INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2007; **53**:1509–1540
Published online 27 September 2006 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1374

A general approximate-state Riemann solver for hyperbolic systems of conservation laws with source terms

Julien Lhomme*,† and Vincent Guinot‡

Hydrosciences Montpellier, UMR 5569, Université Montpellier 2, Maison des Sciences de l'Eau – MSE, 34095 Montpellier Cedex 5, France

SUMMARY

An approximate-state Riemann solver for the solution of hyperbolic systems of conservation laws with source terms is proposed. The formulation is developed under the assumption that the solution is made of rarefaction waves. The solution is determined using the Riemann invariants expressed as functions of the components of the flux vector. This allows the flux vector to be computed directly at the interfaces between the computational cells. The contribution of the source term is taken into account in the governing equations for the Riemann invariants. An application to the water hammer equations and the shallow water equations shows that an appropriate expression of the pressure force at the interface allows the balance with the source terms to be preserved, thus ensuring consistency with the equations to be solved as well as a correct computation of steady-state flow configurations. Owing to the particular structure of the variable and flux vectors, the expressions of the fluxes are shown to coincide partly with those given by the HLL/HLLC solver. Computational examples show that the approximate-state solver yields more accurate solutions than the HLL solver in the presence of discontinuous solutions and arbitrary geometries.

Received 8 February 2006; Revised 13 July 2006; Accepted 13 August 2006

KEY WORDS: Godunov-type schemes; Riemann solvers; source terms; water hammer equations; shallow water equations

1. INTRODUCTION

Godunov-type schemes [1] aim to solve hyperbolic systems of conservation laws in the presence of discontinuous solutions [2–4]. The evolution of the solution in a given cell is determined by the exchanges (via fluxes) at the interfaces with the neighbouring cells. The fluxes are computed

*Correspondence to: Julien Lhomme, Hydrosciences Montpellier, UMR 5569, Université Montpellier 2, Maison des Sciences de l'Eau – MSE, 34095 Montpellier Cedex 5, France.
†E-mail: lhomme@msem.univ-montp2.fr
‡E-mail: guinot@msem.univ-montp2.fr

Contract/grant sponsor: French Ministry of Research; contract/grant number: 9884-2003

by solving Riemann problems at the interfaces. Computing an exact solution to the Riemann problem can be a very time-consuming task because an iterative procedure is needed. Therefore, approximate Riemann solvers are often preferred because they provide satisfactory solutions while using faster algorithms. Two broad families of solvers can be distinguished: (i) solvers where the Riemann problem is simplified (e.g. by linearizing the equations), and (ii) solvers where simplified relationships are used to solve the exact problem.

The first family of solver includes Roe's solver [5], where the flux at the location of the initial discontinuity is calculated via a wave decomposition under the assumption of a constant Jacobian matrix. The Jacobian matrix is approximated in such a way that consistency and conservation conditions are satisfied. An entropy fix is needed when a rarefaction wave extends over the location of the initial discontinuity. Primitive variable Riemann solvers [2] use a linearization of the hyperbolic system with a constant Jacobian matrix in combination with the Rankine–Hugoniot conditions across each wave. This allows a simplified system of equations to be solved for the unknown variables. The Riemann invariants can also be used along the characteristics to obtain the simplified system. A hybrid approach, combining simplified Riemann solvers used in the regions where the solution is smooth and more sophisticated solvers where the solution involves large gradients, has also been proposed [6].

The second family includes the HLL [7] solver for 2×2 systems of conservation laws, based on the assumption of two discontinuities, the propagation speeds of which are estimated *a priori* [8, 9]. Adding a third wave to the system leads to the HLLC solver [10] for an improved resolution of contact discontinuities when 3×3 systems are to be solved. Osher's solver [11] uses vector splitting in the integration of the flux vector in the phase space. The integration path is designed as a broken line, the segments of which are collinear to one of the eigenvectors of the Jacobian matrix. Linde [12] uses the HLLC framework with two intermediate states linked by a heuristic equation, allowing an estimate of the interface flux to be calculated. In another category of solvers, known as approximate-state solvers [13, 14] the same relationships are used indifferently for shock and rarefaction waves, as shock and rarefaction relationships differ only by a third-order term with respect to the conserved variable [15]. This is equivalent to assuming that all waves except contact discontinuities are: (i) rarefaction waves (in which case the Riemann invariants are used across the waves), or (ii) shock waves (in which case the Rankine–Hugoniot relationships are used). The performance of the above-mentioned solvers is analysed on a number of test cases in [16].

The above-mentioned solvers were initially designed for hyperbolic systems of conservation laws without source terms. Stability problems may arise when source terms are present in the conservation laws to be solved. This is the case in particular for the shallow-water equations in the presence of steep bottom slopes.

When the hyperbolic system to be solved includes source terms, the fractional step or time splitting method can be used, resulting in a two step algorithm. This method is generally successful except when the solution is close to a steady state [17].

The source term upwinding approach [18] consists in splitting the source term across the cell interfaces into two parts, to be assigned to the cells located on either sides of the interface. The splitting between the left- and right-hand source terms is determined from the wave celerities in the cells.

Well-Balanced schemes [19] incorporate the non-uniform topography into the Riemann problem with a partial differential equation with respects to time, considering the bottom level as part of the conserved variables vector. The new Riemann problem obtained was initially solved in an exact way [19, 20]. Approximate Riemann solvers were later introduced [21] to reduce the computational cost of the method.

DOI: 10.1002/fld

LeVeque [17] proposed that the variable vector should be split into two vectors with a discontinuity at the centre of the cell. The updated variables are computed using the fluxes at the cell interfaces, the fluxes at the centre of the cells, and the contribution from the source term. The resulting scheme, called the quasi-steady wave-propagation algorithm, is efficient when the system is close to steady state.

The present paper aims to propose an approximate-state solver for hyperbolic systems of conservation laws with source terms. This solver uses the Riemann invariants via a two rarefaction wave assumption. The Riemann invariants are expressed as functions of the components of the flux vector, which allows the flux to be computed directly at the interfaces between the computational cells.

Section 2 is devoted to the presentation of governing equations and solution principle. Section 3 details its application to the water hammer equations. Section 4 focuses on the shallow water equations. Section 5 presents computational examples where the proposed solver is compared to the HLL/HLLC solver, to which it is very similar. Section 6 provides concluding remarks.

2. GOVERNING EQUATIONS AND SOLUTION PRINCIPLE

2.1. General framework

The proposed solver aims to solve $m \times m$ hyperbolic systems of conservation laws that can be written in vector form as

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} = \mathbf{S} \tag{1}$$

where $\mathbf{U}$ is the conserved variable, $\mathbf{F}$ is the flux, $\mathbf{S}$ is the source term, and t and x are the time- and space-co-ordinates respectively. $\mathbf{F}$ is assumed to depend both on $\mathbf{U}$ and on the local value of one or several parameters, while $\mathbf{S}$ is assumed to depend on $\mathbf{U}$, the local value of and the gradient in the parameter(s)

$$\begin{aligned} \mathbf{F} &= \mathbf{F}(\mathbf{U}, \varphi) \\ \mathbf{S} &= \mathbf{S}(\mathbf{U}, \varphi, \partial\varphi/\partial x) \end{aligned} \tag{2}$$

where φ denotes the vector formed by the various parameters of the system of conservation laws. For instance, the shallow water equations in one dimension of space can be written in the form (1)–(2) by defining $\mathbf{U}$, $\mathbf{F}$ and $\mathbf{S}$ as

$$\mathbf{U} = \begin{bmatrix} h \\ hu \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} hu \\ hu^2 + gh^2/2 \end{bmatrix}, \quad \mathbf{S} = \begin{bmatrix} 0 \\ gh\partial z_\mathrm{b}/\partial x \end{bmatrix} \tag{3}$$

where g is the gravitational acceleration, h is the water depth, u is the flow velocity, and z_b is the bottom elevation. The parameter vector φ is defined as

$$\varphi = \begin{bmatrix} g \\ z_\mathrm{b} \end{bmatrix} \tag{4}$$

clearly, the flux $\mathbf{F}$ depends on φ (via g) and the source term depends on φ and $\partial\varphi/\partial x$ (via g and $\partial z_\mathrm{b}/\partial x$).

In what follows, the system of conservation laws (1) is assumed to be hyperbolic, that is, the Jacobian matrix of the flux $\mathbf{F}$ with respect to the conserved variable $\mathbf{U}$ has m real and distinct

DOI: 10.1002/fld

eigenvalues. Equation (1) is discretized in one dimension of space as (see Figure 1)

$$\mathbf{U}_i^{n+1} = \mathbf{U}_i^n + \frac{\Delta t}{\Delta x_i}(\mathbf{F}_{i-1/2}^{n+1/2} - \mathbf{F}_{i+1/2}^{n+1/2}) + (\mathbf{S}_{i-1/2,i}^{n+1/2} + \mathbf{S}_{i+1/2,i}^{n+1/2})\Delta t \tag{5}$$

where $\mathbf{U}_i^n$ is the average value of $\mathbf{U}$ over the cell i at the time level n, $\mathbf{F}_{i+1/2}^{n+1/2}$ is the average value of $\mathbf{F}$ between the time levels n and $n+1$ at the interface $(i+1/2)$ between the cells i and $i+1$, $\mathbf{S}_{i+1/2,i}^{n+1/2}$ is the contribution to the cell i of the source terms estimated from $\mathbf{U}$ and φ between the cells i and $i+1$, Δt is the computational time step and Δx_i is the size of the cell i. When the fluxes are estimated using an explicit approach (i.e. using only the values of $\mathbf{U}$ at the time level n), the so-called Courant–Friedrichs–Lewy (CFL) condition must be satisfied for stability, that is

$$\frac{\max_p |\lambda^{(p)}|\Delta t}{\Delta x_i} \leqslant 1 \quad \forall i \tag{6}$$

where $\lambda^{(p)}$ is the pth eigenvalue of the Jacobian matrix $\partial\mathbf{F}/\partial\mathbf{U}$. Note that Equations (1) and (3) can be generalized to several dimensions of space, but in any case the evolution of the flow variable in a given cell is computed from a balance between the fluxes at the edges of the cell and the sum of the contributions of the source terms. The flux and source terms are usually computed by solving a one-dimensional equation of the type (1) in the local co-ordinate system formed by the normal vector to the cell edge. Therefore, the key issue is the computation of the fluxes and source terms in one dimension of space.

In Godunov-type schemes the flux $\mathbf{F}_{i+1/2}^{n+1/2}$ is computed from the solution of a Riemann problem at the interface $i+1/2$ between the time levels n and $n+1$. The Riemann problem is defined as the following initial-value problem:

$$\begin{aligned} &\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} = \mathbf{S} \\ &\mathbf{U}(x, t^n) = \begin{cases} \mathbf{U}_\mathrm{L} & \text{for } x < x_{i+1/2} \\ \mathbf{U}_\mathrm{R} & \text{for } x > x_{i+1/2} \end{cases} \\ &\varphi(x) = \begin{cases} \varphi_\mathrm{L} & \text{for } x < x_{i+1/2} \\ \varphi_\mathrm{R} & \text{for } x > x_{i+1/2} \end{cases} \end{aligned} \tag{7}$$

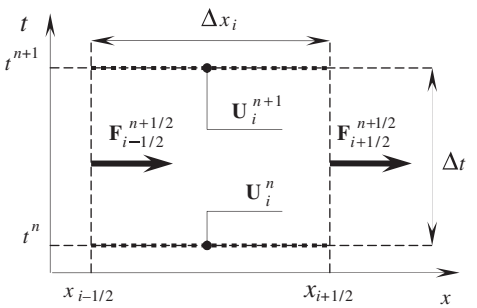

Figure 1. Definition sketch for the finite volume formalism.

DOI: 10.1002/fld

where the constant values $\mathbf{U}_L$ and $\mathbf{U}_R$ are called the left and right states of the Riemann problem, respectively.

The proposed solution method is derived from the following considerations:

1. Most existing Riemann solvers seek a solution to the conservation part of the equation alone and do not take the source term into account. This has the particular consequence that equilibrium conditions cannot always be preserved. Consider for example the solution of the shallow water equations (see Section 4.1 for a detailed treatment) in the case of water initially at rest over a non-horizontal bottom. Since the bottom level is not horizontal, the water depth is not the same on either side of the interface. Solving the Riemann problem leads to a non-zero unit discharge (that is the first component of the flux $\mathbf{F}$) across the interface and to a non-zero impulse (the second component of the flux vector $\mathbf{F}$). The particular consequence of this is that, the first component of the source term $\mathbf{S}$ being nil by definition, the non-zero unit discharge in the first component of Equation (1) cannot be compensated by a proper discretization of the source term (while this is possible for the second component via, e.g. source term upwinding [18, 22]). Spurious oscillations arise in the computed profile and propagate throughout the computational domain. This behaviour can be eliminated by using the free surface elevation as a conserved variable instead of the water depth [23]. It arises again when the source term includes the effects of channel width variations. This bias is eliminated in the present approach by taking the source term into account in the determination of the value of the flux at the interface.
2. The solver used is an approximate-state solver as defined in [2] in that the wave pattern (i.e. the nature of the waves in the solution of the Riemann problem) is assumed *a priori*. In most existing methods all waves are assumed to have the same nature, that is, all of them are assumed to be rarefaction waves (or contact discontinuities) or all of them are assumed to be shock waves. This assumption is justified in that the jump relationships (that are valid across the shock waves) can be seen as approximations of the characteristic relationships that are valid across rarefaction waves [13, 15]. Examples of Riemann solvers using a two-shock assumption for the Euler equations of gas dynamics can be found in [13, 24]. References [14, 25–27] present Riemann solvers based on the assumption of two rarefaction waves for the solution of the Euler equations of gas dynamics, shock waves in compressible liquids, the water hammer equations and a simplified system of equations that describe two-phase flow in pipes. The assumption of a solution made only of rarefaction waves is retained in the proposed approach because it makes the treatment of the source term very simple.
3. The assumption of a solution made only of rarefaction waves allow relationships involving the Riemann invariants to be written across the various waves. In contrast with the classical approach where the Riemann invariants are used to compute the value of $\mathbf{U}$, the relationships are written in a modified form that allows the value of $\mathbf{F}$ to be determined directly at the interface. Therefore the characteristic relationships are not written for $\mathbf{U}$ but for $\mathbf{F}$. This is attractive when the parameter function φ is not constant. In this case, the estimation of $\mathbf{F}$ from the value of $\mathbf{U}$ at the interface between the left and right states of the Riemann problem requires that φ be estimated at the interface. Simple equilibrium considerations using the shallow water equations as an example show that the value of φ at the interface would then depend on the values of $\mathbf{U}$ on both sides. This problem does not arise when the flux $\mathbf{F}$ is calculated directly.

DOI: 10.1002/fld

2.2. Flux and source term calculation

The developments presented in this subsection are derived from the second remark in the previous paragraph, that is, the characteristic relationships most often provide reasonably accurate approximations of the jump relationships across shocks [15]. Therefore, the solution $\mathbf{U}$ (and subsequently the flux $\mathbf{F}$) is assumed to be continuous and differentiable in what follows. Left-hand multiplying Equation (1) by the Jacobian matrix $\mathbf{A}=\partial\mathbf{F}/\partial\mathbf{U}$ leads to

$$\mathbf{A}\frac{\partial\mathbf{U}}{\partial t}+\mathbf{A}\frac{\partial\mathbf{F}}{\partial x}=\mathbf{AS} \tag{8}$$

that can be rewritten as

$$\frac{\partial\mathbf{F}}{\partial t}+\mathbf{A}\frac{\partial\mathbf{F}}{\partial x}=\mathbf{AS} \tag{9}$$

The matrix $\mathbf{K}$ of the eigenvectors of $\mathbf{A}$ is introduced. Left-hand multiplying Equation (9) by $\mathbf{K}^{-1}$ yields

$$\mathbf{K}^{-1}\frac{\partial\mathbf{F}}{\partial t}+\mathbf{K}^{-1}\mathbf{A}\frac{\partial\mathbf{F}}{\partial x}=\mathbf{K}^{-1}\mathbf{AS} \tag{10}$$

Equation (10) is rewritten so as to involve the vector $\mathbf{K}^{-1}\,\mathrm{d}\mathbf{F}$ and the diagonal form $\Lambda=\mathbf{K}^{-1}\mathbf{AK}$ of the matrix $\mathbf{A}$:

$$\mathbf{K}^{-1}\frac{\partial\mathbf{F}}{\partial t}+\mathbf{K}^{-1}\mathbf{AK}\mathbf{K}^{-1}\frac{\partial\mathbf{F}}{\partial x}=\mathbf{K}^{-1}\mathbf{AK}\mathbf{K}^{-1}\mathbf{S} \tag{11}$$

Equation (11) can be rewritten as

$$\frac{\partial\mathbf{W}}{\partial t}+\Lambda\frac{\partial\mathbf{W}}{\partial x}=\Lambda\mathbf{S}' \tag{12}$$

where the differential $\mathrm{d}\mathbf{W}$ and the source term $\mathbf{S}'$ are defined as

$$\begin{aligned}\mathrm{d}\mathbf{W}&=\mathbf{K}^{-1}\,\mathrm{d}\mathbf{F}\\ \mathbf{S}'&=\mathbf{K}^{-1}\mathbf{S}\end{aligned} \tag{13}$$

and Λ is the diagonal matrix formed by the eigenvalues of $\mathbf{A}$. Equation (12) is equivalent to the following set of differential relationships:

$$\frac{\mathrm{d}W_k}{\mathrm{d}t}=\lambda^{(k)}S'_k \quad \text{along } \frac{\mathrm{d}x}{\mathrm{d}t}=\lambda^{(k)}, \quad k=1,\ldots,m \tag{14}$$

where the index k indicates that the relationship is written for the kth component of the vector equation (12). Note that Equation (13) can also be rewritten as

$$\mathrm{d}W_k=\sum_{j=1}^{m}K^{-1}_{k,j}\,\mathrm{d}F_j \tag{15}$$

where $K^{-1}_{k,j}$ is the element on the kth row and jth column of the matrix $\mathbf{K}^{-1}$ and F_j is the jth component of the flux vector $\mathbf{F}$. If the differential vector $\mathrm{d}\mathbf{W}$ can be integrated, the components

DOI: 10.1002/fld

W_k are called the Riemann invariants of the system. Except for particular cases, d$\mathbf{W}$ cannot be integrated to provide independent Riemann invariants for $m \neq 2$ [28]. Therefore, the method is presented for the general formulation (14).

The next step in the method consists in integrating Equation (14) along each characteristic line $\mathrm{d}x/\mathrm{d}t = \lambda^{(k)}$ between the point $M(x_{i+1/2},\ t^{n+1/2})$ and the foot $A^{(k)}$ of the kth characteristic leading to M (Figure 2). The value of the flux computed at M is used as an estimate for $\mathbf{F}_{i+1/2}^{n+1/2}$ using a middle point rule

$$\int_{A^{(k)}}^{M} \mathrm{d}W_k = \frac{\Delta t}{2}\lambda^{(k)}\overline{S'_k} \tag{16}$$

where the overbar denotes the average value of the source term along the integration path.

Substituting Equation (15) into Equation (16) leads to

$$\int_{A^{(k)}}^{M} \sum_{j=1}^{m} K_{k,j}^{-1}\,\mathrm{d}F_j = \frac{\Delta t}{2}\lambda^{(k)}\overline{S'_k} \tag{17}$$

The coefficients in the kth row of the matrix $\mathbf{K}^{-1}$ are approximated using the constant coefficients on the side of the interface to which the kth characteristic is coming

$$K_{k,j}^{-1} = \begin{cases} K_{k,j}^{-1}(\mathbf{U}_{\mathrm{L}}, \varphi_{\mathrm{L}}) & \text{if } \lambda^{(p)} > 0 \\ K_{k,j}^{-1}(\mathbf{U}_{\mathrm{R}}, \varphi_{\mathrm{R}}) & \text{if } \lambda^{(p)} < 0 \end{cases} \tag{18}$$

This allows the coefficients $K_{k,j}^{-1}$ in Equation (17) to be taken out of the integral. Equation (17) leads to the following set of scalar equations:

$$\sum_{j=1}^{m} K_{k,j}^{-1}(F_{M,j} - F_{k,j}) = \frac{\Delta t}{2}\lambda^{(k)}\overline{S'_k}, \quad k = 1, \ldots, m \tag{19}$$

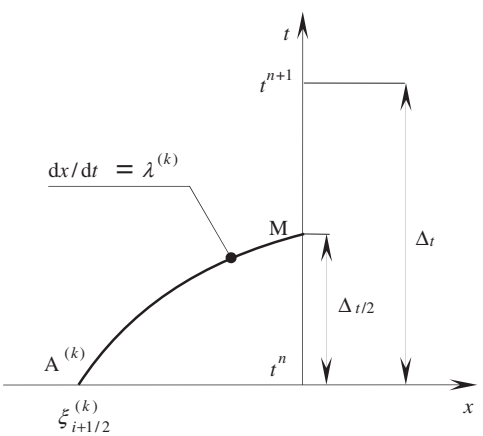

Figure 2. Definition sketch for the integration path along the characteristic line.

DOI: 10.1002/fld

where $F_{k,j}$ and $F_{M,j}$ denote the jth component of $\mathbf{F}$ at the points $A^{(k)}$ and M, respectively. $F_{k,j}$ is given by the left state of the Riemann problem if $\lambda^{(k)}$ is positive and by the right state if $\lambda^{(k)}$ is negative. Equation (19) is rewritten as

$$\sum_{j=1}^{m} K_{k,j}^{-1} F_{M,j} = \sum_{j=1}^{m} K_{k,j}^{-1} F_{k,j} + \frac{\Delta t}{2} \lambda^{(k)} \overline{S'_k}, \quad k = 1, \ldots, m \tag{20}$$

with

$$F_{k,j} = \begin{cases} F_{\mathrm{L},j} & \text{if } \lambda^{(k)} > 0 \\ F_{\mathrm{R},j} & \text{if } \lambda^{(k)} < 0 \end{cases} \tag{21}$$

Solving the $m \times m$ system of Equations (20) for the flux components $F_{M,j}$ provides the value of $\mathbf{F}$ at $(x_{i+1/2},\ t^{n+1/2})$. This value is used as an estimate for the average value $\mathbf{F}_{i+1/2}^{n+1/2}$ over the time step. To do so, the average source term $\overline{S'_k}$ must be estimated. The proposed estimate takes the form

$$\overline{S'_k} = S'_k(\mathbf{U}_{\mathrm{LR}}, \varphi_{\mathrm{LR}}, \partial\varphi/\partial x) \tag{22}$$

where $\partial\varphi/\partial x$ is given by the average slope over the domain of dependence of the solution

$$\frac{\partial \varphi}{\partial x} \approx \frac{\varphi_{\mathrm{R}} - \varphi_{\mathrm{L}}}{\max_k \lambda^{(k)} - \min_k \lambda^{(k)}} \frac{2}{\Delta t} \tag{23}$$

In the general case, many different estimates can be provided for $\mathbf{U}_{\mathrm{LR}}$ and φ_{LR}. Such estimates however should satisfy the consistency condition:

$$\begin{aligned} \mathbf{U}_{\mathrm{LR}} &= \mathbf{U}_{\mathrm{LR}}(\mathbf{U}_{\mathrm{L}}, \mathbf{U}_{\mathrm{R}}) \\ \varphi_{\mathrm{LR}} &= \varphi_{\mathrm{LR}}(\varphi_{\mathrm{L}}, \varphi_{\mathrm{R}}) \\ \mathbf{U}_{\mathrm{LR}}(\mathbf{U}, \mathbf{U}) &= \mathbf{U} \\ \varphi_{\mathrm{LR}}(\varphi, \varphi) &= \varphi \end{aligned} \tag{24}$$

and the equilibrium conditions should be preserved, that is, any combination of the left and right states that satisfy steady-state conditions should yield a steady-state flux. These two conditions are in general sufficient to yield a unique estimate for $\mathbf{U}_{\mathrm{LR}}$ and φ_{LR}.

The calculation of the contributions $\mathbf{S}_{i-1/2,i}^{n+1/2}$ and $\mathbf{S}_{i+1/2,i}^{n+1/2}$ of the source term in Equation (5) is carried out using a classical source term upwinding technique [18] that yields the following vector expression:

$$\begin{aligned} \mathbf{S}_{i-1/2,i}^{n+1/2} &= \frac{-\min_k \lambda^{(k)}}{\max_k \lambda^{(k)} - \min_k \lambda^{(k)}} \mathbf{S}_{i-1/2}(\mathbf{U}_{\mathrm{LR}}, \varphi_{\mathrm{LR}}, \partial\varphi/\partial x) \\ \mathbf{S}_{i+1/2,i}^{n+1/2} &= \frac{\max_k \lambda^{(k)}}{\max_k \lambda^{(k)} - \min_k \lambda^{(k)}} \mathbf{S}_{i+1/2}(\mathbf{U}_{\mathrm{LR}}, \varphi_{\mathrm{LR}}, \partial\varphi/\partial x) \end{aligned} \tag{25}$$

DOI: 10.1002/fld

3. APPLICATION TO THE WATER HAMMER EQUATIONS

The frictionless water hammer equations in a horizontal pipe can be written in the form (1) by defining **U**, **F** and **S** as

$$\mathbf{U}=\begin{bmatrix}\rho A\\ \rho Q\end{bmatrix},\quad \mathbf{F}=\begin{bmatrix}\rho Q\\ Ap\end{bmatrix},\quad \mathbf{S}=\begin{bmatrix}0\\ p\,\partial A/\partial x\end{bmatrix} \tag{26}$$

where A is the cross-sectional area of the pipe, p is the pressure, Q is the volume discharge and ρ is the fluid density. The matrix **A** is given by

$$\mathbf{A}=\begin{bmatrix}0 & 1\\ c^2 & 0\end{bmatrix} \tag{27}$$

where the propagation speed c of the pressure waves is classically defined as

$$c=\left[\frac{\mathrm{d}(Ap)}{\mathrm{d}(\rho A)}\right]^{1/2} \tag{28}$$

Assuming that the pressure regime remains in the elastic domain for both the pipe material and the fluid, c is a constant that depends only on the fluid compressibility and on the local geometry and mechanical properties of the pipe. The Riemann problem is to be solved for a configuration where c and A are variable in space

$$\begin{aligned} A&=\begin{cases}A_{\mathrm{L}} & \text{for } x<x_{i+1/2}\\ A_{\mathrm{R}} & \text{for } x>x_{i+1/2}\end{cases}\\ c&=\begin{cases}c_{\mathrm{L}} & \text{for } x<x_{i+1/2}\\ c_{\mathrm{R}} & \text{for } x>x_{i+1/2}\end{cases}\end{aligned} \tag{29}$$

The matrices of eigenvalues and eigenvectors of **A** are given by

$$\mathbf{\Lambda}=\begin{bmatrix}-c & 0\\ 0 & c\end{bmatrix},\quad \mathbf{K}=\begin{bmatrix}1 & 1\\ -c & c\end{bmatrix} \tag{30}$$

This leads to the following expressions for the vectors $\mathrm{d}\mathbf{W}$ and $\mathbf{S}'$ in Equation (12)

$$\mathrm{d}\mathbf{W}=\frac{1}{2}\,\mathrm{d}\begin{bmatrix}\rho Q-\dfrac{Ap}{c}\\ \rho Q+\dfrac{Ap}{c}\end{bmatrix},\quad \mathbf{S}'=\frac{1}{2}\begin{bmatrix}p\dfrac{\partial A}{\partial x}\\ p\dfrac{\partial A}{\partial x}\end{bmatrix} \tag{31}$$

Substituting Equation (31) into Equation (20) with estimates (22)–(23) and multiplying by 2 yields

$$\begin{aligned} c_{\mathrm{R}}\rho Q-Ap&=c_{\mathrm{R}}(\rho Q)_{\mathrm{R}}-(Ap)_{\mathrm{R}}+c_{\mathrm{R}}\frac{A_{\mathrm{R}}-A_{\mathrm{L}}}{c_{\mathrm{L}}+c_{\mathrm{R}}}p_{\mathrm{LR}}\\ c_{\mathrm{L}}\rho Q+Ap&=c_{\mathrm{L}}(\rho Q)_{\mathrm{L}}+(Ap)_{\mathrm{L}}+c_{\mathrm{L}}\frac{A_{\mathrm{R}}-A_{\mathrm{L}}}{c_{\mathrm{L}}+c_{\mathrm{R}}}p_{\mathrm{LR}}\end{aligned} \tag{32}$$

DOI: 10.1002/fld

where p_{LR} is determined as explained below. Solving Equation (32) for Ap and ρQ leads to

$$\begin{aligned} \rho Q &= \frac{(Ap)_L - (Ap)_R}{c_L + c_R} + \frac{c_L(\rho Q)_L + c_R(\rho Q)_R}{c_L + c_R} + \frac{A_R - A_L}{c_L + c_R} p_{LR} \\ Ap &= \frac{c_R(Ap)_L + c_L(Ap)_R}{c_L + c_R} + c_L c_R \frac{(\rho Q)_L - (\rho Q)_R}{c_L + c_R} \end{aligned} \tag{33}$$

The expression for p_{LR} is determined as follows. Consider the equilibrium state

$$\begin{aligned} (\rho Q)_L = (\rho Q)_R = (\rho Q)_0 \\ p_L = p_R = p_0 \end{aligned} \tag{34}$$

with $c_L \neq c_R$ and/or $A_L \neq A_R$. Assuming that $\partial A/\partial x$ is estimated as in Equation (23), substituting Equation (34) into Equation (33) yields the following expression for ρQ:

$$\rho Q = (\rho Q)_0 + \frac{A_L - A_R}{c_L + c_R} p_0 + \frac{A_R - A_L}{c_L + c_R} p_{LR} \tag{35}$$

Any expression for p_{LR} that satisfies the consistency condition ($p_{LR} = p_0$) leads to $\rho Q = (\rho Q)_0$, which is the desired result. Therefore the consistency condition is not sufficient to derive a satisfactory estimate of p_{LR}.

The most obvious estimate of p_{LR} is obtained by assuming a linearly varying section and a constant celerity on each side of the discontinuity. In this case the cross-sectional area at the interface is given by

$$A = \frac{A_L c_R + A_R c_L}{c_L + c_R} \tag{36}$$

Substituting Equation (36) into the second equation (31) leads to

$$p_{LR} = \frac{A_L c_R p_L + A_R c_L p_R}{A_L c_R + A_R c_L} + \frac{c_L c_R}{A_L c_R + A_R c_L}[(\rho Q)_L - (\rho Q)_R] \tag{37}$$

This expression for p_{LR} can also be achieved by assuming that the cross-sectional area is piecewise constant, with the discontinuity located at the interface. It is easily checked that writing Equations (32) with $\partial A/\partial x = 0$ on each side of the discontinuity and stating that p is continuous across the interface also leads to Equation (37). Equation (33) is to be compared to that given by the classical HLL solver

$$\begin{aligned} \rho Q &= \frac{c_L(\rho Q)_L + c_R(\rho Q)_R}{c_L + c_R} + c_L c_R \frac{(\rho A)_L - (\rho A)_R}{c_L + c_R} \\ Ap &= \frac{c_L(Ap)_L + c_R(Ap)_R}{c_L + c_R} + c_L c_R \frac{(\rho Q)_L - (\rho Q)_R}{c_L + c_R} \end{aligned} \tag{38}$$

When $c_L = c_R$ and $A_L = A_R$, Equations (33) and (38) yield identical expressions for Ap and ρQ (note that $\mathrm{d}(Ap) = c^2\,\mathrm{d}(\rho A)$):

$$\begin{aligned} \rho Q &= \frac{(Ap)_L - (Ap)_R}{2c} + \frac{(\rho Q)_L + (\rho Q)_R}{2} \\ Ap &= \frac{(Ap)_L + (Ap)_R}{2} + c\frac{(\rho Q)_L - (\rho Q)_R}{2} \end{aligned} \tag{39}$$

DOI: 10.1002/fld

When $A_L \neq A_R$ Equation (38) leads to a non-zero value for Q. This non-zero value cannot be compensated by the source term, the first component of which is nil. Consequently, the solution becomes oscillatory and may be unstable, as shown in Section 5.1.

4. APPLICATION TO THE SHALLOW WATER EQUATIONS

4.1. General formulation

The one-dimensional shallow water equations, i.e. the continuity and momentum equations with an additional equation for transverse momentum conservation, can be written in the form (1) by defining **U**, **F** and **S** as

$$\mathbf{U}=\begin{bmatrix} h \\ hu \\ hv \end{bmatrix}, \quad \mathbf{F}=\begin{bmatrix} hu \\ hu^2+gh^2/2 \\ huv \end{bmatrix}, \quad \mathbf{S}=\begin{bmatrix} 0 \\ S_0-S_f \\ 0 \end{bmatrix} \tag{40}$$

where h is the water depth, u and v are the x- and y-velocities, respectively, S_0 and S_f are the source terms arising from the bottom slope and from friction in the x-directions, respectively. The source terms S_0 and S_f are defined as

$$\begin{aligned} S_0 &= -gh\frac{\partial z_b}{\partial x} \\ S_f &= gh\frac{(u^2+v^2)^{1/2}}{K^2h^{4/3}}u \end{aligned} \tag{41}$$

The matrix **A** is given by

$$\mathbf{A}=\begin{bmatrix} 0 & 1 & 0 \\ c^2-u^2 & 2u & 0 \\ -uv & v & u \end{bmatrix} \tag{42}$$

where the propagation speed c of the pressure waves is given by

$$c=\left[\frac{\mathrm{d}(gh^2/2)}{\mathrm{d}h}\right]^{1/2}=(gh)^{1/2} \tag{43}$$

The matrices of eigenvalues and eigenvectors of **A** are, respectively,

$$\mathbf{\Lambda}=\begin{bmatrix} \lambda^{(1)} & 0 & 0 \\ 0 & \lambda^{(2)} & 0 \\ 0 & 0 & \lambda^{(3)} \end{bmatrix}, \quad \mathbf{K}=\begin{bmatrix} 1 & 0 & 1 \\ \lambda^{(1)} & 0 & \lambda^{(3)} \\ v & 1 & v \end{bmatrix} \tag{44}$$

where

$$\begin{aligned} \lambda^{(1)} &= u-c=\min(u_L-c_L, u_R-c_R) \\ \lambda^{(3)} &= u+c=\max(u_L+c_L, u_R+c_R) \\ \lambda^{(2)} &= u=\frac{\lambda^{(1)}+\lambda^{(3)}}{2} \end{aligned} \tag{45}$$

DOI: 10.1002/fld

The inverse matrix of $\mathbf{K}$ is

$$\mathbf{K}^{-1} = \frac{1}{2c}\begin{bmatrix} \lambda^{(3)} & -1 & 0 \\ -2cv & 0 & 2c \\ -\lambda^{(1)} & 1 & 0 \end{bmatrix} \tag{46}$$

Considering that friction is accounted via a time-splitting procedure, the friction term S_f does not appear in the computation of the fluxes in what follows. This leads to the following expressions for the vectors $\mathrm{d}\mathbf{W}$ and $\mathbf{S}'$ used in Equation (12):

$$\mathrm{d}\mathbf{W} = \frac{1}{2c}\begin{bmatrix} \lambda^{(3)}\,\mathrm{d}F_1 - \mathrm{d}F_2 \\ -2cv\,\mathrm{d}F_1 + 2c\,\mathrm{d}F_3 \\ -\lambda^{(1)}\,\mathrm{d}F_1 + \mathrm{d}F_2 \end{bmatrix}, \quad \mathbf{S}' = \begin{bmatrix} -S_0 \\ 0 \\ S_0 \end{bmatrix} \tag{47}$$

Considering constant values for $\lambda^{(k)}$ $(k = 1, 2, 3)$ and substituting Equation (47) into Equation (19) leads to

$$\begin{aligned} &\lambda^{(3)}(F_1 - F_{1\mathrm{R}}) - (F_2 - F_{2\mathrm{R}}) = -\lambda^{(1)} S_0^{(1)} \frac{\Delta t}{2}, \quad \frac{\mathrm{d}x}{\mathrm{d}t} = \lambda^{(1)} \\ &-\lambda^{(1)}(F_1 - F_{1\mathrm{L}}) + (F_2 - F_{2\mathrm{L}}) = \lambda^{(3)} S_0^{(2)} \frac{\Delta t}{2}, \quad \frac{\mathrm{d}x}{\mathrm{d}t} = \lambda^{(3)} \\ &-v(F_1 - F_{1,A}) + (F_3 - F_{3,A}) = 0, \quad \frac{\mathrm{d}x}{\mathrm{d}t} = \lambda^{(2)} \end{aligned} \tag{48}$$

where

$$F_{i,A} = \begin{cases} F_{i,\mathrm{L}} & \text{if } \lambda^{(2)} > 0, \\ F_{i,\mathrm{R}} & \text{if } \lambda^{(2)} < 0, \end{cases} \quad i = 1, 2, 3 \tag{49}$$

and the source term is obtained using Equation (23) for the estimate of $\partial z_\mathrm{b}/\partial x$:

$$S_0^{(i)} = -g h_{\mathrm{LR}}^{(i)} \frac{z_{\mathrm{bR}} - z_{\mathrm{bL}}}{\lambda^{(3)} - \lambda^{(1)}} \frac{2}{\Delta t}, \quad i = 1, 2 \tag{50}$$

where $h_{\mathrm{LR}}^{(i)}$ is the estimated depth at the interface between cells L and R, and z_{bL} and z_{bR} are the bottom elevations on the left- and right-hand sides of the interface respectively. System (48) can be solved uniquely for F_{1*}, F_{2*} and F_{3*} in the intermediate zone of constant state:

$$\begin{aligned} F_{1*} &= U_{2*} = \frac{F_{2,\mathrm{L}} - F_{2,\mathrm{R}} - \lambda^{(1)} F_{1,\mathrm{L}} + \lambda^{(3)} F_{1,\mathrm{R}}}{\lambda^{(3)} - \lambda^{(1)}} - g h_{\mathrm{LR}}^{(1)} \frac{z_{\mathrm{bR}} - z_{\mathrm{bL}}}{\lambda^{(3)} - \lambda^{(1)}} \\ F_{2*} &= \frac{\lambda^{(3)} F_{2,\mathrm{L}} + \lambda^{(1)} F_{2,\mathrm{R}} - \lambda^{(1)}\lambda^{(3)}(F_{1,\mathrm{L}} - F_{1,\mathrm{R}})}{\lambda^{(3)} - \lambda^{(1)}} - g h_{\mathrm{LR}}^{(2)} \frac{z_{\mathrm{bR}} - z_{\mathrm{bL}}}{\lambda^{(3)} - \lambda^{(1)}} (\lambda^{(3)} + \lambda^{(1)}) \\ F_{3*} &= \begin{cases} v_\mathrm{L} F_1 & \text{if } \lambda^{(2)} > 0 \\ v_\mathrm{R} F_1 & \text{if } \lambda^{2)} < 0 \end{cases} \end{aligned} \tag{51}$$

DOI: 10.1002/fld

Note that the estimates $h_{\mathrm{LR}}^{(1)}$ and $h_{\mathrm{LR}}^{(2)}$ in the formulae of F_{1*} and F_{2*} are not necessarily identical. Their derivation is the subject of next subsection.

4.2. Source term discretization in the continuity equation

The source term $S_0^{(1)}$ must be estimated so as to preserve equilibrium conditions, i.e. assuming a non-horizontal bottom and a static equilibrium condition, the discharge at the interface should remain nil. Many estimates can be provided for $h_{\mathrm{LR}}^{(1)}$, the most obvious of which is

$$h_{\mathrm{LR}}^{(1)} = \frac{h_{\mathrm{L}} + h_{\mathrm{R}}}{2} \tag{52}$$

Another estimate can be proposed, where the depth on each side of the discontinuity is weighted by the propagation speed of the wave that comes from this side:

$$h_{\mathrm{LR}}^{(1)} = \frac{\lambda^{(3)} h_{\mathrm{L}} - \lambda^{(1)} h_{\mathrm{R}}}{\lambda^{(3)} - \lambda^{(1)}} \tag{53}$$

where $\lambda^{(1)}$ and $\lambda^{(3)}$ are defined in Equation (45). Equation (53) is equivalent to Equation (52) in the static equilibrium case,

$$h_{\mathrm{LR}}^{(1)} = \frac{c h_{\mathrm{L}} - (-c) h_{\mathrm{R}}}{c - (-c)} = \frac{h_{\mathrm{L}} + h_{\mathrm{R}}}{2} \tag{54}$$

where $c = (\lambda^{(3)} - \lambda^{(1)})/2$ from Equation (45), and a nil mass flux is obtained

$$\begin{aligned} F_1 &= \frac{1}{\lambda^{(3)} - \lambda^{(1)}} \left(g \frac{h_{\mathrm{L}}^2}{2} - g \frac{h_{\mathrm{R}}^2}{2} \right) - g h_{\mathrm{LR}}^{(1)} \frac{h_{\mathrm{L}} - h_{\mathrm{R}}}{\lambda^{(3)} - \lambda^{(1)}} \\ &= \frac{1}{\lambda^{(3)} - \lambda^{(1)}} g (h_{\mathrm{L}} - h_{\mathrm{R}}) \left(\frac{h_{\mathrm{L}} + h_{\mathrm{R}}}{2} - h_{\mathrm{LR}}^{(1)} \right) = 0 \end{aligned} \tag{55}$$

Estimate (53) is retained in the computational examples shown in Section 5.

4.3. Source term discretization in the momentum equation

Two estimates of the source terms appear in the momentum equation: the estimate $S_0^{(2)}$ in the momentum flux F_2 and the estimate $S_0^{(3)}$ that comes directly from the source term. These two estimates cannot be assumed identical *a priori*. The second component of the vector equation (3) can be rewritten as

$$\begin{aligned} (uh)_i^{n+1} - (uh)_i^n &= \frac{\Delta t}{\Delta x_i} [(F_2)_{i-1/2}^{n+1/2} - (F_2)_{i+1/2}^{n+1/2}] + [(S_0^{(3)})_{i-1/2,i}^{n+1/2} + (S_0^{(3)})_{i+1/2,i}^{n+1/2}] \Delta t \\ &= \frac{\Delta t}{\Delta x_i} \left[\frac{\lambda_{i-1/2}^{(3)} (F_2)_{i-1}^n - \lambda_{i-1/2}^{(1)} (F_2)_i^n - \lambda_{i-1/2}^{(1)} \lambda_{i-1/2}^{(3)} [(F_1)_{i-1}^n - (F_1)_i^n]}{\lambda_{i-1/2}^{(3)} - \lambda_{i-1/2}^{(1)}} \right. \end{aligned}$$

DOI: 10.1002/fld

$$
\begin{aligned}
&- g(h_{\mathrm{LR}}^{(2)})_{i-1/2}^{n} \frac{(z_{\mathrm{b}})_{i}^{n} - (z_{\mathrm{b}})_{i-1}^{n}}{\lambda_{i-1/2}^{(3)} - \lambda_{i-1/2}^{(1)}} (\lambda_{i-1/2}^{(3)} + \lambda_{i-1/2}^{(1)}) \Bigg] \\
&- \frac{\Delta t}{\Delta x_i} \Bigg[\frac{\lambda_{i+1/2}^{(3)} (F_2)_{i}^{n} - \lambda_{i+1/2}^{(1)} (F_2)_{i+1}^{n} - \lambda_{i+1/2}^{(1)} \lambda_{i+1/2}^{(3)} [(F_1)_{i}^{n} - (F_1)_{i+1}^{n}]}{\lambda_{i+1/2}^{(3)} - \lambda_{i+1/2}^{(1)}} \\
&- g(h_{\mathrm{LR}}^{(2)})_{i+1/2}^{n} \frac{(z_{\mathrm{b}})_{i+1}^{n} - (z_{\mathrm{b}})_{i}^{n}}{\lambda_{i+1/2}^{(3)} - \lambda_{i+1/2}^{(1)}} (\lambda_{i+1/2}^{(3)} + \lambda_{i+1/2}^{(1)}) \Bigg] \\
&+ \Delta t \Bigg[-g(h_{\mathrm{LR}}^{(3)})_{i-1/2}^{n} \frac{(z_{\mathrm{b}})_{i}^{n} - (z_{\mathrm{b}})_{i-1}^{n}}{\lambda_{i-1/2}^{(3)} - \lambda_{i-1/2}^{(1)}} \frac{2}{\Delta t} \\
&- g(h_{\mathrm{LR}}^{(3)})_{i+1/2}^{n} \frac{(z_{\mathrm{b}})_{i+1}^{n} - (z_{\mathrm{b}})_{i}^{n}}{\lambda_{i+1/2}^{(3)} - \lambda_{i+1/2}^{(1)}} \frac{2}{\Delta t} \Bigg]
\end{aligned} \tag{56}
$$

where Δx_i is the size of the cell i and Δt is the computational time step. From a practical point of view, the following combination of estimates was found to be efficient for shallow water simulations:

(i) $h_{\mathrm{LR}}^{(2)}$ is set to zero in the calculation of the second flux component F_2,

(ii) the following estimate $h_{\mathrm{LR}}^{(3)}$ is used in the calculation of the source term $S_0^{(3)}$:

$$
h_{\mathrm{LR}}^{(3)} = \begin{cases} h_{\mathrm{L}} - \Delta z_{\mathrm{b}}/2 & \text{if } \Delta z_{\mathrm{b}} > 0 \\ h_{\mathrm{R}} + \Delta z_{\mathrm{b}}/2 & \text{if } \Delta z_{\mathrm{b}} < 0 \end{cases} \tag{57}
$$

where $\Delta z_{\mathrm{b}} = z_{\mathrm{bR}} - z_{\mathrm{bL}}$. This third estimate $h_{\mathrm{LR}}^{(3)}$ is determined using the assumption that the variation in the bottom level takes the form of a step (Figure 3). Then the reaction exerted by the step onto the control volume formed by two cells is given by

$$
g h_{\mathrm{LR}}^{(3)} \Delta z_{\mathrm{b}} = \begin{cases} -\displaystyle\int_{z_{\mathrm{bL}}}^{z_{\mathrm{bR}}} g(z_{\mathrm{L}} - z)\,\mathrm{d}z & \text{if } \Delta z_{\mathrm{b}} > 0 \\ \displaystyle\int_{z_{\mathrm{bR}}}^{z_{\mathrm{bL}}} g(z_{\mathrm{R}} - z)\,\mathrm{d}z & \text{if } \Delta z_{\mathrm{b}} < 0 \end{cases} \tag{58}
$$

In the case of equilibrium state, this leads to

$$
\begin{aligned}
h_{\mathrm{LR}}^{(3)} &= \begin{cases} h_{\mathrm{L}} - \dfrac{h_{\mathrm{L}} - h_{\mathrm{R}}}{2} & \text{if } \Delta z_{\mathrm{b}} > 0 \\ h_{\mathrm{R}} + \dfrac{h_{\mathrm{L}} - h_{\mathrm{R}}}{2} & \text{if } \Delta z_{\mathrm{b}} < 0 \end{cases} \\
&= \frac{h_{\mathrm{L}} + h_{\mathrm{R}}}{2}
\end{aligned} \tag{59}
$$

because the water levels z_{L} and z_{R} are equal on both sides of the interface. The main reason why a nil estimate is used for $h_{\mathrm{LR}}^{(2)}$ in the expression of the flux component F_2 is that the static equilibrium

DOI: 10.1002/fld

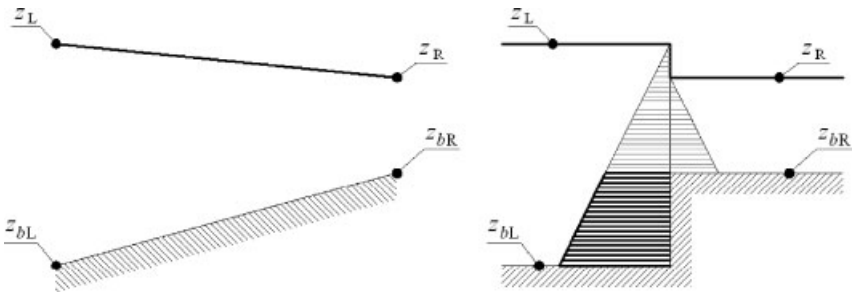

Figure 3. Analogy between forces exerted by the bottom slope and by a step.

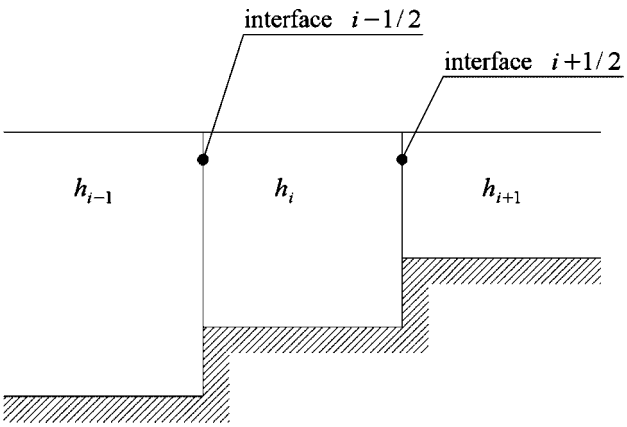

Figure 4. Definition sketch for the one-dimensional sequence of three cells with increasing bottom level.

condition is satisfied for any expression the estimate $h_{\mathrm{LR}}^{(2)}$, provided that the estimate $h_{\mathrm{LR}}^{(3)}$ in Equation (56) is equal to the estimate $h_{\mathrm{LR}}^{(1)}$ in Equation (52). Moreover, numerical experiments have shown that using a non-zero estimate for $h_{\mathrm{LR}}^{(2)}$ often leads to less accurate numerical solutions.

Consider a one-dimensional sequence of three cells with an increasing bottom level (Figure 4) under static equilibrium. The celerities used for the interface calculations are:

$$\begin{aligned}
\lambda_{i-1/2}^{(3)} &= \max(c_i, c_{i-1}) = c_{i-1} \\
\lambda_{i-1/2}^{(1)} &= \min(-c_i, -c_{i-1}) = -c_{i-1} \\
\lambda_{i+1/2}^{(3)} &= \max(c_i, c_{i+1}) = c_i \\
\lambda_{i+1/2}^{(1)} &= \min(-c_i, -c_{i+1}) = -c_i
\end{aligned} \tag{60}$$

DOI: 10.1002/fld

The momentum flux at the interface $i-1/2$ reduces to

$$(F_2)_{i-1/2}^{n+1/2} = \frac{1}{2}\left(g\frac{h_{i-1}^{n^2}}{2} + g\frac{h_i^{n^2}}{2}\right) \tag{61}$$

The momentum flux at the interface $i+1/2$ reduces to

$$(F_2)_{i+1/2}^{n+1/2} = \frac{1}{2}\left(g\frac{h_i^{n^2}}{2} + g\frac{h_{i+1}^{n^2}}{2}\right) \tag{62}$$

The source terms can be rewritten as

$$\begin{aligned}(S_0^{(3)})_{i-1/2,i}^{n+1/2} &= \frac{c_{i-1}}{2c_{i-1}}\frac{1}{\Delta x_i}\{-g(h_{\mathrm{LR}}^{(3)})_{i-1/2}^{n+1/2}[(z_\mathrm{b})_i^n - (z_\mathrm{b})_{i-1}^n]\}\\ (S_0^{(3)})_{i+1/2,i}^{n+1/2} &= \frac{c_i}{2c_i}\frac{1}{\Delta x_i}\{-g(h_{\mathrm{LR}}^{(3)})_{i+1/2}^{n+1/2}[(z_\mathrm{b})_{i+1}^n - (z_\mathrm{b})_i^n]\}\end{aligned} \tag{63}$$

The momentum equation (56) can now be rewritten, replacing the bottom levels with the depths and using Equation (59)

$$\begin{aligned}(uh)_i^{n+1} - (uh)_i^n = \frac{\Delta t}{\Delta x_i}\frac{1}{2}g\Bigg[&\frac{1}{2}(h_{i-1}^{n^2} - h_{i+1}^{n^2}) - \frac{h_{i-1}^n + h_i^n}{2}(h_{i-1}^n - h_i^n)\\ &- \frac{h_i^n + h_{i+1}^n}{2}(h_i^n - h_{i+1}^n)\Bigg]\end{aligned} \tag{64}$$

and therefore

$$(uh)_i^{n+1} - (uh)_i^n = \frac{\Delta t}{\Delta x_i}\frac{1}{2}g\left[\frac{1}{2}(h_{i-1}^{n^2} - h_{i+1}^{n^2}) - \frac{h_{i-1}^{n^2} - h_i^{n^2}}{2} - \frac{h_i^{n^2} - h_{i+1}^{n^2}}{2}\right] = 0 \tag{65}$$

In the static equilibrium case, the unit discharge uh remains constant in cell i, regardless of the expression of $h_{\mathrm{LR}}^{(2)}$ in Equation (56).

4.4. Practical implementation

Note that the values of $\mathbf{F}$ determined in Sections 4.2 and 4.3 are only the values in the 'star region', that is, in the intermediate region of constant state. The value of interest, however, is the value of $\mathbf{F}$ at the location of the initial discontinuity. In practice, five different cases can be defined from the respective locations of the intermediate zone and the interface. These cases correspond to subcritical, critical and supercritical flow conditions (Figure 5). The variables u^* and c^* of the intermediate zone are computed using the classical Riemann invariants along the characteristics:

$$\begin{aligned}u^* + 2c^* &= u_\mathrm{L} + 2c_\mathrm{L}\\ u^* - 2c^* &= u_\mathrm{R} - 2c_\mathrm{R}\end{aligned} \tag{66}$$

and consequently

$$\begin{aligned}u^* &= \frac{u_\mathrm{L} + u_\mathrm{R}}{2} + c_\mathrm{L} - c_\mathrm{R}\\ c^* &= \frac{u_\mathrm{L} - u_\mathrm{R}}{4} + \frac{c_\mathrm{L} + c_\mathrm{R}}{2}\end{aligned} \tag{67}$$

DOI: 10.1002/fld

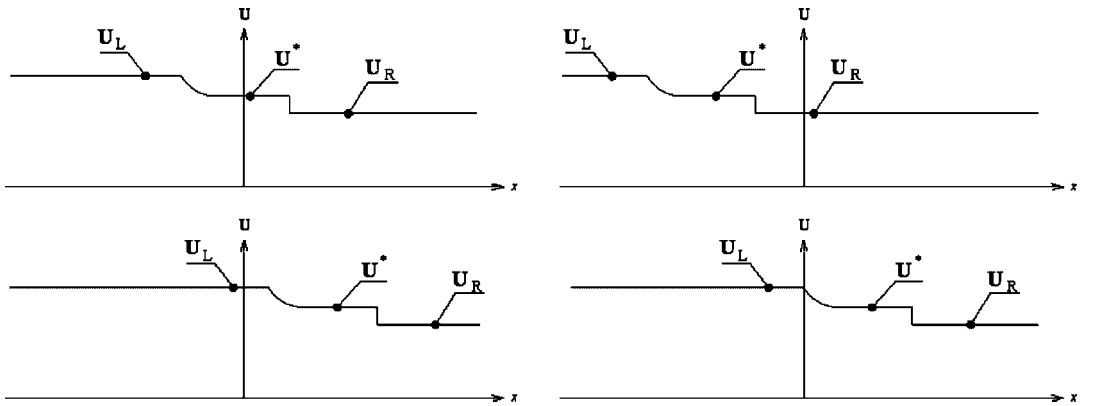

Figure 5. Relative position of the interface to the intermediate zone: top left, subcritical case, top right and bottom left, supercritical case, bottom right, critical case.

The celerities $\lambda_*^{(1)}$ and $\lambda_*^{(3)}$ of the intermediate zone are computed using u^* and c^*:

$$\begin{aligned} \lambda_*^{(3)} &= u^* + c^* \\ \lambda_*^{(1)} &= u^* - c^* \end{aligned} \tag{68}$$

and then compared with $\lambda_{\mathrm{L}}^{(1)}$ and $\lambda_{\mathrm{R}}^{(3)}$, allowing the region where the interface is located (side L, side R or the intermediate zone *) to be identified:

$$\begin{cases} F_1 = F_{1\mathrm{L}} \\ F_2 = F_{2\mathrm{L}} \end{cases} \quad \text{if } \lambda_{\mathrm{L}}^{(1)} > 0$$

$$\begin{cases} F_1 = (u_{\mathrm{L}} + 2c_{\mathrm{L}})^3/(27g) \\ F_2 = (u_{\mathrm{L}} + 2c_{\mathrm{L}})^4/(54g) \end{cases} \quad \text{if } \lambda_{\mathrm{L}}^{(1)} < 0 \text{ and } \lambda_*^{(1)} > 0$$

$$\begin{cases} F_1 = F_{1*} \\ F_2 = F_{2*} \end{cases} \quad \text{if } \lambda_*^{(1)} < 0 \text{ and } \lambda_*^{(3)} > 0 \tag{69}$$

$$\begin{cases} F_1 = (u_{\mathrm{R}} - 2c_{\mathrm{R}})^3/(27g) \\ F_2 = (u_{\mathrm{R}} - 2c_{\mathrm{R}})^4/(54g) \end{cases} \quad \text{if } \lambda_*^{(3)} < 0 \text{ and } \lambda_{\mathrm{R}}^{(3)} > 0$$

$$\begin{cases} F_1 = F_{1\mathrm{R}} \\ F_2 = F_{2\mathrm{R}} \end{cases} \quad \text{if } \lambda_{\mathrm{R}}^{(3)} < 0$$

In the critical case, the fluxes are computed using the first relation (66) and the condition $u = c$ when the flow is directed from left to right, and the second relation (66) combined to the condition $u = -c$ when the flow is directed from right to left.

DOI: 10.1002/fld

4.5. Comparison with the HLL solver

Expressions (51) can be compared with those given by the HLL method, recalled here

$$\begin{aligned}\mathbf{F} &= \frac{1}{\lambda^+ - \lambda^-}[\lambda^+\mathbf{F}_\mathrm{L} - \lambda^-\mathbf{F}_\mathrm{R} - \lambda^+\lambda^-(\mathbf{U}_\mathrm{L} - \mathbf{U}_\mathrm{R})] \\ \mathbf{U} &= \frac{1}{\lambda^+ - \lambda^-}(\mathbf{F}_\mathrm{L} - \mathbf{F}_\mathrm{R} - \lambda^-\mathbf{U}_\mathrm{L} + \lambda^+\mathbf{U}_\mathrm{R})\end{aligned} \tag{70}$$

where λ^+ and λ^- are usually estimated as in [8]

$$\begin{aligned}\lambda^+ &= \max(\lambda_\mathrm{L}^+, \lambda_\mathrm{R}^+, 0) \\ \lambda^- &= \min(\lambda_\mathrm{L}^-, \lambda_\mathrm{R}^-, 0)\end{aligned} \tag{71}$$

Except for the source terms, the first two equations (51) are similar to Equations (70). The first equation (51) is equivalent to the second component of the second equation (70). However, the second equation (70) is not used in the HLL/HLLC approach because the first equation suffices to estimate both components of the flux. The proposed approximate-state solver can be seen as a variation from the HLL/HLLC formalism, where the mass flux is derived from the equation in **U** (that is, the conserved variable) and the momentum flux is derived from the equation in **F** (that is, the flux). This apparently minor difference in the expression of the mass flux yields a substantial difference in the behaviour of the analytical solution. Unlike Equation (51), Equation (70) leads to a non-zero value for F_1 in the static equilibrium case because of the absence of source terms

$$F_1 = U_2 = \frac{1}{\lambda^+ - \lambda^-}(gh_\mathrm{L}^2/2 - gh_\mathrm{R}^2/2) \tag{72}$$

A numerical trick [23] is required to avoid in this case non-zero discharge at the interface with the HLL method. **U** is redefined as

$$\mathbf{U} = \begin{bmatrix} z \\ hu \\ hv \end{bmatrix} \tag{73}$$

With this new definition of the conserved variable, z is identical on both sides of the interface under static equilibrium conditions and a zero mass flux is obtained.

Note that the two formulations partly coincide only because the first component of the mass flux (that is, the unit discharge) coincides with the second component of the conserved variable (that is, the unit momentum). Due to the close similarity of the proposed formulation with the HLL formulation, the proposed solver is compared to the HLL solver in the computational examples of Section 5.

5. COMPUTATIONAL EXAMPLES

5.1. Water hammer simulations

The following two examples illustrate the performance of the Riemann solver defined by Equations (30) and (34). The propagation of a pressure wave into a pipe with a piecewise constant cross-sectional area is simulated. The variation in the cross-section yields a non-zero source term $p\,\partial A/\partial x$

DOI: 10.1002/fld

in the momentum equation. As shown at the end of Section 3, using a classical solver such as the HLL solver under such a configuration usually triggers spurious oscillations and sometimes numerical instability, even when static configurations are considered. Indeed, when the cross-section is variable Equation (39) yields a non-zero discharge even when the initial discharge is equal to zero and the pressures p_L and p_R are identical. However, the proposed solver, that has the same order of accuracy as the HLL solver (see Appendix A for the details of the consistency analysis), does not exhibit such a behaviour, as shown by the following two simulations.

The water is initially at rest in the pipe at an initial pressure p_0. A pressure wave is initiated by rising suddenly the pressure to a constant value p_b at the left-hand extremity of the pipe. At the section discontinuity, the pressure of the impinging wave is modified. Therefore part of the pressure and discharge signal is reflected while part of it propagates into the right-hand part of the pipe. A narrowing section (Table I, Figure 6) causes the pressure to rise, while an

Table I. Water hammer simulation.

Symbol	Meaning	Value
A_1	Pipe cross-sectional area on the left-hand side of the discontinuity	$0.1\ \mathrm{m}^2$
A_2	Pipe cross-sectional area on the right-hand side of the discontinuity	$1\ \mathrm{m}^2$
c_1	Sound celerity on the left-hand side of the discontinuity	3000 m/s
c_2	Sound celerity on the right-hand side of the discontinuity	2000 m/s
L_1	Length of the pipe on the left-hand side of the discontinuity	750 m
L_2	Length of the pipe on the right-hand side of the discontinuity	500 m
p_0	Initial pressure	10^5 Pa
p_b	Pressure at the left-hand boundary	2×10^5 Pa
Q_0	Initial discharge	$0\ \mathrm{m}^3/\mathrm{s}$
Δt	Computational time step	5×10^{-2} s
Δx_1	Cell width on the left-hand side of the discontinuity	75 m
Δx_2	Cell width on the right-hand side of the discontinuity	50 m
ρ	Water density under the initial pressure	$10^3\ \mathrm{kg}/\mathrm{m}^3$

Parameters for the first test case.

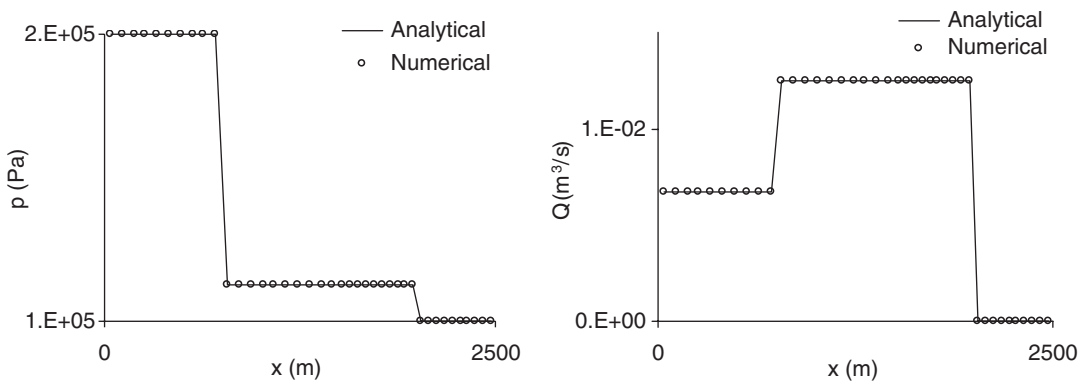

Figure 6. Water hammer simulations. Pressure (left) and discharge (right) profiles at $t = 1.5$ s for the first test case.

DOI: 10.1002/fld

Table II. Water hammer simulation.

Symbol	Meaning	Value
A_1	Pipe cross-sectional area on the left-hand side of the discontinuity	$1\ \mathrm{m}^2$
A_2	Pipe cross-sectional area on the right-hand side of the discontinuity	$0.1\ \mathrm{m}^2$
c_1	Sound celerity on the left-hand side of the discontinuity	$2000\ \mathrm{m/s}$
c_2	Sound celerity on the right-hand side of the discontinuity	$3000\ \mathrm{m/s}$
L_1	Length of the pipe on the left-hand side of the discontinuity	$500\ \mathrm{m}$
L_2	Length of the pipe on the right-hand side of the discontinuity	$750\ \mathrm{m}$
p_0	Initial pressure	$10^5\ \mathrm{Pa}$
p_b	Pressure at the left-hand boundary	$2 \times 10^5\ \mathrm{Pa}$
Q_0	Initial discharge	$0\ \mathrm{m}^3/\mathrm{s}$
Δt	Computational time step	$5 \times 10^{-2}\ \mathrm{s}$
Δx_1	Cell width on the left-hand side of the discontinuity	$50\ \mathrm{m}$
Δx_2	Cell width on the right-hand side of the discontinuity	$75\ \mathrm{m}$
ρ	Water density under the initial pressure	$10^3\ \mathrm{kg/m}^3$

Parameters for the second test case.

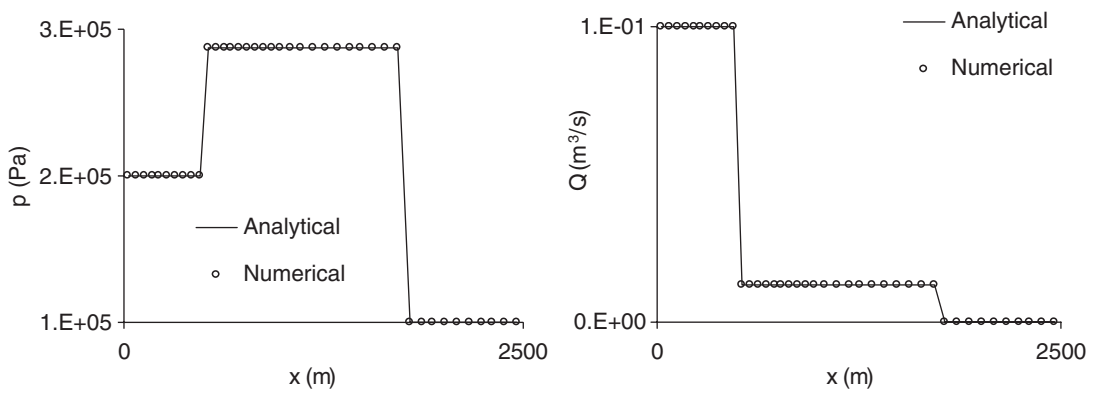

Figure 7. Water hammer simulations. Pressure (left) and discharge (right) profiles at $t = 1.5$ s for the second test case.

expanding section leads to a lower pressure (Table II, Figure 7). Figures 6 and 7 show that the numerical solutions match the analytical solutions in both cases. Simulations using the HLL solver have been shown to yield unstable solutions because of the impossibility for the HLL solver to maintain equilibrium conditions when the cross-sectional area is not constant. Due to this instability, the numerical solutions obtained using the HLL solver are not displayed in Figures 6 and 7.

5.2. Shallow water simulation: hydraulic jump with a slope discontinuity

In this test a hydraulic jump is simulated in a channel with a slope discontinuity (Table III). The purpose of this test is to assess the ability of the proposed solver to handle transcritical, steady-state conditions. The bottom slope is piecewise constant, with a steep central region and mild slopes in the upstream and downstream parts (Figure 8, top). A transition from subcritical to

DOI: 10.1002/fld

Table III. Shallow water simulation.

Symbol	Meaning	Value
g	Gravitational acceleration	$9.81\ \mathrm{m/s^2}$
h	Prescribed depth at the downstream boundary	$1\ \mathrm{m}$
K	Strickler friction coefficient	$60\ \mathrm{m^{1/3}/s}$
q	Prescribed unit discharge at the upstream boundary	$1\ \mathrm{m^2/s}$
S_1	Bottom slope on the left-hand side of the slope discontinuity	$5 \times 10^{-2}\ \mathrm{m/m}$
S_2	Bottom slope on the right-hand side of the slope discontinuity	$3 \times 10^{-3}\ \mathrm{m/m}$
Δt	Computational time step	$5 \times 10^{-2}\ \mathrm{s}$
Δx	Cell width	$1\ \mathrm{m}$

Parameters for the first test case (hydraulic jump).

supercritical conditions occurs in the form of a critical point at the transition from the mild to the steep slope. Conversely, a hydraulic jump is located next to the transition from the supercritical to the subcritical flow regime. The water profiles computed with the HLL Riemann solver and the proposed solver defined, respectively, by Equations (70) and (51) are shown in Figure 8. It can be seen that the jump is steeper with the proposed solver, the transition between subcritical and supercritical flows taking place within only two cells, while it occurs within four cells with the HLL solver. Note that this difference in the behaviours of the solvers could not be expected from the consistency analysis carried out in Appendix A. The consistency analysis is based on the assumption of a continuous, differentiable and smoothly varying solution from one computational cell to the next, which is not the case when a hydraulic jump occurs. The difference between the two solutions is only due to the presence of the source term in the continuity equation.

5.3. Shallow water simulation: frictionless dambreak

The purpose of the second shallow water simulation is to assess the ability of the solver to handle the propagation of moving shocks and the stability of critical points under unsteady conditions. A frictionless dam-break is simulated over a wetted horizontal bottom for which an analytical solution exists [29]. The parameters of the test case are given in Table IV. The water profiles computed with the HLL Riemann solver and the proposed solver at time $t = 10\ \mathrm{s}$ are shown in Figure 9. The results given by the two solvers are very close to each other, however the location of the shock wave computed using the proposed solver is closer to the analytical solution than that given by the HLL solver.

The third example is the frictionless simulation of a dam-break on a dry horizontal bottom (Table V). The water profiles computed with the HLL solver and the proposed solver are shown at time $t = 10\ \mathrm{s}$ in Figure 10. Once again the results given by the two solvers are very close to each other, but the front wave position computed using the proposed solver is closer to the analytical solution [29] than that computed using the HLL solver.

5.4. Shallow water simulation: steady flow over a bump

The fourth example is the frictionless simulation of a steady flow over a bump (Tables VI and VII). This test aims to illustrate the performance of the proposed solver in the presence of a spatially varying parameter in the source term (here the bottom slope $\partial z_\mathrm{b}/\partial x$). This bump has a parabolic shape defined by its height and length. The simulation is carried out from static initial

DOI: 10.1002/fld

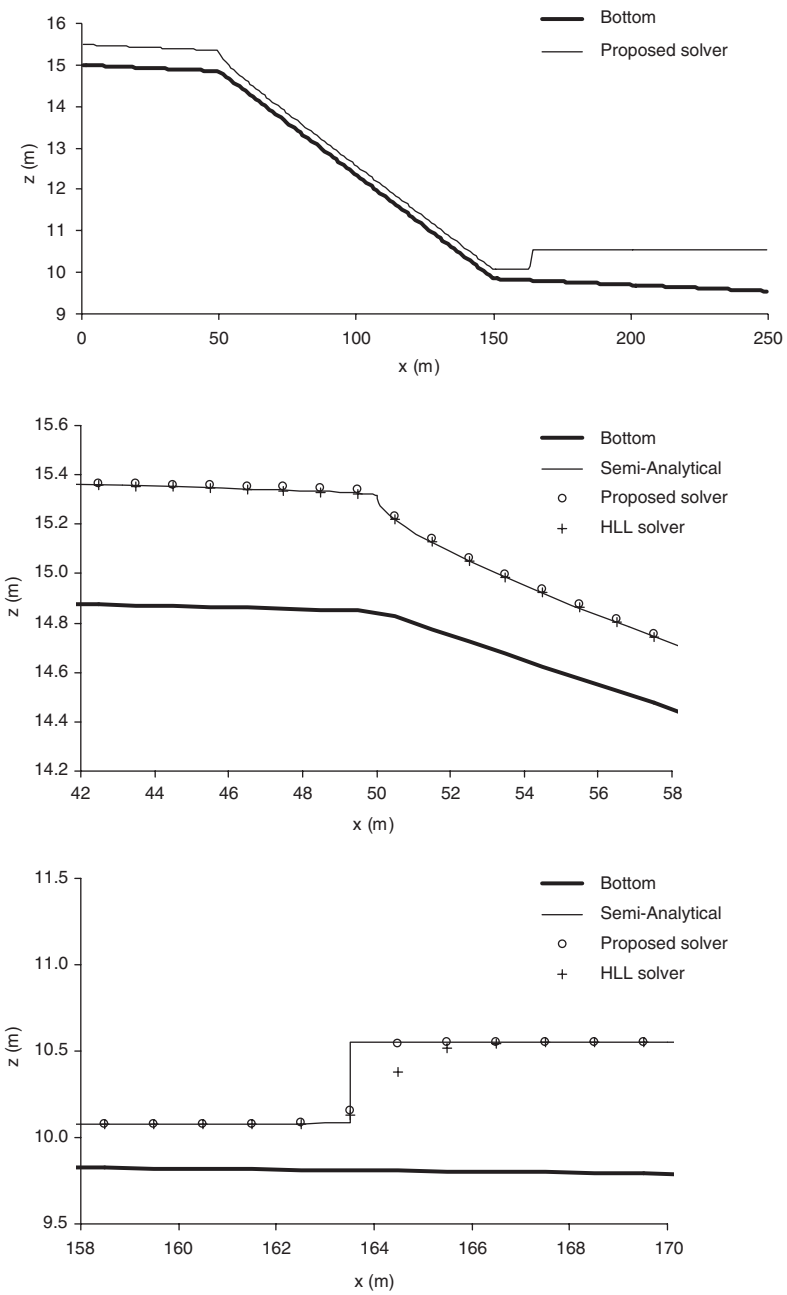

Figure 8. Shallow water simulations. Computed water level and depth for the first test case, top, plan view of the simulated flow, middle, detailed view of the hydraulic drop, bottom, detailed view of the hydraulic jump.

DOI: 10.1002/fld

Table IV. Shallow water simulation.

Symbol	Meaning	Value
g	Gravitational acceleration	$9.81\ \mathrm{m/s^2}$
h_0	Initial depth of water in the dam	$30\ \mathrm{m}$
Δt	Computational time step	$5 \times 10^{-2}\ \mathrm{s}$
Δx	Cell width	$10^{-1}\ \mathrm{m}$

Parameters for the second test case (dam-break on a wetted horizontal bottom).

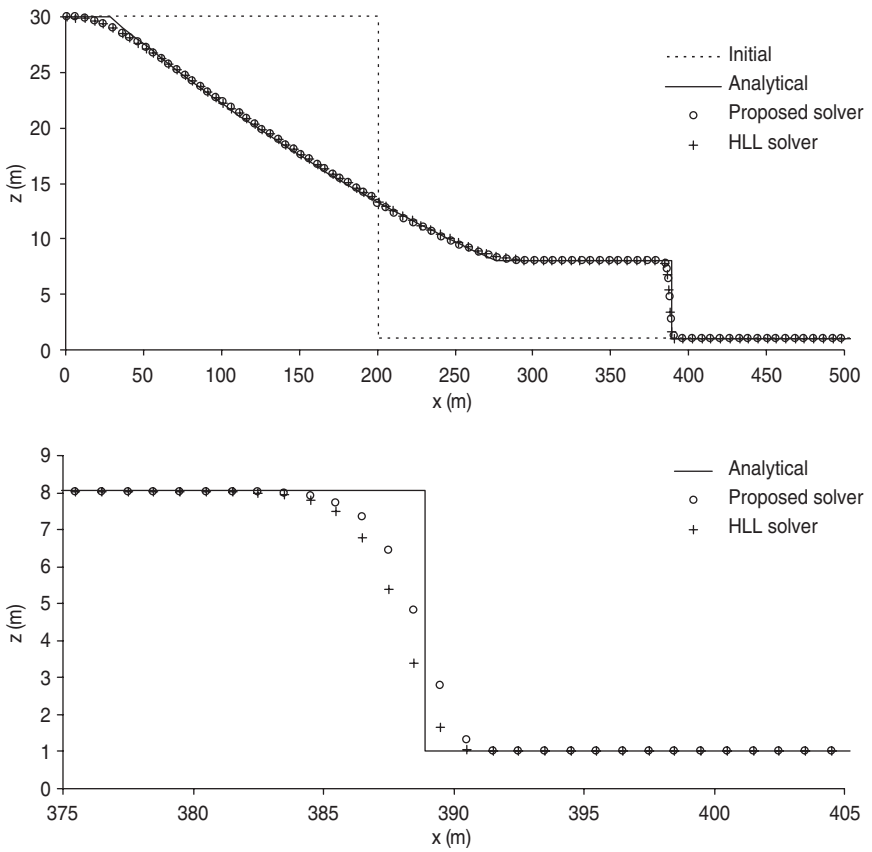

Figure 9. Shallow water simulations. Computed water levels for the second test case (frictionless simulation of a dam-break on a wetted horizontal bottom); top, plan view of the simulated dam-break flow, bottom, detailed view of the front wave.

conditions where the initial water level is arbitrary and prescribing constant boundary conditions. The simulation is carried out until steady flow conditions are reached over the computational domain. In a first simulation, the boundary conditions are defined such that the flow is subcritical over the entire domain. In a second simulation, the flow is subcritical upstream of the bump and supercritical downstream of it.

DOI: 10.1002/fld

Table V. Shallow water simulation.

Symbol	Meaning	Value
g	Gravitational acceleration	$9.81\ \mathrm{m/s^2}$
h_L	Initial depth of water in the left side of the dam	30 m
h_R	Initial depth of water in the right side of the dam	1 m
Δt	Computational time step	5×10^{-2} s
Δx	Cell width	10^{-1} m

Parameters for the third test case (dam-break on a dry horizontal bottom).

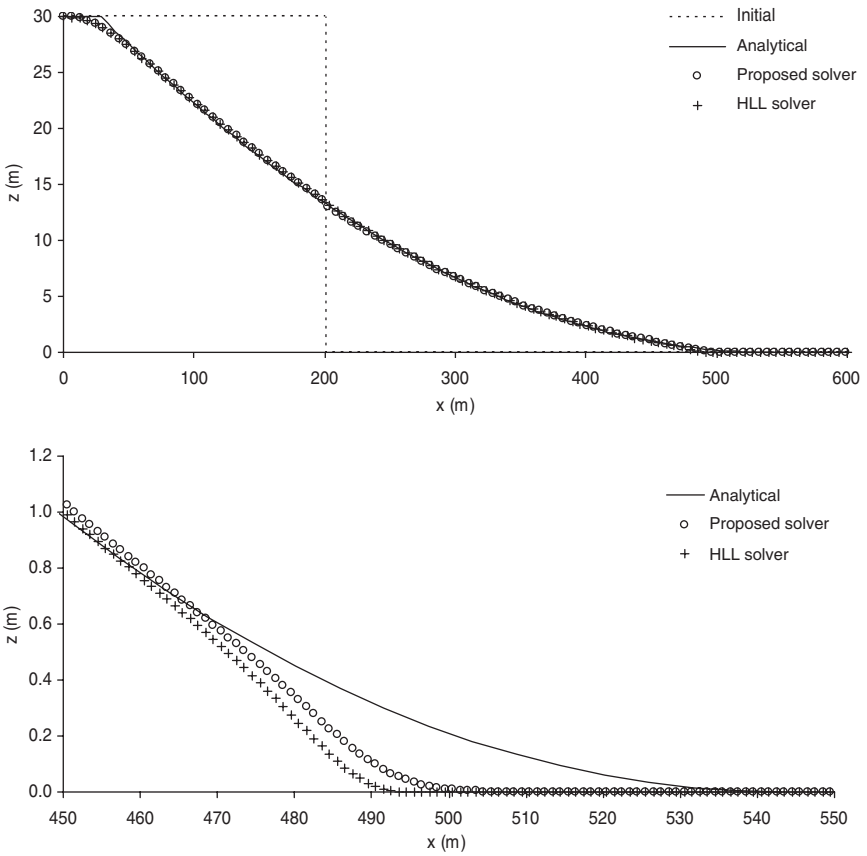

Figure 10. Shallow water simulations. Computed water levels for the third test case (frictionless simulation of a dam-break on a dry horizontal bottom); top, plan view of the simulated dam-break flow, bottom, detailed view of the front wave.

The water profiles computed with the HLL solver and the proposed solver for a subcritical flow are shown in Figure 11 and those computed for a transcritical flow are shown in Figure 12. The analytical solution is obtained by solving Bernoulli's classical steady-state equations numerically. In both cases, the water level is correctly computed by the two solvers, but the proposed

DOI: 10.1002/fld

Table VI. Shallow water simulation.

Symbol	Meaning	Value
g	Gravitational acceleration	$9.81\ \mathrm{m/s^2}$
h_b	Bump height	0.2 m
l_b	Bump length	4 m
q	Prescribed unit discharge at the upstream boundary	$4.42\ \mathrm{m^2/s}$
z_0	Initial water level	2 m
z_d	Prescribed level at the downstream boundary	2 m
Δt	Computational time step	5×10^{-2} s
Δx	Cell width	1 m

Parameters for the fourth test case (steady subcritical flow over a bump).

Table VII. Shallow water simulation.

Symbol	Meaning	Value
g	Gravitational acceleration	$9.81\ \mathrm{m/s^2}$
h_b	Bump height	0.2 m
l_b	Bump length	4 m
q	Prescribed unit discharge at the upstream boundary	$1.53\ \mathrm{m^2/s}$
z_0	Initial water level	0.66 m
z_d	Prescribed level at the downstream boundary	No condition
Δt	Computational time step	5×10^{-2} s
Δx	Cell width	1 m

Parameters for the fourth test case (steady transcritical flow over a bump).

solver computes a constant discharge over the bump, while the HLL solver predicts a discharge variation.

The reason for the difference in the unit discharge profiles obtained using the HLL solver and the proposed solver was investigated as follows. The non-constant unit discharge produced by the HLL solver may be due to several reasons, including (i) the impossibility for the HLL solver to recover satisfactorily from the transient that stems from the arbitrary initial state, or (ii) the impossibility for the HLL solver to compute correct steady-state conditions, even when a correct initial state is prescribed. Figure 13 shows the results of the following two numerical experiments. In the first experiment, the depth and velocity profiles computed by the HLL solver are used as initial conditions for a simulation using the proposed solver (Figure 13(a)). In the second experiment, the depth and velocity profiles computed by the proposed solver are used as initial conditions for a simulation using the HLL solver (Figure 13(b)). As shown in Figure 13(a), the proposed approximate state solver allows a correct unit discharge profile to be recovered from the inaccurate initial conditions given by the HLL solver. In contrast, the HLL solver quickly drifts from the correct initial state obtained from the approximate state solver, as shown in Figure 13(b). In both cases, the profile reaches its final shape approximately 10 s after the beginning of the simulation.

It is stressed that the non-uniform unit discharge profile in the HLL simulation does not indicate a loss of continuity. The finite volume formulation used in the present examples makes such a loss impossible by construction. When steady-state conditions are reached, the mass fluxes are necessarily

DOI: 10.1002/fld

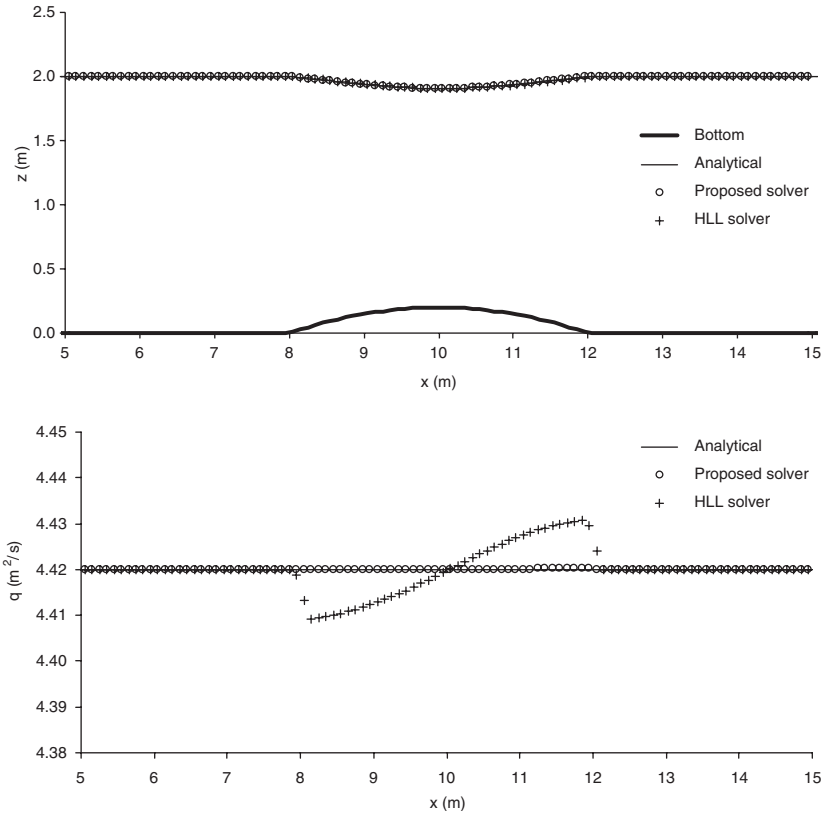

Figure 11. Shallow water simulations. Frictionless steady-state flow over a bump under subcritical conditions. Comparison of the water level (top) and discharge (bottom) computed by the HLL solver and the proposed solver with the analytical solution.

identical all throughout the domain. It should be kept in mind that the unit discharge profiles displayed in Figures 11–13 are not the mass fluxes at the interfaces between the computational cells but the average values of the unit discharge over the computational cells, that is, the conserved variable in the momentum equation. The non-constant unit discharge profile therefore indicates an improper estimate of some terms in the momentum balance rather than in the mass balance.

6. CONCLUSIONS

An approximate solver for hyperbolic systems of conservation laws with source terms is proposed. A two rarefaction wave assumption allows the Riemann invariants to be used across the waves. Introducing the flux components into the derivation of the Riemann invariants allows the fluxes to be directly calculated at the interface and the influence of the source term to be taken into account.

The proposed expressions for the fluxes partly coincide with those of the classical HLL/HLLC solver. This can be explained by (i) the equality between the first component of the flux and the

DOI: 10.1002/fld

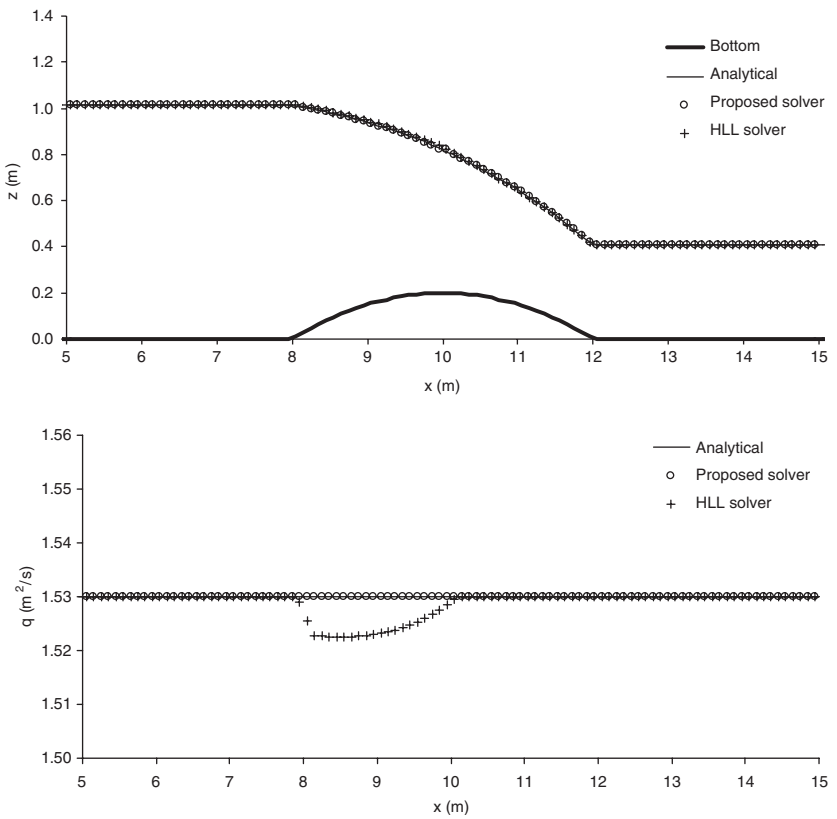

Figure 12. Shallow water simulations. Frictionless steady-state flow over a bump under supercritical conditions. Comparison of the water level (top) and discharge (bottom) computed by the HLL solver and the proposed solver with the analytical solution.

second component of the conserved variable, and (ii) the equality between the second row of the matrix $\mathbf{K}^{-1}$ of eigenvectors and the eigenvalues of the Jacobian matrix $\mathbf{A}$. A consistency analysis shows that both solvers introduce the same amount of numerical diffusion in the absence of source terms when the solution is class C^2 in the limit of small cell sizes.

Applications to the water hammer equations and the shallow water equations are presented. The solver requires the estimate of the pressure terms at the cell interfaces (via the pressure for the water hammer equations, via the depth for the shallow water equations). The minor differences with the HLL/HLLC solver suffice to introduce significant improvements in the presence of flow discontinuities and/or irregular geometries.

In presence of a sudden variation in the pipe cross-section, the pressure and discharge profiles are very close to the analytical ones. A comparison with the HLL solver shows an improvement of variables computation. Dam-break flows are equivalently simulated by the proposed solver and the HLL solver. A clear improvement on simulations of hydraulic jump and steady flow over a bump is noticed. Although the implicit assumption of a continuous and differentiable variable is used in the derivation of the fluxes, this assumption is shown to provide reliable estimates of the fluxes

DOI: 10.1002/fld

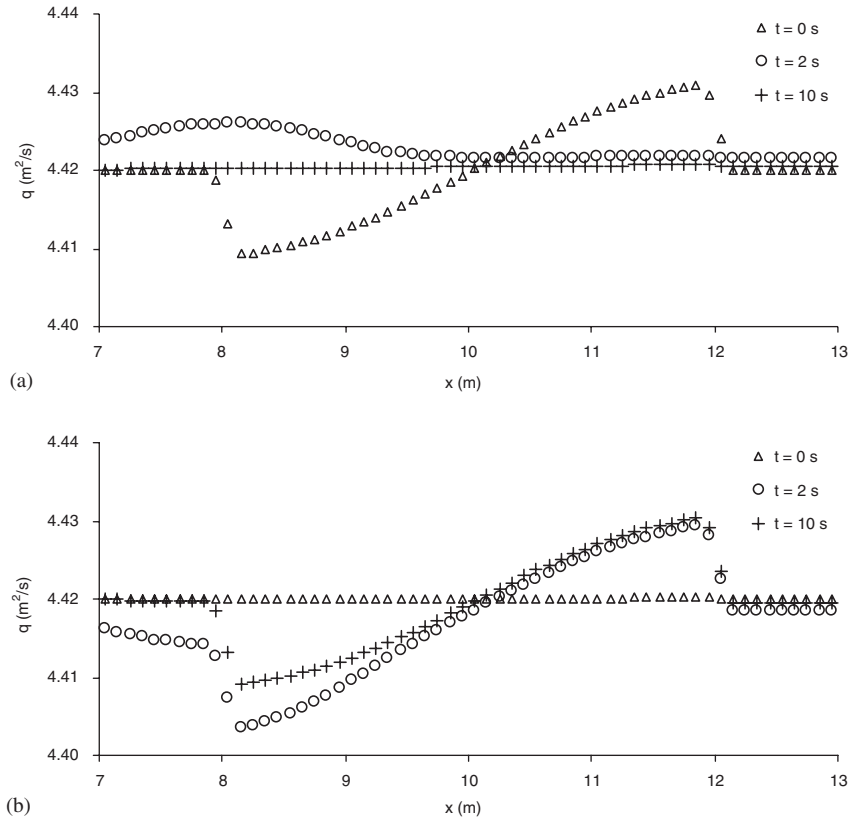

Figure 13. Shallow water simulations: (a) computed discharge for the fourth test case with the proposed solver computed from the HLL steady state; and (b) computed discharge with the HLL solver computed from the proposed solver steady state.

in situations where the flow is discontinuous and the characteristic formulation of the governing equations should not be expected to hold *a priori*.

APPENDIX A: CONSISTENCY ANALYSIS OF THE APPROXIMATE STATE AND HLL RIEMANN SOLVERS

The present appendix is devoted to a consistency analysis of the proposed solver and the HLL solver as applied to the shallow water equations. Applying Equations (51) for a horizontal bottom yields the following expressions for F_{1*} and F_{2*}:

$$F_{1*} = \frac{F_{2,\mathrm{L}} - F_{2,\mathrm{R}} - \lambda^{(1)} F_{1,\mathrm{L}} + \lambda^{(3)} F_{1,\mathrm{R}}}{\lambda^{(3)} - \lambda^{(1)}}$$
$$F_{2*} = \frac{\lambda^{(3)} F_{2,\mathrm{L}} - \lambda^{(1)} F_{2,\mathrm{R}} - \lambda^{(1)} \lambda^{(3)} (F_{1,\mathrm{L}} - F_{1,\mathrm{R}})}{\lambda^{(3)} - \lambda^{(1)}} \tag{A1}$$

DOI: 10.1002/fld

where the components F_1 and F_2 of the flux vector are defined from Equation (40)

$$\begin{aligned} F_1 &= q \\ F_2 &= \frac{q^2}{h} + g\frac{h^2}{2} \end{aligned} \tag{A2}$$

The HLL solver (see Equations (70)) yields:

$$\begin{aligned} F_{1*} &= \frac{\lambda^{(3)}F_{1,\mathrm{L}} - \lambda^{(1)}F_{1,\mathrm{R}} - \lambda^{(1)}\lambda^{(3)}(U_{1,\mathrm{L}} - U_{1,\mathrm{R}})}{\lambda^{(3)} - \lambda^{(1)}} \\ F_{2*} &= \frac{\lambda^{(3)}F_{2,\mathrm{L}} - \lambda^{(1)}F_{2,\mathrm{R}} - \lambda^{(1)}\lambda^{(3)}(U_{2,\mathrm{L}} - U_{2,\mathrm{R}})}{\lambda^{(3)} - \lambda^{(1)}} \end{aligned} \tag{A3}$$

where $\lambda^{(1)}$ and $\lambda^{(3)}$ are estimated as in Equation (71)

$$\begin{aligned} \lambda^{(1)} &= \min(u_{\mathrm{L}} - c_{\mathrm{L}}, u_{\mathrm{R}} - c_{\mathrm{R}}, 0) \\ \lambda^{(3)} &= \max(u_{\mathrm{L}} + c_{\mathrm{L}}, u_{\mathrm{R}} + c_{\mathrm{R}}, 0) \end{aligned}$$

The consistency analysis is carried out at the interface $i + 1/2$ between the cells i and $i + 1$. The cell width Δx is assumed to be constant. For the sake of simplicity the proposed solver and the HLL solver are assumed to be used in the framework of the first-order Godunov scheme, where the left and right states of the Riemann problem are assumed to be equal to the average values of the flow variables over the cells i and $i + 1$, respectively. Using the location of the interface as the origin for the x-coordinate, a second-order Taylor series expansion in U and F yields

$$\begin{aligned} U_k(x) &= U_k + x\frac{\partial U_k}{\partial x} + \frac{x^2}{2}\frac{\partial^2 U_k}{\partial x^2} + \mathrm{HOT}(x) \\ F_k(x) &= F_k + x\frac{\partial F_k}{\partial x} + \frac{x^2}{2}\frac{\partial^2 F_k}{\partial x^2} + \mathrm{HOT}(x) \end{aligned} \qquad k = 1, 2 \tag{A4}$$

where U and F are the point values of $U(x)$ and $F(x)$ at the interface, and the function $\mathrm{HOT}(x)$ are polynomials of degree at least three in x. The left and right states of the Riemann problem to be used in Equations (A1) and (A3) are obtained by averaging the expressions (A4) over the cells i and $i + 1$, respectively. The origin of the x-axis being located at the interface $i + 1/2$, the cell i extends from $x = -\Delta x$ to 0, while the cell $i + 1$ is located between $x = 0$ and $+\Delta x$.

$$\begin{aligned} U_{k,\mathrm{L}} &= \frac{1}{\Delta x}\int_{-\Delta x}^{0} U_k(x)\,\mathrm{d}x \\ U_{k,\mathrm{R}} &= \frac{1}{\Delta x}\int_{0}^{\Delta x} U_k(x)\,\mathrm{d}x \\ F_{k,\mathrm{L}} &= \frac{1}{\Delta x}\int_{-\Delta x}^{0} F_k(x)\,\mathrm{d}x \\ F_{k,\mathrm{R}} &= \frac{1}{\Delta x}\int_{0}^{\Delta x} F_k(x)\,\mathrm{d}x \end{aligned} \tag{A5}$$

DOI: 10.1002/fld

In the limit of a vanishing cell size, $\lambda^{(1)}$ and $\lambda^{(3)}$ tend to the following values:

$$\begin{aligned}\lambda^{(1)} &\approx u - c \\ \lambda^{(3)} &\approx u + c\end{aligned} \tag{A6}$$

Substituting Equations (A4) into Equations (A5) yields

$$\begin{aligned}U_{k,\mathrm{L}} &= U_k - \frac{\Delta x}{2}\frac{\partial U_k}{\partial x} + \frac{\Delta x^2}{6}\frac{\partial^2 U_k}{\partial x^2} + \mathrm{HOT}(\Delta x) \\ U_{k,\mathrm{R}} &= U_k + \frac{\Delta x}{2}\frac{\partial U_k}{\partial x} + \frac{\Delta x^2}{6}\frac{\partial^2 U_k}{\partial x^2} + \mathrm{HOT}(\Delta x) \\ F_{k,\mathrm{L}} &= F_k - \frac{\Delta x}{2}\frac{\partial F_k}{\partial x} + \frac{\Delta x^2}{6}\frac{\partial^2 F_k}{\partial x^2} + \mathrm{HOT}(\Delta x) \\ F_{k,\mathrm{R}} &= F_k + \frac{\Delta x}{2}\frac{\partial F_k}{\partial x} + \frac{\Delta x^2}{6}\frac{\partial^2 F_k}{\partial x^2} + \mathrm{HOT}(\Delta x)\end{aligned} \qquad k = 1, 2 \tag{A7}$$

Substituting Equations (A7) into (A1) yields the following expression for the fluxes in the proposed approximate-state solver:

$$\begin{aligned}F_{1*} &= F_1 - \frac{\Delta x}{2c}\left(-u\frac{\partial F_1}{\partial x} + \frac{\partial F_2}{\partial x}\right) + \frac{\Delta x^2}{6}\frac{\partial^2 F_1}{\partial x^2} + \mathrm{HOT}(\Delta x) \\ F_{2*} &= F_2 - \frac{\Delta x}{2c}\left[(c^2 - u^2)\frac{\partial F_1}{\partial x} + u\frac{\partial F_2}{\partial x}\right] + \frac{\Delta x^2}{6}\frac{\partial^2 F_2}{\partial x^2} + \mathrm{HOT}(\Delta x)\end{aligned} \tag{A8}$$

For the purpose of comparison with the HLL solver, expressions (A8) are rewritten as follows. The derivative $\partial F_2/\partial x$ is expressed as a function of $\partial F_1/\partial x$ and $\partial U_1/\partial x$ from relationships (A2)

$$\frac{\partial F_2}{\partial x} = (c^2 - u^2)\frac{\partial U_1}{\partial x} + 2u\frac{\partial F_1}{\partial x} \tag{A9}$$

Note that the equality $U_2 = F_1$ is used to introduce $\partial F_1/\partial x$ in Equation (A9). Substituting Equation (A9) into (A8) leads to the following expressions:

$$\begin{aligned}F_{1*} &= F_1 - \frac{\Delta x}{2c}\left[(c^2 - u^2)\frac{\partial U_1}{\partial x} + u\frac{\partial F_1}{\partial x}\right] + \frac{\Delta x^2}{6}\frac{\partial^2 F_1}{\partial x^2} + \mathrm{HOT}(\Delta x) \\ F_{2*} &= F_2 - \frac{\Delta x}{2c}\left[(c^2 - u^2)\frac{\partial U_2}{\partial x} + u\frac{\partial F_2}{\partial x}\right] + \frac{\Delta x^2}{6}\frac{\partial^2 F_2}{\partial x^2} + \mathrm{HOT}(\Delta x)\end{aligned} \tag{A10}$$

This expression is to be compared to the expression for the HLL solver that is obtained by introducing Equations (A7) into (A3)

$$\begin{aligned}F_{1*} &= F_1 - \frac{\Delta x}{2c}\left[(c^2 - u^2)\frac{\partial U_1}{\partial x} + u\frac{\partial F_1}{\partial x}\right] + \frac{\Delta x^2}{6}\frac{\partial^2 F_1}{\partial x^2} + \mathrm{HOT}(\Delta x) \\ F_{2*} &= F_2 - \frac{\Delta x}{2c}\left[(c^2 - u^2)\frac{\partial U_2}{\partial x} + u\frac{\partial F_2}{\partial x}\right] + \frac{\Delta x^2}{6}\frac{\partial^2 F_2}{\partial x^2} + \mathrm{HOT}(\Delta x)\end{aligned} \tag{A11}$$

Expressions (A10) and (A11) can be seen to be identical.

DOI: 10.1002/fld

The present consistency analysis leads to the following conclusions: (i) both the proposed and HLL solver yield a consistent estimate of the fluxes at the interfaces between the computational cells because F_{1*} and F_{2*} tend to the exact values F_1 and F_2, respectively, as the cell size Δx tends to 0, (ii) both solvers have the same truncation error up to the second order in Δx, and (iii) both solvers introduce the same amount of numerical diffusion in the computation of the fluxes. Note that the diffusive flux, denoted by $\mathbf{F}_D$, is expressed in vector form as

$$\mathbf{F}_D = -\frac{\Delta x}{2}\frac{c^2 - u^2}{c}\frac{\partial \mathbf{U}}{\partial x} = \frac{\Delta x}{2c}\lambda^{(1)}\lambda^{(3)}\frac{\partial \mathbf{U}}{\partial x} \tag{A12}$$

This formulation expresses the well-known property of the HLL solver that overestimating the magnitude of the wave celerities $\lambda^{(1)}$ and/or $\lambda^{(3)}$ leads to an increased stability of the numerical solution because the amount of numerical diffusion is increased. This property also applies to the proposed approximate-state solver, the degree of consistency of which is identical to that of the HLL solver up to the second order.

The difference between the behaviours of the approximate-state and the HLL solvers in Figure 8 can therefore not be attributed to a difference in the amount of numerical diffusion introduced by these solvers. Comparing Equations (A1) and (A3) shows that the approximate-state solver and the HLL solver use the same expression for the momentum flux F_2. The only difference between these two formulations lies in the expression of the mass fluxes F_1.

ACKNOWLEDGEMENTS

This work was supported by the French Ministry of Research under MRT Grant 9884-2003. The authors wish to express their gratitude to B. Cappelaere (IRD) for his comments and suggestions that proved very useful to the finalization of the present paper.

REFERENCES

1. Godunov SK. A difference method for numerical calculation of discontinuous equations of hydrodynamic equations (in Russian). *Matematicheskii Sbornik* 1959; **47**:271–300.
2. Toro EF. *Riemann Solvers and Numerical Methods for Fluid Dynamics*. Springer: Berlin, 1997.
3. Guinot V. *Godunov-type Schemes*, *An Introduction for Engineers*. Elsevier: Amsterdam, 2003.
4. LeVeque RJ. *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press: Cambridge, MA, 2002.
5. Roe PL. Approximate Riemann solvers, parameter vectors and difference scheme. *Journal of Computational Physics* 1981; **43**:357–372.
6. Toro EF. A linearised Riemann solver for the time-dependent Euler equations of gas dynamics. *Proceedings of the Royal Society London*, *Series A* 1991; **434**:683–693.
7. Harten A, Lax PD, Van Leer B. On upstream differencing and Godunov-type schemes for hyperbolic conservation laws. *Journal of Computational Physics* 1983; **50**:235–269.
8. Davis SF. Simplified second-order Godunov-type methods. *SIAM Journal on Scientific and Statistical Computing* 1988; **9**:445–473.
9. Einfeldt B. On Godunov-type methods for gas dynamics. *SIAM Journal on Numerical Analysis* 1988; **25**(2):294–318.
10. Toro EF, Spruce M, Speares W. Restoration of the contact surface in the HLL–Riemann solver. *Shock Waves* 1994; **4**:25–34.
11. Osher S, Solomon F. Upwind difference schemes for hyperbolic conservation laws. *Mathematics of Computation* 1982; **38**:339–374.
12. Linde T. A practical, general-purpose, two-state HLL Riemann solver for hyperbolic conservation laws. *International Journal for Numerical Methods in Fluids* 2002; **40**:391–402.

DOI: 10.1002/fld

13. Dukowicz JK. A general, non-iterative Riemann solver for Godunov's method. *Journal of Computational Physics* 1985; **61**:119–137.
14. Guinot V. Riemann solvers for water hammer simulations by Godunov method. *International Journal for Numerical Methods in Engineering* 2000; **49**:851–870.
15. Lax PD. Hyperbolic systems of conservation laws. *Communications on Pure and Applied Mathematics* 1957; **10**:537–566.
16. Erduran KS, Kutija V, Hewett JM. Performance of finite volume solutions to the shallow water equations with shock-capturing schemes. *International Journal for Numerical Methods in Fluids* 2002; **40**:1237–1273.
17. LeVeque RJ. Balancing source terms and flux gradients in high-resolution Godunov methods. *Journal of Computational Physics* 1998; **146**:346–365.
18. Bermudez A, Vazquez ME. Upwind methods for hyperbolic conservation laws with source terms. *Computers and Fluids* 1994; **8**:1049–1071.
19. Greenberg JM, Le Roux AY. A well balanced scheme for the numerical processing of source terms in hyperbolic equation. *SIAM Journal on Numerical Analysis* 1996; **33**(1):1–16.
20. Alcrudo F, Benkhaldoun F. Exact solutions to the Riemann problem of the shallow water equations with a bottom step. *Computers and Fluids* 2001; **30**:643–671.
21. Gallouët T, Hérard JM, Seguin N. Some approximate Godunov schemes to compute shallow-water equations with topography. *Computers and Fluids* 2003; **32**:479–513.
22. Garcia-Navarro P, Vazquez-Cendon ME. On numerical treatment of the source terms in the shallow water equations. *Computers and Fluids* 2000; **29**:951–979.
23. Nujic M. Efficient implementation of non-oscillatory schemes for the computation of free-surface flows. *Journal of Hydraulic Research* 1995; **33**:101–111.
24. Colella P. Glimm's method for gas dynamics. *SIAM Journal on Scientific and Statistical Computing* 1982; **3**:76–110.
25. Toro EF. Direct Riemann solvers for the time-dependent Euler equations. *Shock Waves* 1995; **5**:75–80.
26. Ivings MJ, Causon DM, Toro EF. On Riemann solvers for compressible liquids. *International Journal for Numerical Methods in Fluids* 1998; **28**:395–418.
27. Guinot V. Numerical simulation of two-phase flow in pipes using Godunov method. *International Journal for Numerical Methods in Engineering* 2001; **50**:1169–1189.
28. Jeffrey A, Taniuti T. *Non-Linear Wave Propagation*. Academic Press: New York, 1964.
29. Stoker JJR. *Water Waves*. Interscience: New York, 1957.

DOI: 10.1002/fld