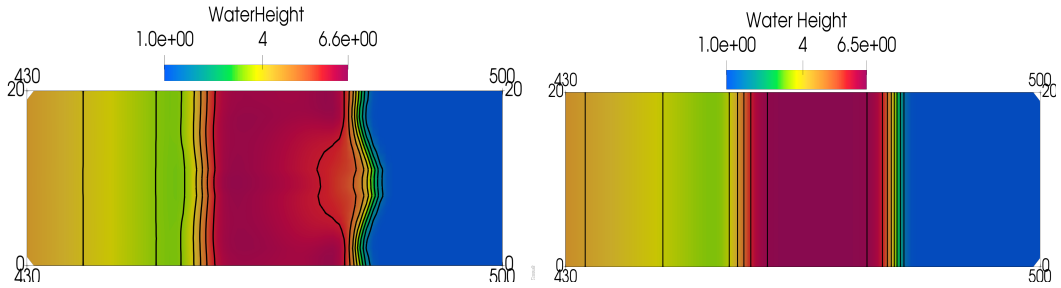
(a) Two-point scheme. (b) Multi-point scheme.

Figure 14: Colliding flow. Numerical water height at $t = 2$s for the two-point (left) and multi-point (right) schemes and a 50×50 quad grid.



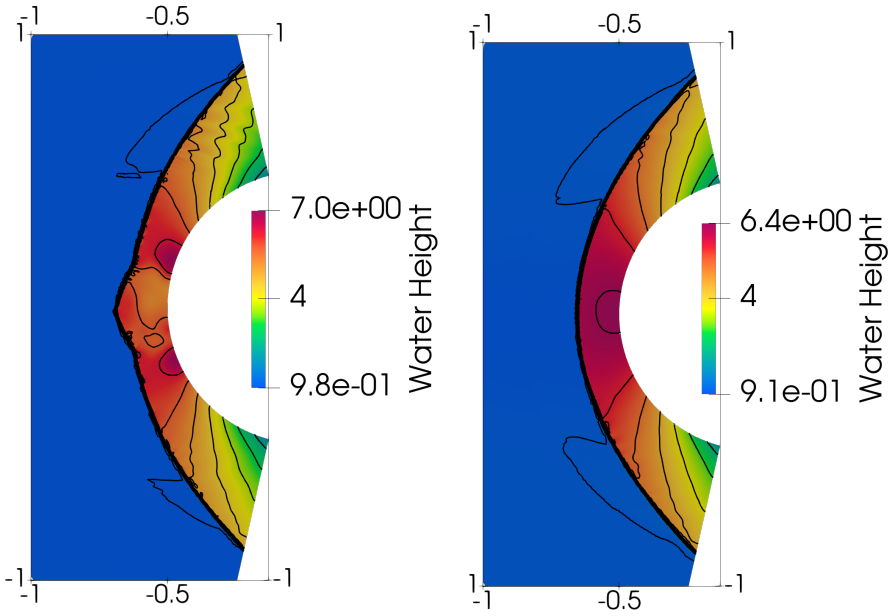
(a) Two-point scheme. (b) Multi-point scheme.

Figure 15: Odd-even decoupling test. Water height h at time $t = 20$s. Results obtained with the first-order two- (left) and multi-point (right) method.



(a) Two-point scheme. (b) Multi-point scheme.

Figure 16: Odd-even decoupling test. Water height h at time $t = 30$s. Results obtained with the first-order two- (left) and multi-point (right) method.



(a) Two-point scheme.

(b) Multi-point scheme.

Figure 17: Flow past a cylinder. Water height h obtained with first-order two-point (left) and multi-point (right) method.

5.8. Flow past a cylinder

Let us now consider another problem that is well-known to give rise to the Carbuncle effect in gas dynamics: the flow past a cylinder [36]. In case of shallow water flows, the cylinder could be a pier, for instance of a bridge [42]. Hence, in the domain, we insert a cylinder of radius 0.5m. The topography and initial conditions are given by $h(x, y, t = 0) = 1 + B_{max} - B(x, y)$, $u(x, y, t = 0) = 5\sqrt{gh(x, y, t = 0)}$ and $v(x, y, t = 0) \equiv 0$, where the topography reads

$$B(x, y) = \begin{cases} \dfrac{0.3(0.75 - r^2)^{0.3}}{0.5} & \text{if} \quad r^2 \leq 0.75 \\ 0 & \text{otherwise} \end{cases}$$

with $r = \sqrt{x^2 + y^2}$ and, in particular, $B_{max} = \max_{x,y}(B(x, y))$. Regarding the boundary conditions, we impose a wall-type one at the cylinder surface, while free boundary conditions are considered at the bottom and up-side of the domain. Then, at the left boundary, we impose an inflow condition: $h_{\text{inf}} = 1$ and $u_{\text{inf}} = 5\sqrt{gh_{\text{inf}}}$. We use an unstructured triangular mesh of 81272 cells and insert the results in figure 17. Since the numerical solution should converge to a steady state, the simulation is stopped if the residuals are smaller than 10^{-6}, otherwise we use $t = 200$s as ending time. On the left, we insert the results for two-point scheme, while the multi-point method outputs are on the right. We immediately see that the Carbuncle phenomenon is present in the two-point solution, while the multi-point one is correctly computed. Once again, the second-order extension of the multi-point scheme does not present the Carbuncle effect as its first-order version.

6. Conclusions and perspectives

In this paper, we presented an original numerical scheme to deal with shallow water balance laws in 2D. One of the characteristics of this numerical method is its direct link to its Lagrangian counterpart. Indeed, if a mapping Lagrange to Euler exists, then it is reasonable to assume that this link must be preserved by the numerical method. Bearing this in mind, we have developed a Lagrangian Riemann solver to deal with balance laws, directly inspired by its version for gas dynamics developed in [32, 35, 15, 36]. This Lagrangian Riemann Solver (RS) and its link to the Eulerian framework allows to ensure, -a- an appropriate wave speeds ordering, -b- the positivity preservation, and -c- an entropy inequality. It is also constructed to assure that the steady-state solutions (lake at rest) are exactly retrieved by the RS, in other words the RS is well-balanced. Using this RS as a building block, we designed an un-classical Finite Volume (FV) scheme. The idea of this method is, *de facto*, to involve all surrounding cells in the update of the current one. This feature is obtained thanks to a so-called nodal solver first introduced for gas dynamics Lagrangian schemes [48]. Without this feature, the numerical scheme resembles a classical HLLC-like two-point flux FV scheme, even if with different wave speeds' definition. Due to the nodal solver, this FV scheme is not conservative in the classical way, meaning that the conservation is retrieved on a node-based fashion. The method is also proven to be well-balanced for lake at rest steady-state solutions. A classical extension to second order of accuracy in space and time is also proposed.
Several numerical results are presented to assess the behavior of this numerical method in its classical (two-point flux) and current (multi-point flux) version. The multi-point flux scheme seems insensitive to the so-called carbuncle effect. As expected the second order version produces more accurate, still robust, positive and stable results.
This work has been developed under the fundamental hypothesis that the domain is, and remains, wet. A possible improvement would be to treat wet-dry transition area, which would demand some technical adjustments on the implementation level. A second line of improvement is the extension to higher orders of accuracy using polynomial reconstructions and *a posteriori* MOOD type limiting procedure [16, 17, 15]. Then, regarding the well-balanced property of the method, we could test if the RS designed for the homogeneous case (of isentropic Euler equation) could be coupled with so-called 'reconstruction' technique to deal with the source term as in [12, 13].
Finally, it should be pointed out that the numerical scheme proposed here could easily be extended to other hyperbolic system of balance laws as the Euler equations with gravity source term [34] and the magnetohydrodynamic equations [54].

Acknowledgments

We would like to thank Pr. G. Russo, Ass. Pr S.Boscharino, E. Macca who have welcomed us in Catania, Italy for a scientific refreshing week in October 2020 which partly led to this work.
This work has been undertaken in the framework of the jointed Laboratory ANABASE (**A**nalyse **N**umérique **A**ppliqué de **B**ordeaux **A** la **S**imulation d'**E**coulements complexes) gathering scientists from CEA, CNRS, Bordeaux INP and Bordeaux University working on the development of innovative numerical methods for complex fluid flows.

References

[1] E. Audusse, F. Bouchut, M.-O. Bristeau, R. Klein, and B. Perthame. A fast and stable well-balanced scheme with hydrostatic reconstruction for shallow water flows. *SIAM J. Sci. Comput.*, 25:2050–2065, 2004. (Cited page 2.)

[2] F. Aureli, A. Maranzoni, P. Mignosa, and C. Ziveri. A weighted surface-depth gradient method for the numerical integration of the 2d shallow water equations with topography. *Advances in Water Resources*, 31(7):962–974, 2008. (Cited page 30.)
[3] Wendroff B. Two-dimensional HLLE Riemann solver and associated Godunov-type difference scheme for gas dynamics. *Computers and Mathematics with Applications*, 38(11):175 – 185, 1999. (Cited page 2.)
[4] A. Bermúdez and M. E. Vázquez. Upwind methods for hyperbolic conservation laws with source terms. *Computers & Fluids*, 23(8):1049–1071, 1994. (Cited page 2.)
[5] A. Bermúdez and M.E. Vazquez. Upwind methods for hyperbolic conservation laws with source terms. *Computers & Fluids*, 23:1049–1071, 1994. (Cited page 2.)
[6] C. Berthon and C. Chalons. A fully well-balanced, positive and entropy-satisfying Godunov-type method for the shallow-water equations. *Mathematics of Computation*, 85(299):1281–1307, 2016. (Cited page 2.)
[7] C. Berthon, A. Duran, F. Foucher, K. Saleh, and J. Zabsonré. Improvement of the hydrostatic reconstruction scheme to get fully discrete entropy inequalities. *Journal of Scientific Computing*, pages 1–33, 2019. (Cited page 2.)
[8] C. Berthon and F. Foucher. Efficient well-balanced hydrostatic upwind schemes for shallow-water equations. *J. Comput. Phys.*, 231:4993–5015, 2012. (Cited page 2.)
[9] F. Bouchut. *Nonlinear Stability of Finite Volume Methods for Hyperbolic Conservation Laws*. Birkhauser, first edition, 2004. (Cited page 2, 5 et 22.)
[10] A. Canestrelli, A. Siviglia, M. Dumbser, and E. F. Toro. Well-balanced high-order centred schemes for non-conservative hyperbolic systems. Applications to shallow water equations with fixed and mobile bed. *Advances in Water Resources*, 32:834–844, 2009. (Cited page 2.)
[11] M. J. Castro and C. Parés. Well-balanced high-order finite volume methods for systems of balance laws. *J. Sci. Comput.*, 82(2), February 2020. (Cited page 2.)
[12] M.J. Castro, J.A. López-García, and Carlos Parés. High order exactly well-balanced numerical methods for shallow water systems. *Journal of Computational Physics*, 246:242–264, 2013. (Cited page 38.)
[13] M.J. Castro, T. Morales de Luna, and C. Parés. Chapter 6 - Well-balanced schemes and path-conservative numerical methods. In Rémi Abgrall and Chi-Wang Shu, editors, *Handbook of Numerical Methods for Hyperbolic Problems*, volume 18 of *Handbook of Numerical Analysis*, pages 131–175. Elsevier, 2017. (Cited page 2, 5 et 38.)
[14] C. Chalons and A. Del Grosso. Exploring different possibilities for second-order well-balanced lagrange-projection numerical schemes applied to shallow water Exner equations. *International Journal for Numerical Methods in Fluids*, 94:505 – 535, 2022. (Cited page 2 et 27.)
[15] A. Chan, G. Gallice, R. Loubère, and P.-H. Maire. Positivity preserving and entropy consistent approximate riemann solvers dedicated to the high-order MOOD-based finite volume discretization of Lagrangian and Eulerian gas dynamics. *Computers & Fluids*, 229:105056, 2021. (Cited page 2, 6, 14, 15 et 38.)
[16] S. Clain, S. Diot, and R. Loubère. A high-order finite volume method for systems of conservation laws – multi-dimensional optimal order detection (MOOD). *J. Comput. Phys.*, 230(10):4028 – 4050, 2011. (Cited page 38.)
[17] S. Clain and J. Figueiredo. The MOOD method for the non-conservative shallow-water system. *Computers & Fluids*, 145:99–128, 2017. (Cited page 38.)
[18] Adhémar Jean Claude Barré de Saint-Venant. Théorie du mouvement non permanent des eaux, avec application aux crues des rivières et a l'introduction de marées dans leurs lits. *Comptes rendus hebdomadaires des séances de l'Académie des sciences*, 73:147–154, 237–240, 1871. (Cited page 3.)
[19] A. Del Grosso. *Numerical simulation of geophysical flows using high-order and well-balanced Lagrange-Projection methods*. Theses, Université Paris-Saclay, September 2022. (Cited page 2, 7 et 17.)
[20] B. Després and C. Mazeran. Lagrangian gas dynamics in two dimensions and Lagrangian systems. *Arch. Rational Mech. Anal.*, 178:327–372, 2005. (Cited page 3.)
[21] V. Desveaux, M. Zenk, C. Berthon, and C. Klingenberg. A well-balanced scheme to capture non-explicit steady states in the Euler equations with gravity. *International Journal for Numerical Methods in Fluids*, 81:104 – 127, 2016. (Cited page 2.)
[22] Balsara D.S. Multidimensional HLLE riemann solver: Application to euler and magnetohydrodynamic flows. *Journal of Computational Physics*, 229(6):1970 – 1993, 2010. (Cited page 2.)
[23] Balsara D.S. A two-dimensional HLLC riemann solver for conservation laws: Application to euler and magnetohydrodynamic flows. *Journal of Computational Physics*, 231(22):7476 – 7503, 2012. (Cited page 2.)
[24] Balsara D.S. Multidimensional riemann problem with self-similar internal structure. Part i - Application to hyperbolic conservation laws on structured meshes. *Journal of Computational Physics*, 277:163 – 200, 2014. (Cited page 2.)
[25] Balsara D.S. and Nkonga B. Multidimensional riemann problem with self-similar internal structure – part iii – a multidimensional analogue of the HLLI riemann solver for conservative hyperbolic systems. *Journal of Computational Physics*, 346:25 – 48, 2017. (Cited page 2.)
[26] Balsara D.S. and Dumbser M. Multidimensional riemann problem with self-similar internal structure. Part ii - Application to hyperbolic conservation laws on unstructured meshes. *Journal of Computational Physics*,

287:269 – 292, 2015. (Cited page 2 et 6.)
[27] Balsara D.S., Dumbser M., and Abgrall R. Multidimensional HLLC riemann solver for unstructured meshes - with application to euler and MHD flows. *Journal of Computational Physics*, 261:172 – 208, 2014. (Cited page 2.)
[28] P. Garcia-Navarro F. Alcrudo. A high resolution Godunov-type scheme in finite volumes for the 2d shallow water equations. *Int. J. Numer. Methods Fluids*, 16:489–505, 1993. (Cited page 27, 29 et 30.)
[29] N. Fleischmann, S. Adami, X. Hu, and N. Adams. A low dissipation method to cure the grid-aligned shock instability. *Journal of Computational Physics*, 401:109004, 10 2019. (Cited page 3.)
[30] K. Friedemann. Heuristical and numerical considerations for the carbuncle phenomenon. *Appl. Math. Comput.*, 320:596–613, 2018. (Cited page 3, 33 et 35.)
[31] F. Kemm G. Bader. The carbuncle phenomenon in shallow water simulations. In *2nd International Conference on Computational Science and Engineering*, August 2014. (Cited page 3 et 27.)
[32] G. Gallice. Schémas de type Godunov entropiques et positifs préservant les discontinuités de contact. *CRAS*, 1(331):149–152, 2000. (Cited page 2, 5, 14 et 38.)
[33] G. Gallice. Approximation numérique de systèmes hyperboliques non-linéaires conservatifs ou non-conservatifs. 2002. Habilitation à diriger des recherches. (Cited page 2, 5, 7, 14 et 17.)
[34] G. Gallice. Solveurs simples positifs et entropiques pour les systèmes hyperboliques avec terme source. *Comptes Rendus Mathematique*, 334(8):713–716, 2002. (Cited page 2, 6, 11, 12, 14 et 38.)
[35] G. Gallice. Positive and entropy stable godunov-type schemes for gas dynamics and MHD equations in Lagrangian or Eulerian coordinates. *Numer. Math.*, 94(4):673–713, June 2003. (Cited page 2, 14 et 38.)
[36] G. Gallice, A. Chan, R. Loubère, and P.-H. Maire. Entropy stable and positivity preserving Godunov-type schemes for multidimensional hyperbolic systems on unstructured grid. *Journal of Computational Physics*, 468, 2022. (Cited page 2, 6, 8, 9, 14, 15, 17, 19, 20, 23, 29, 35, 37 et 38.)
[37] S.K. Godunov, A. Zabrodine, M. Ivanov, A. Kraiko, and G. Prokopov. *Résolution numérique des problèmes multidimensionnels de la dynamique des gaz*. Mir, 1979. (Cited page 2.)
[38] A. Del Grosso, M. J. Castro Díaz, C. Chalons, and T. Morales de Luna. On well-balanced implicit-explicit Lagrange-projection schemes for two-layer shallow water equations. *Appl. Math. Comput.*, 442:127702, 2023. (Cited page 2.)
[39] V. Guinot. *Wave propagation in fluids*. Iste. ISTE Ltd and John Wiley & Sons, London, England, 2 edition, September 2010. (Cited page 2.)
[40] Vides J., Nkonga B., and Audit E. A simple two-dimensional extension of the HLL riemann solver for hyperbolic systems of conservation laws. *Journal of Computational Physics*, 280:643 – 675, 2015. (Cited page 2.)
[41] Gallardo J.M., Schneider K.A., and Castro M.J. On a class of two-dimensional incomplete riemann solvers. *Journal of Computational Physics*, 386:541 – 567, 2019. (Cited page 2 et 3.)
[42] F. Kemm. A note on the carbuncle in shallow water simulations. *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik*, 94, 06 2014. (Cited page 3, 27 et 37.)
[43] D. Ketcheson and M. Quezada de Luna. Numerical simulation and entropy dissipative cure of the carbuncle instability for the shallow water circular hydraulic jump. *International Journal for Numerical Methods in Fluids*, 94, 01 2022. (Cited page 3.)
[44] R. J. LeVeque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge Texts in Applied Mathematics. Cambridge University Press, 2002. (Cited page 26.)
[45] Ching Loh and Philip Jorgenson. Multi-dimensional dissipation for cure of pathological behaviors of upwind scheme. *J. Comput. Physics*, 228:1343–1346, 03 2009. (Cited page 3.)
[46] Brio M., Zakharian A.R., and Webb G.M. Two-dimensional Riemann solver for Euler equations of gas dynamics. *Journal of Computational Physics*, 167(1):177 – 195, 2001. (Cited page 2.)
[47] Dudzinski M. and Lukáčová-Medvid'ová M. Well-balanced bicharacteristic-based scheme for multilayer shallow water flows including wet/dry fronts. *Journal of Computational Physics*, 235:82 – 113, 2013. (Cited page 2.)
[48] P.-H. Maire. A high-order cell-centered Lagrangian scheme for two-dimensional compressible fluid flows on unstructured meshes. *J. Comput. Phys.*, 228:2391–2425, 2009. (Cited page 17 et 38.)
[49] P.-H. Maire. Contribution to the numerical modeling of inertial confinement fusion. 2011. (Cited page 6.)
[50] P.-H. Maire, R. Abgrall, J. Breil, and J. Ovadia. A cell-centered Lagrangian scheme for compressible flow problems. *SIAM J. Sci. Comput.*, 29(4):1781–1824, 2007. (Cited page 3 et 6.)
[51] A. Navas-Montilla and J. Murillo. 2d well-balanced augmented ADER schemes for the shallow water equations with bed elevation and extension to the rotating frame. *Journal of Computational Physics*, 372:316–348, 2018. (Cited page 27 et 32.)
[52] C. Parés and M. J. Castro. On the well-balance property of Roe's method for nonconservative hyperbolic systems. Applications to shallow-water systems. *ESAIM: mathematical modelling and numerical analysis*, 38(5), 2004. (Cited page 2.)
[53] K. Peery and S. Imlay. Blunt-body flow simulations. In *24th Joint Propulsion Conference*. American Institute of Aeronautics and Astronautics, July 1988. (Cited page 3.)
[54] Kenneth G. Powell. *An Approximate Riemann Solver for Magnetohydrodynamics*, pages 570–583. Springer Berlin Heidelberg, Berlin, Heidelberg, 1997. (Cited page 38.)

[55] J. J. Quirk. A contribution to the great Riemann solver debate. *International Journal for Numerical Methods in Fluids*, 18:555–574, 1994. (Cited page 3, 27 et 35.)
[56] LeVeque R.J. Wave propagation algorithms for multidimensional hyperbolic systems. *Journal of Computational Physics*, 131(2):327 – 353, 1997. (Cited page 2.)
[57] A.V. Rodionov. Artificial viscosity Godunov-type schemes to cure the carbuncle phenomenon. *J. Comput. Phys.*, 345:308–329, 2017. (Cited page 3.)
[58] P L Roe. Upwind differencing schemes for hyperbolic conservation laws with source terms. In *Lecture Notes in Mathematics*, Lecture notes in mathematics, pages 41–51. Springer Berlin Heidelberg, Berlin, Heidelberg, 1987. (Cited page 2.)
[59] R. Sanders, E. Morano, and M.-C. Druguet. Multidimensional dissipation for upwind schemes: Stability and applications to gas dynamics. *Journal of Computational Physics*, 145:511–537, 09 1998. (Cited page 3.)
[60] K. A. Schneider, J. M. Gallardo, D. S. Balsara, B. Nkonga, and C. Parés. Multidimensional approximate riemann solvers for hyperbolic nonconservative systems. Applications to shallow water systems. *Journal of Computational Physics*, 444:110547, 2021. (Cited page 2 et 3.)
[61] Z. Shen, W. Yan, and G. Yuan. A robust and contact resolving Riemann solver on unstructured mesh, Part I, Euler method. *J. Comput. Phys.*, 268:432–455, 2014. (Cited page 17.)
[62] E. F. Toro. *Shock-capturing methods for free-surface shallow flows*. John Wiley & Sons, Chichester, England, February 2001. (Cited page 2.)
[63] E. F. Toro and P. García-Navarro. Godunov-type methods for free-surface shallow flows: A review. *Journal of Hydraulic Research*, 45:736 – 751, 2007. (Cited page 2.)
[64] E.F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics*. Springer, second edition, 1999. (Cited page 2 et 16.)
[65] B. van Leer. *The Development of Numerical Fluid Mechanics and Aerodynamics since the 1960's: US and Canada*, volume 100, pages 159–185. 05 2009. (Cited page 3.)
[66] V. Venkatakrishnan. Convergence to steady state solutions of the Euler equations on unstructured grids with limiters. *Journal of Computational Physics*, 118(1):120–130, 1995. (Cited page 26.)