

A numerical method for junctions in networks of shallow-water channels

Francesca Bellamoli^a, Lucas O. Müller^{b,*}, Eleuterio F. Toro^a

^a*Laboratory of Applied Mathematics, DICAM,
University of Trento,
Trento, Italy.*

^b*Division of Biomechanics, Department of Structural Engineering,
Norwegian University of Science and Technology,
Trondheim, Norway.*

Abstract

There is growing interest in developing mathematical models and appropriate numerical methods for problems involving networks formed by, essentially, one-dimensional (1D) domains joined by junctions. Examples include hyperbolic equations in networks of gas tubes, water channels and vessel networks for blood and lymph in the human circulatory system. A key point in designing numerical methods for such applications is the treatment of junctions, *i.e.* points at which two or more 1D domains converge and where the flow exhibits multidimensional behaviour. This paper focuses on the design of methods for networks of water channels. Our methods adopt the finite volume approach to make full use of the two-dimensional shallow water equations on the true physical domain, locally at junctions, while solving the usual one-dimensional shallow water equations away from the junctions. In addition to mass conservation, our methods enforce conservation of momentum at junctions; the latter seems to be the missing element in methods currently available. Apart from simplicity and robustness, the salient feature of the proposed methods is their ability to successfully deal with transcritical and supercritical flows at junctions, a property not enjoyed by existing published methodologies. Systematic assessment of the proposed methods for a variety of flow configurations is carried out. The methods are directly applicable to other systems, provided the multidimensional versions of the 1D equations are available.

Keywords:

Networks, junctions, shallow water flows, supercritical flow, Godunov methods, ADER methods.

1. Introduction

There are multidimensional physical problems modelled by partial differential equations in networks of spatial domains than are essentially straight. In such cases the governing equations can be assumed to be one-dimensional (1D), potentially resulting in significant computing savings. Examples include gas flow in pipes [6, 13, 5, 34, 9], traffic flow [15, 11, 12] water flows [21, 1, 4, 42, 10, 22] and blood flow in the human circulatory system [32, 18, 31, 36, 26, 3, 24, 25, 19, 23, 28, 29, 41]. The challenge, however, is how to connect these 1D domains in a way that accounts for the multidimensional character of the equations, even in an approximately manner.

Current methods are reported to perform well in most cases. However, a shortcoming of existing methods is their inability to deal with transcritical and supercritical flow through junctions. In some cases, these methods fail even for subcritical flows at moderately high Froude numbers. Transcritical and supercritical flows are important flow regimes that may occur more often than one is aware of, for example at junctions, locally. In physiological flows this is found to be the case in the venous system, under postural changes.

*Corresponding author

Email address: lucas.o.muller@ntnu.no (Lucas O. Müller)

In open channel flows the occurrence of supercritical flows is not rare and may potentially take place in inundating flows emerging from dam-break events and tsunami waves. Supercritical regimes may also appear in networks of tubes transporting compressible fluids under extreme accidental conditions.

In this paper we present methods for dealing with junctions connecting 1D domains and illustrate the ideas for junctions of 1D shallow water channels. We note that the full problem is governed by the two-dimensional (2D) shallow water equations. The methods presented here make use of the finite volume approach, whereby the true geometry is accounted for locally at junctions, whereas away from junctions, the usual 1D equations are solved. In addition to mass conservation, our methods enforce conservation of momentum at junctions, which constitutes an improvement over methods currently available. It is noted that the approach, as applied to complex networks of channels, can lead to very significant computing savings, as compared to solving the full multidimensional problem, without compromising the solution quality. Systematic assessment of the methods for a variety of flow configurations is carried out. It is worth noting that a similar method, which combines a 1D model for the channels and a 2D model on unstructured grids for junctions, has been investigated both in hydrodynamics (Miglio et al. [26]) and in haemodynamics (Formaggia et al [17, 33]). However, Miglio et al. ([26]) use a finite element scheme and investigate only the case of subcritical flows.

The rest of this paper is structured as follows. In Section 2 we briefly present the underlying mathematical models and the numerical framework upon which the proposed methods are constructed. Next, the novel methodology for coupling 1D domains at junctions is illustrated in Section 3. Section 4 is devoted to the validation of the proposed methods, while concluding remarks are presented in Section 5. [Appendix A](#) describes an existing method for junctions and [Appendix B](#) provides details on the 2D unstructured mesh method used to produce reliable reference solutions used to assess the methods presented in this paper.

2. Mathematical models and numerical method

We are concerned with free-surface shallow water flow under gravity in a network system consisting of interconnected straight (or essentially straight) channels joined at junctions, as illustrated in Fig. 1. The flow field has a significant 2D structure only in the vicinity of junctions, while it is essentially 1D along the straight channels, away from junctions. The purpose of this work is to develop a method that combines the use of a 1D model in channels and a 2D model only at junctions, coupling these two models with appropriate matching conditions. As we shall show later, the resulting methods show a huge computational efficiency gain with respect to solving the full 2D equations on the entire domain.

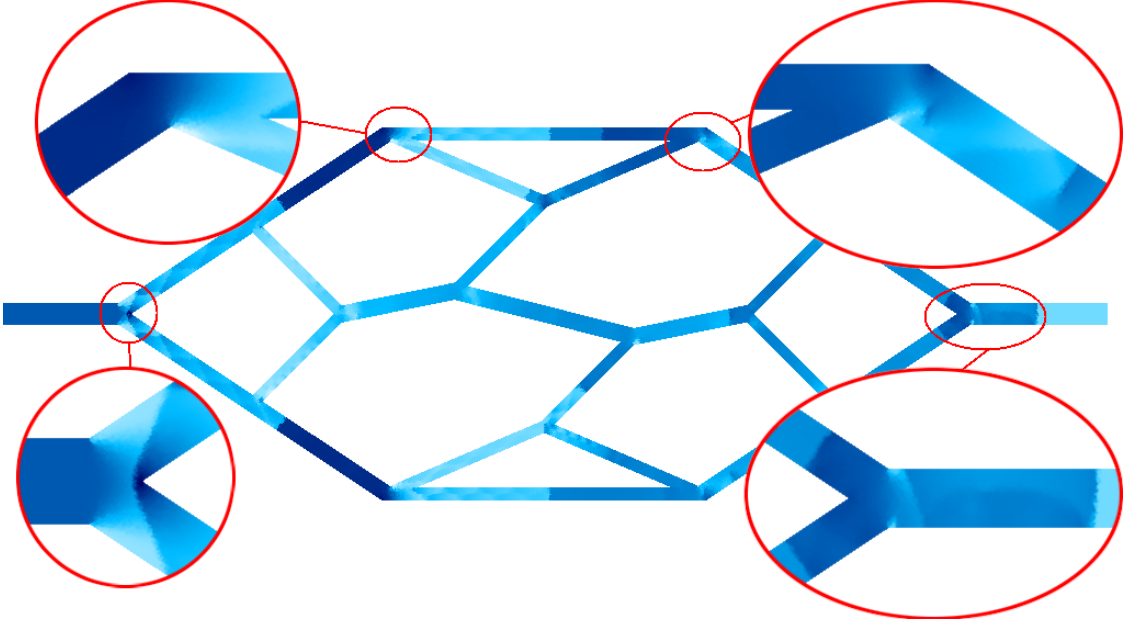


Figure 1: Example of a channel network. Regions with two-dimensional behaviour are encircled and zoomed in.

The methods proposed in this paper adopt the finite volume framework and combine the 2D equations and its corresponding local geometry in a neighbourhood of junctions, along with the 1D equations in the straight channels. Next, we recall the governing equations and the finite volume method.

2.1. The shallow water equations

The time-dependent, non-linear, 2D shallow water equations written in conservation form read

$$\partial_t \mathbf{Q} + \partial_x \mathbf{F}(\mathbf{Q}) + \partial_y \mathbf{G}(\mathbf{Q}) = \mathbf{S}(\mathbf{Q}), \quad (1)$$

with

$$\left. \begin{aligned} \mathbf{Q} &= \begin{bmatrix} h \\ hu \\ hv \end{bmatrix}, & \mathbf{F}(\mathbf{Q}) &= \begin{bmatrix} hu \\ hu^2 + \frac{1}{2}gh^2 \\ huv \end{bmatrix}, & \mathbf{G}(\mathbf{Q}) &= \begin{bmatrix} hv \\ hvu \\ hv^2 + \frac{1}{2}gh^2 \end{bmatrix}, \\ \mathbf{S}(\mathbf{Q}) &= \begin{bmatrix} 0 \\ gh(S_{ox} - S_{fx}) \\ gh(S_{oy} - S_{fy}) \end{bmatrix}. \end{aligned} \right\} \quad (2)$$

Here \mathbf{Q} is the vector of conserved variables; $\mathbf{F}(\mathbf{Q})$ and $\mathbf{G}(\mathbf{Q})$ are the fluxes in the x and y directions, respectively, and $\mathbf{S}(\mathbf{Q})$ is the vector of source terms. The physical variables are water depth $h(x, y, t)$ and velocity components $u(x, y, t)$ and $v(x, y, t)$, in the x and y directions respectively. In this paper the source term vector $\mathbf{S}(\mathbf{Q})$ accounts for the variation of the bottom topography

$$S_{ox} = -\partial_x b(x, y) \quad \text{and} \quad S_{oy} = -\partial_y b(x, y) \quad (3)$$

and the bed friction

$$S_{fx} = \frac{n^2 u \sqrt{u^2 + v^2}}{h^{4/3}} \quad \text{and} \quad S_{fy} = \frac{n^2 v \sqrt{u^2 + v^2}}{h^{4/3}}. \quad (4)$$

Here $b(x, y)$ represents bottom elevation above a horizontal datum, n is the Manning's coefficient and g is the acceleration due to gravity. In this work we only consider channels with horizontal bottom, that is $S_{ox} = 0$,

$S_{oy} = 0$. Equations (1) form a system of partial differential equations of hyperbolic type. For background on the shallow water equations and associated numerical methods see, for instance, [39] and the many references therein.

The one-dimensional version of (1), in the generic s direction, reads

$$\partial_t \mathbf{Q}_{1D} + \partial_s \mathbf{F}_{1D}(\mathbf{Q}_{1D}) = \mathbf{S}_{1D}(\mathbf{Q}_{1D}), \quad (5)$$

with

$$\mathbf{Q}_{1D} = \begin{bmatrix} h_{1D} \\ h_{1D} u_{1D} \end{bmatrix}, \quad \mathbf{F}_{1D}(\mathbf{Q}_{1D}) = \begin{bmatrix} h_{1D} u_{1D} \\ h_{1D} u_{1D}^2 + \frac{1}{2} g h_{1D}^2 \end{bmatrix}, \quad \mathbf{S}_{1D}(\mathbf{Q}_{1D}) = \begin{bmatrix} 0 \\ g h_{1D} (S_{os} - S_{fs}) \end{bmatrix}, \quad (6)$$

where $h_{1D} = h$ and u_{1D} is the velocity along s .

2.2. Rotational invariance and numerical method

As previously stated, the methods proposed in this paper combine 1D equations (5) in a generic direction s and the 2D equations (1), locally, at a single 2D element with an arbitrary number of edges. Thus, it is convenient to recall the rotational invariance property of equations (1). First, let us define an arbitrary 2D spatial control volume V with boundary Ω , as depicted in Fig. 2, top frame. Equations (1), expressed in integral form, read

$$\frac{\partial}{\partial t} \int_V \mathbf{Q} dV + \int_{\Omega} [\cos \theta \mathbf{F}(\mathbf{Q}) + \sin \theta \mathbf{G}(\mathbf{Q})] d\Omega = \mathbf{0}. \quad (7)$$

Moreover, we define the outward unit vector normal to Ω , \mathbf{n} , as

$$\mathbf{n} \equiv [n_1, n_2] \equiv [\cos \theta, \sin \theta]. \quad (8)$$

The top frame of Fig. 2 depicts the control volume V in the Cartesian plane, where x denotes the chosen reference direction, while the bottom frame depicts a typical computational finite volume with five vertices and five edges. Equations (1) satisfy the rotational invariance property [39]

$$\mathbf{H} \equiv \mathbf{n} \cdot [\mathbf{F}(\mathbf{Q}), \mathbf{G}(\mathbf{Q})] = \cos \theta \mathbf{F}(\mathbf{Q}) + \sin \theta \mathbf{G}(\mathbf{Q}) = \mathbf{T}^{-1} \mathbf{F}(\mathbf{T}(\mathbf{Q})) \quad (9)$$

for all vectors \mathbf{Q} and for all real angles θ , or equivalently, normal directions of Ω . Here $\mathbf{T} = \mathbf{T}(\theta)$ is a rotation matrix and $\mathbf{T}^{-1}(\theta)$ is its inverse, given respectively as

$$\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & \sin \theta \\ 0 & -\sin \theta & \cos \theta \end{bmatrix}, \quad \mathbf{T}^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{bmatrix}. \quad (10)$$

Now, we choose a computational control volume V_k in two-dimensional space, as shown for example in the bottom frame of Fig. 2. Moreover, we define a space-time control volume $I_k = [t^{n+1}, t^n] \times V_k$, over which we integrate (1), yielding

$$\mathbf{Q}_k^{n+1} = \mathbf{Q}_k^n - \frac{\Delta t}{|V_k|} \sum_{e=1}^N \mathcal{F}_e, \quad (11)$$

with cell averages defined as

$$\mathbf{Q}_k^n = \frac{1}{|V_k|} \int_{V_k} \mathbf{Q}(x, y, t^n) dV, \quad (12)$$

where $|V_k|$ denotes the volume of V_k (or area in the 2D case), while the intercell flux for edge e is

$$\mathcal{F}_e = \int_{A_e}^{A_{e+1}} \mathbf{T}_e^{-1} \mathbf{F}(\mathbf{T}_e(\mathbf{Q})) dA \approx \mathcal{L}_e \mathbf{T}_e^{-1} \hat{\mathbf{F}}_e. \quad (13)$$

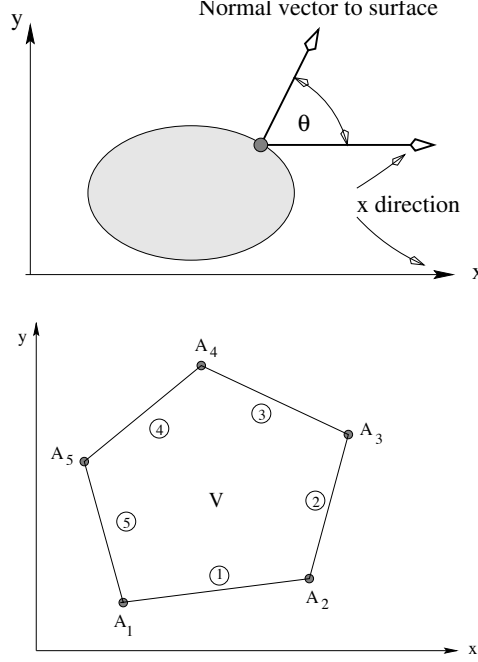


Figure 2: Control volumes, rotational invariance and finite volumes. Top frame: arbitrary control volume V in Cartesian plane; the x -direction is the reference direction, θ is angle between the outward unit normal vector \mathbf{n} and the reference x -direction. Bottom frame: typical finite volume in the Cartesian plane.

Here \mathcal{L}_e is the length of edge e , the segment $A_e A_{e+1}$. $\hat{\mathbf{F}}_e \approx \mathbf{F}(\mathbf{T}_e(\mathbf{Q}))$ is an approximation to the flux \mathbf{F} on the edge e evaluated at the rotated state $\mathbf{T}_e(\mathbf{Q})$, where the rotation is performed in the normal direction to side e through the transformation matrix \mathbf{T}_e . The final expression of the finite volume scheme becomes

$$\mathbf{Q}_k^{n+1} = \mathbf{Q}_k^n - \frac{\Delta t}{|V_k|} \sum_{e=1}^N \mathcal{L}_e \mathbf{T}_e^{-1} \hat{\mathbf{F}}_e . \quad (14)$$

For completeness we now illustrate the computation of the flux $\hat{\mathbf{F}}_e$ for an arbitrary edge e , see Fig.3. Conventionally, the left side L of edge e is always in the interior of the control volume of interest and the right side R is outside. The computation of the numerical flux $\hat{\mathbf{F}}_e$, as for a first-order Godunov type method for example, involves the augmented, local one-dimensional Riemann problem in the rotated frame normal to the edge, namely

$$\left. \begin{array}{l} \text{PDEs in normal direction: } \partial_t \mathbf{Q}_{1D} + \partial_s \mathbf{F}_{1D}(\mathbf{Q}_{1D}) = \mathbf{0} , \\ \text{Rotated initial conditions: } \mathbf{Q}_{1D}(s, 0) = \begin{cases} \mathbf{Q}_{1D,L} = \mathbf{T}_e(\mathbf{Q}_{2D,L}) & \text{if } s < 0 , \\ \mathbf{Q}_{1D,R} = \mathbf{T}_e(\mathbf{Q}_{2D,R}) & \text{if } s > 0 . \end{cases} \end{array} \right\} \quad (15)$$

The steps to be followed in order to solve problem (15) can be summarised as:

1. Calculate the angle θ_e between the outward unit normal to edge e and the fixed reference direction x , being positive in the counter-clock direction.
2. Calculate the corresponding rotation matrix \mathbf{T}_s and its inverse from (10).
3. Rotate left and right data as in (15).

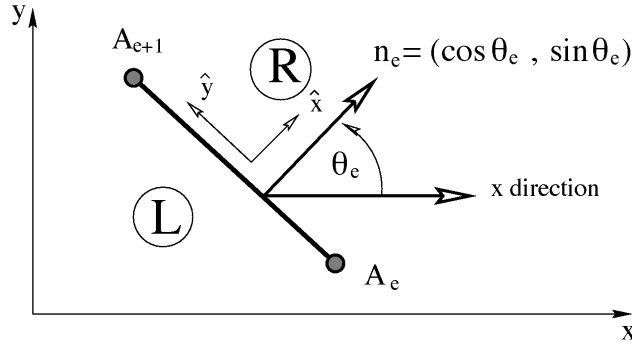


Figure 3: Generic edge of a control volume V in x - y space, where by convention the left side L lies inside the control volume and the right side R lies outside. The outward normal unit vector, with respect to the fixed reference direction x , is depicted as well as its corresponding angle.

4. Solve the 1D Riemann problem (15) on rotated data and compute the corresponding flux $\hat{\mathbf{F}}_e$.
5. Rotate back the flux as in (9) and multiply it by edge length to get the final intercell numerical flux for edge e .

Once numerical fluxes for all edges have been calculated, element k can be updated through the finite volume formula (11). This description applies to any two-dimensional finite volume method on a general mesh, assumed here to be unstructured. More details are given in Appendix B. However, for the junction method proposed in this paper, will only use the above description at a single 2D element placed right at the junction.

3. A methodology for channel junctions/bifurcations

The geometrical and numerical approaches of our proposed junction method are described below.

3.1. The approach

In short, our method for a configuration as shown in Fig. 1 uses 1D formulations for every straight channel and a single 2D element at each junction, as depicted in Fig. 4. The 2D subdomain is then linked to 1D channels through appropriate matching conditions, to be described. As previously noted, similar methods have been investigated in the past, both in hydrodynamics [26] and in haemodynamics [17, 33]). Miglio et al. [26] used a finite element scheme and investigated only the case of subcritical flows. In the present work we are interested in general configurations and, principally, in all possible flow regimes: subcritical, transcritical and supercritical. Our approach is independent of the particular numerical method chosen for solving the shallow water equations, but here we implement first, second and third order accurate Godunov-type finite volume methods in the ADER framework [40].

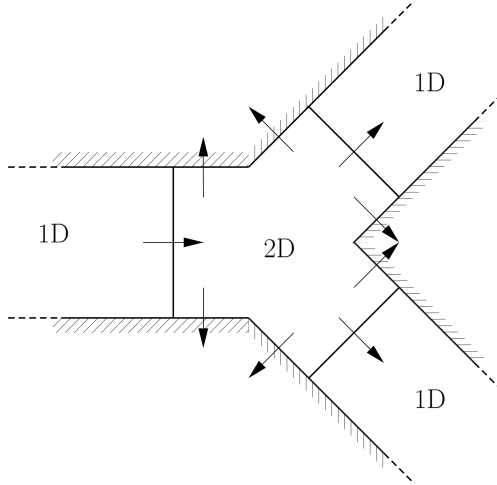


Figure 4: Single 2D element at junction. The single element exchanges fluxes with all connected 1D domains. At solid walls of the rectangular cross-section channel, suitable reflective boundary conditions apply through the corresponding numerical fluxes.

We remark that the choice of the shape for the 2D junction element is important and there many possible choices for fitting a single finite volume method at the junction. After investigating several possibilities we concluded that the best choice is that of a *junction-shaped* 2D element, as displayed in Fig. 4. This 2D element protrudes into the 1D converging channels by 0.1 times the channel width, incorporating in this manner, geometrical information on the direction of the 1D domains. Other choices for the shape of the 2D element were explored in [8]. The resulting method is called **Method A** throughout this paper. A simple variation, called **Method B**, results from the insertion of a local 2D unstructured grid composed of more than one element to represent the junction and its vicinity.

Regarding the numerical methodology for the 1D and the 2D shallow water equations we use Godunov-type methods with the approximate Riemann solver HLLC [37]. First, second and third order accurate methods are implemented. The high-order methods follow the ADER approach [38] with the Harten-type method to solve the generalised Riemann problem [20]. For background on the ADER approach see chapters 19 and 20 of [40].

The time step Δt is computed imposing a CFL condition on both the 1D and the 2D junction elements in the usual manner. Then the size Δt is taken as the minimum among all time steps and applied to the full domain. Note that in the 2D case the maximum CFL number for stability is $CFL_{2D} = CFL_{1D}/2$. In what follows we address in more detail each of the issues arising from the coupling of 1D domains and 2D elements.

3.2. Computing two-dimensional and one-dimensional fluxes

We calculate the 2D fluxes by solving the rotated 1D Riemann problem (15) in local coordinates. To this end we use the HLLC approximate Riemann solver [37]. As initial data we have x -velocity and y -velocity components in the 2D domain, where we use a global, predefined, reference system. We also have axial velocity and transverse velocity (that is zero) in the 1D domain, where we use a local reference system, see figure 5. However, for the computation of 2D fluxes we need x -velocity and y -velocity components both to the right and to the left of each edge of the element. Therefore, we need to rotate the vectors of conserved variables as follows:

$$\mathbf{Q}_R = \begin{bmatrix} h_{2D} \\ h_{2D}u_{2D} \\ h_{2D}v_{2D} \end{bmatrix} \quad \mathbf{Q}_L = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix} \begin{bmatrix} h_{1D} \\ h_{1D}u_{1D} \\ h_{1D}v_{1D} \end{bmatrix}, \quad (16)$$

where u_{1D} , v_{1D} and h_{1D} , are the variables in the 1D domain, while u_{2D} , v_{2D} and h_{2D} denote the variables in the 2D domain, as depicted in Fig. 5. For a second or higher order scheme, these data values result from reconstructed polynomials evaluated at the edges. Once the variables are available, we can apply the classical

HLLC solver [37] as described in Appendix B. For each channel and for each side of the junction there is a reference angle which we call α .

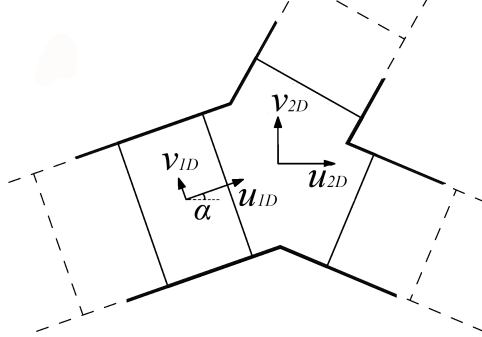


Figure 5: Reference frame for the 2D element and the 1D domain of the left channel.

Reflective boundary conditions are set on the remaining edges of the junction-shaped element, giving rise to symmetric Riemann problems, see [40] for details. With regard to the 1D channel on the left side of Fig. 5, the problem is inverted; we need axial velocity and transversal velocity both to the right and to the left of each edge of the 1D cell. Vectors of conserved variables become:

$$\mathbf{Q}_L = \begin{bmatrix} h_{1D} \\ h_{1D}u_{1D} \\ h_{1D}v_{1D} \end{bmatrix} \quad \mathbf{Q}_R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & \sin(\alpha) \\ 0 & -\sin(\alpha) & \cos(\alpha) \end{bmatrix} \begin{bmatrix} h_{2D} \\ h_{2D}u_{2D} \\ h_{2D}v_{2D} \end{bmatrix}. \quad (17)$$

3.3. Dealing with transverse velocity in 1D domains

Obviously, in all 1D elements we assume 1D motion and thus the transverse velocity component is zero. However, a problem arises at the element of a 1D channel adjacent to a 2D junction element, since we might end up with a non-zero transverse velocity. In the 2D elements, at time t^n we generally will have two non-zero velocity components, and consequently we could obtain a non-zero transversal velocity also in the 1D element at time t^{n+1} due to the 2D flux at the edge of the 2D element. To deal with this difficulty we have considered two approaches. One possibility is to simply set to zero the transverse velocity and take as 1D axial velocity the normal velocity component. The second option, which we prefer, is to calculate the axial 1D velocity as

$$u_{1D}^{n+1} = \text{sign}(u_{1D}^{n+1}) \sqrt{(u_{2D}^{n+1})^2 + (v_{2D}^{n+1})^2}. \quad (18)$$

This means that the 2D velocity vector has been rotated in the direction of the 1D channel and consequently the transverse velocity component is zero. Inevitably, in both approaches, momentum balance at the 1D elements adjacent to the 2D element is effectively altered, even though in the 2D element momentum balance is strictly satisfied.

3.4. Spatial reconstruction for high-order accuracy

For 1D cells adjacent to 2D cells we perform a modified version of the spatial reconstruction described in Appendix B, by projecting the distance between the centroid of the 2D element and the centre of the 1D cell along the normal to the boundary. See figure 6(b).

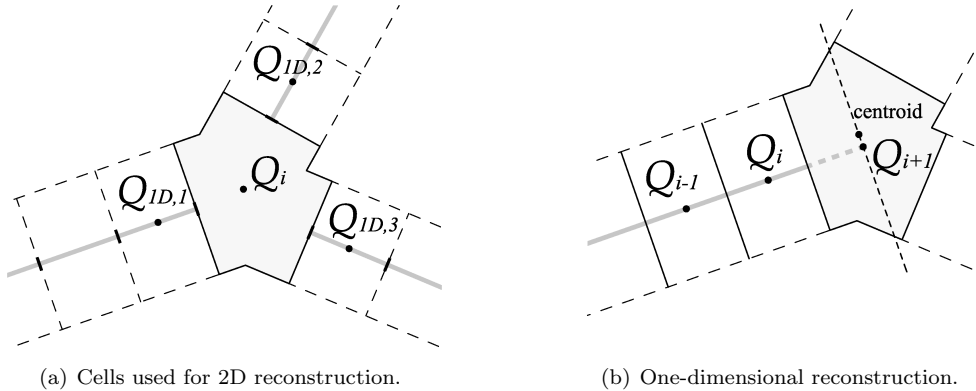


Figure 6: Illustration and notation for the spatial reconstruction in 2D (a) and 1D (b).

Concerning 2D elements, particular attention must be paid to the reconstruction process. As in the 1D case, at any given time level n one has a set of constant volume averages that are approximations to integral averages within each finite volume. For a second-order scheme, we need to approximate the solution in the 2D element with a first order polynomial. To this end we need three equations, for which we consider the three neighbouring 1D cells, as shown in Fig. 6(a). We do not use fictitious elements near reflective boundaries for the reconstruction.

The 1D reconstruction delivers a slope in the axial direction, while 2D reconstructions results in slopes in x - and y -directions. When passing from 1D to 2D or vice versa we need to transform the first into the second, so we have to rotate not only the vector of conserved variables, but also the gradients. In fact, in the 1D domain we have $\partial_n u$ and $\partial_n v$, but to apply Harten's approach to solve the generalized Riemann problem we need $\partial_x U$, $\partial_y U$, $\partial_x V$ and $\partial_y V$ (being u and v velocities in axial and transversal directions, and U and V velocity components in x and y directions). These slopes can be calculated as

$$\begin{pmatrix} \partial_x u \\ \partial_y u \end{pmatrix} = \begin{pmatrix} \cos \alpha \\ \sin \alpha \end{pmatrix} \partial_n u, \quad \begin{pmatrix} \partial_x v \\ \partial_y v \end{pmatrix} = \begin{pmatrix} \cos \alpha \\ \sin \alpha \end{pmatrix} \partial_n v \quad (19)$$

and

$$\begin{pmatrix} \partial_x U \\ \partial_x V \end{pmatrix} = \begin{bmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{bmatrix} \begin{pmatrix} \partial_x u \\ \partial_x v \end{pmatrix}, \quad (20)$$

$$\begin{pmatrix} \partial_y U \\ \partial_y V \end{pmatrix} = \begin{bmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{bmatrix} \begin{pmatrix} \partial_y u \\ \partial_y v \end{pmatrix}.$$

In the next section we assess the performance of the junction methods presented in this paper using a comprehensive suite of test problems, comparing results to 2D reference solutions obtained from an unstructured 2D second-order method discribed in [Appendix B](#).

4. Test problems and assesment of the methods

In this paper we consider three methods to deal with junctions in the context of shallow-water channels, **Method A** being our main contribution, in which a single 2D element is inserted in each junction. **Method B** generalises **method A** by inserting a local 2D unstructured grid in the vicinity of each junction, see Fig. 7. The third method considered for comparison is the method proposed by Peiró, Sherwin, Formaggia and Parker [35, 36], which in this paper will be called **PSFP method**. This method is summarised in [Appendix A](#). Solutions from all 3 methods are compared to reference 2D solutions. All methods have been implemented to second-order accuracy in both space and time. Here we present results for five tests. For additional tests see [8].

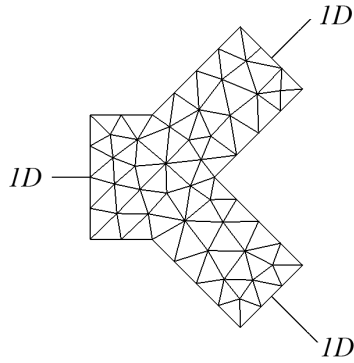


Figure 7: Example of a local 2D grid used in the vicinity of the junction region in **Method B**.

4.1. Single-junction test problems

Test 1: Subcritical wave in channel with a 90° bifurcation.

In this test we consider a channel configuration as shown on the left of Fig. 8. We impose a subcritical wave ($Fr_{max} \simeq 0.4$) that gradually steepens up and becomes a shock wave just after a 90° bifurcation. Results are shown in Fig. 8. Method A and B give very satisfactory results, as compared to the 2D reference solution, for channel 1, while the PSFP method gives rather inaccurate results. For channel 2 all three methods give quite similar results, methods A and B being slightly more accurate than the PSFP method.

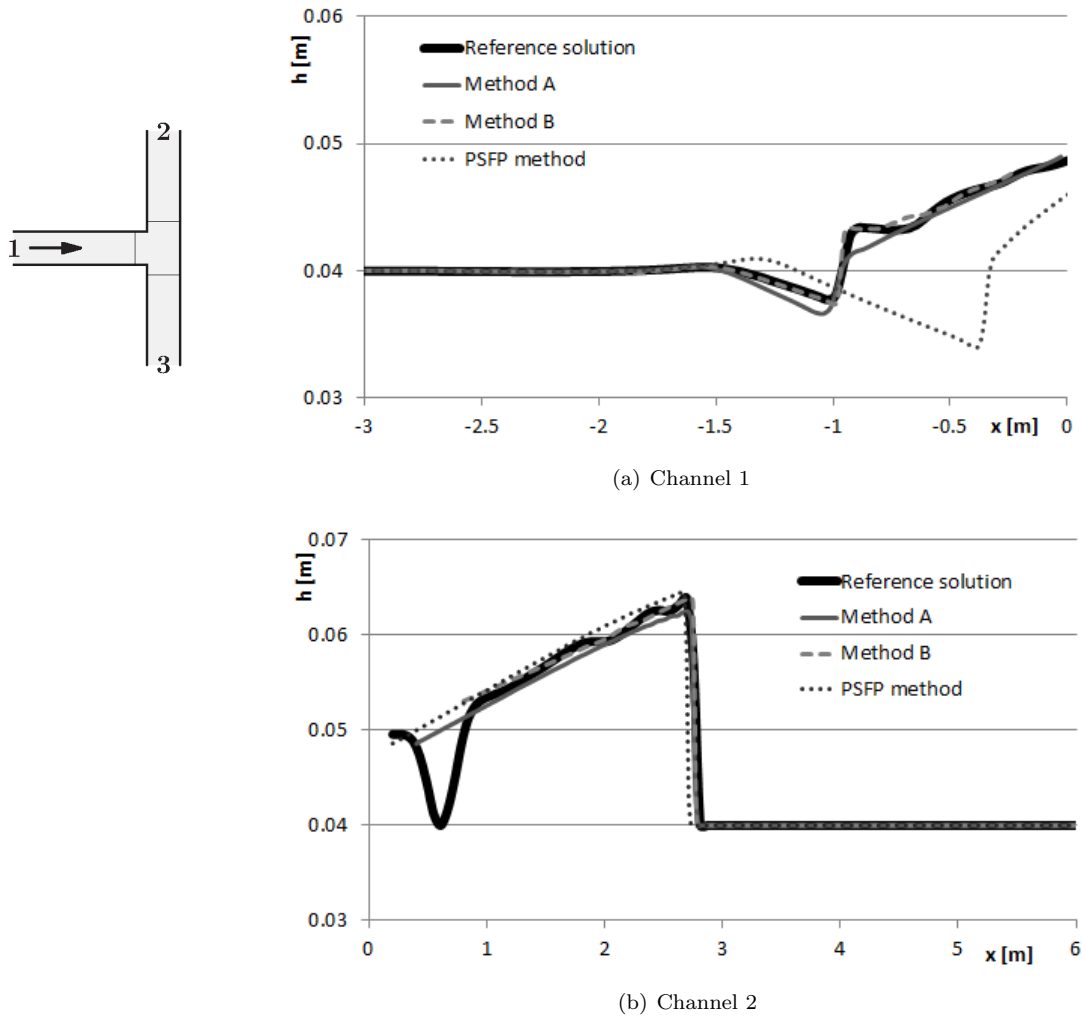


Figure 8: Test 1: Subcritical wave. Water height at time $t = 8$ s.

Test 2: Subcritical wave in a channel with a 90° asymmetrical bifurcation.

In this test we consider an asymmetrical channel configuration as shown on the left of Fig. 9. As for the previous test, methods A and B give very satisfactory results as compared to the reference 2D solution, outperforming the PSFP method.

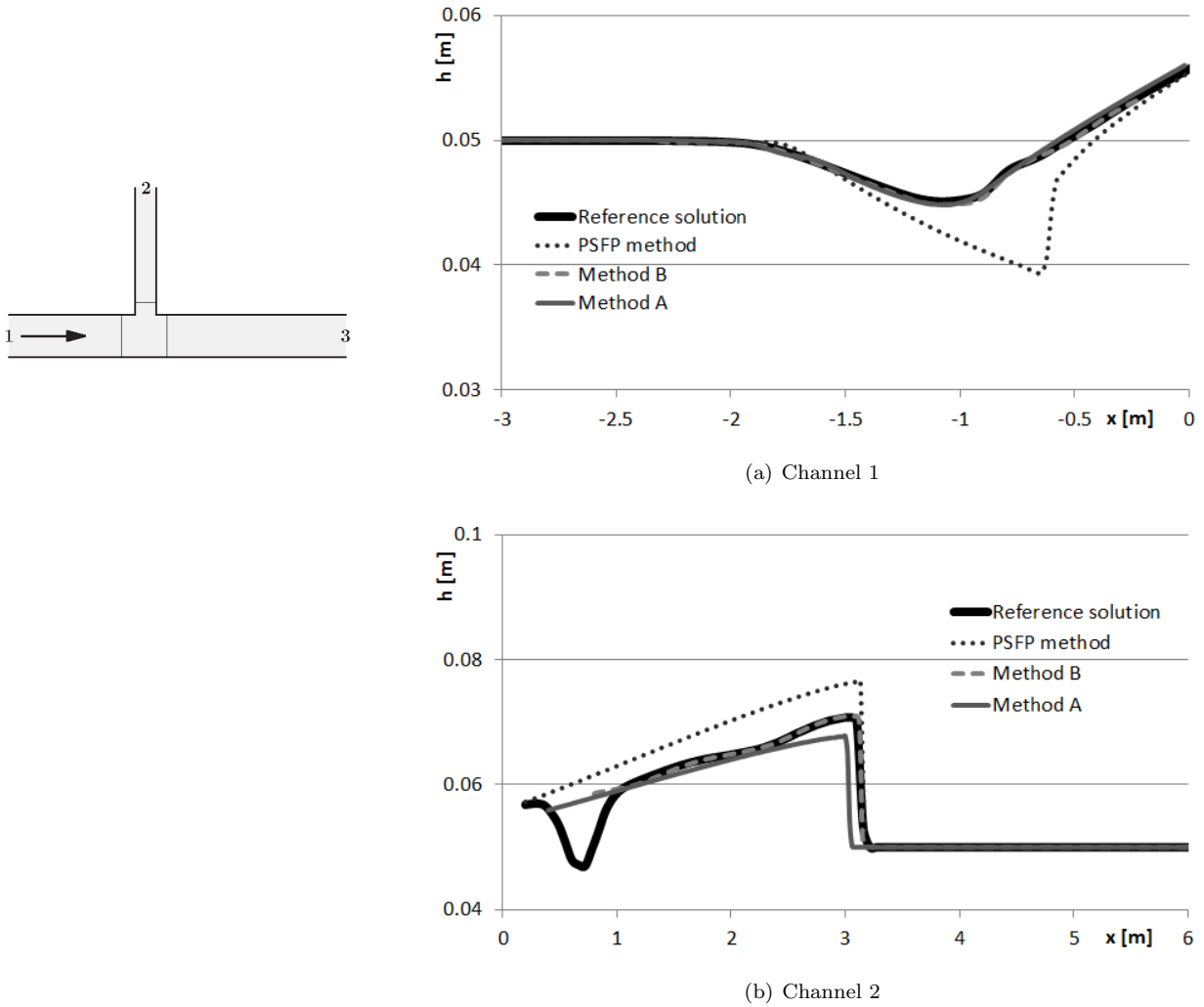


Figure 9: Test 2: Subcritical wave (asymmetrical case): Water height at time $t = 8$ s.

Test 3: Shock wave in a channel with a 45° bifurcation.

In this test we consider a channel configuration as shown on the left of Fig. 11. From channel 1 we send a shock with Froude number $Fr = 0.75$. Results are shown in Fig. 11. It is seen that the performance of methods A and B is very satisfactory, as far as the shock wave is concerned. The PSFP method did not run for this test.

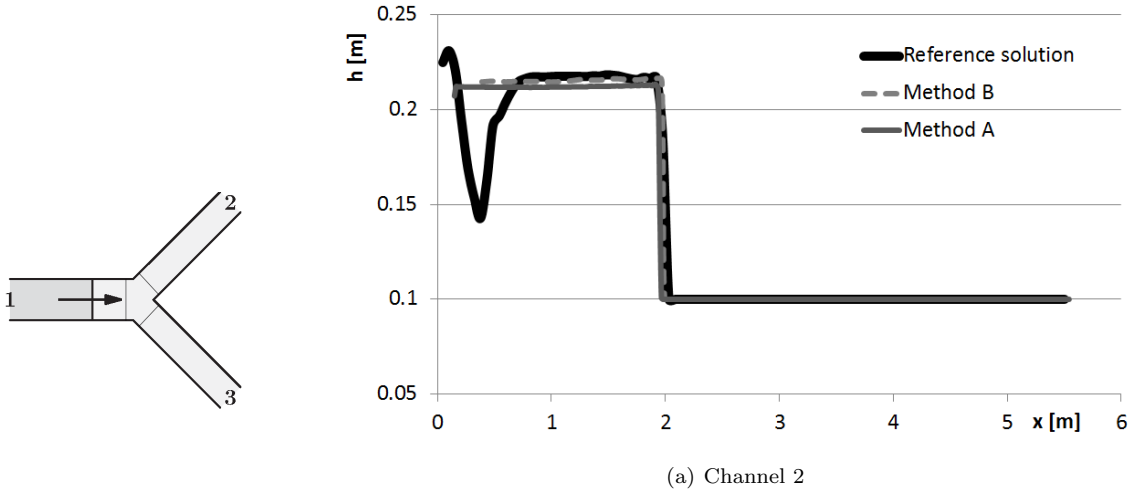


Figure 10: Test 3: Supercritical shock wave (45°). Water height at time $t = 2$ s.

Test 4: Supercritical shock wave in a channel with a 90° bifurcation.

Finally we test our methods with a severe problem: a supercritical shock of Froude number $Fr = 1.135$. Results are shown in Fig. 11. Results obtained with method B are again very satisfactory, thanks to the local 2D grid. On the other hand, results obtained with method A are less accurate than that obtained in the previous case, because of the severity of the test. Again, the PSFP method did not run for this test.

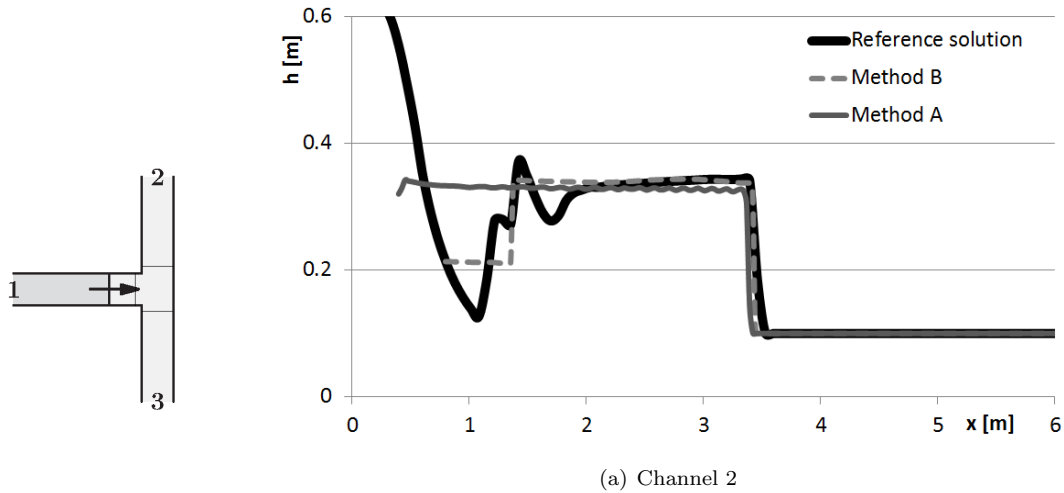


Figure 11: Test 4: Supercritical shock wave (90° bifurcation). Water height at time $t = 2$ s.

4.2. Test 5: the CADAM test problem.

In this section we apply the methods to the CADAM test 1 (CADAM, Concerted Action on Dam-Break Modelling, 1996-1999), for which experimental measurements are available as well as numerous numerical simulations. For a full description of the test see [27]. The geometrical configuration is depicted in Fig. 12 in which a 2D reservoir is connected to a straight channel with a 45° bend. Figs. 12(a) and 12(b) show how the 45° bend was treated for methods A and B. In both cases the reservoir is discretised with a 2D

unstructured mesh, while for the 45° bend method B inserts a local 2D grid in the vicinity of the bend while method A considers a single 2D element.

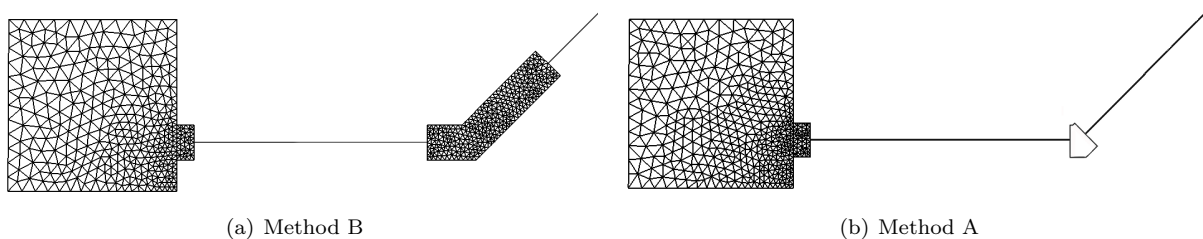
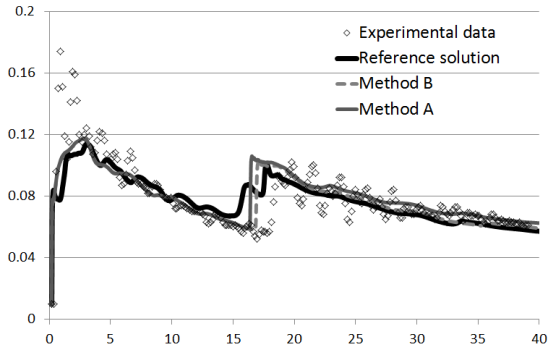
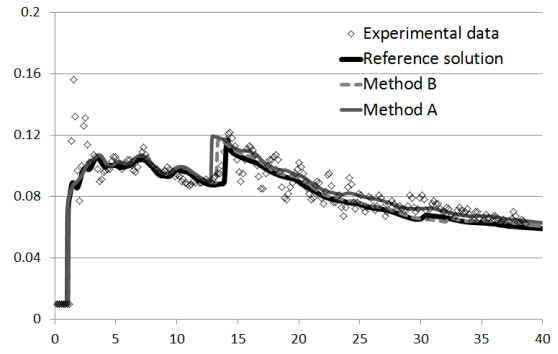


Figure 12: Test 5: the CADAM test problem. 2D and 1D domains used for numerical simulation of CADAM test 1.

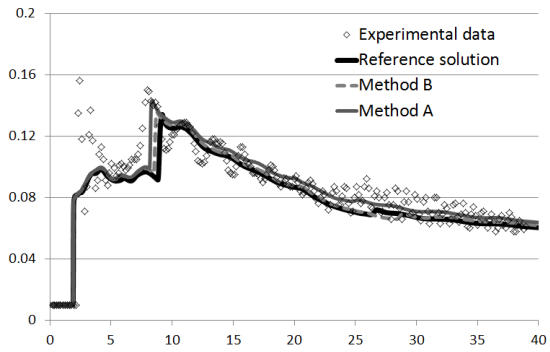
In the CADAM experiment, measuring gauges 5 to 7 are placed around the bend, where the motion of the fluid is more complex. Gauges 2, 3, 4 and 9 are placed along the straight channels. For full details see [27]. Numerical results and experimental measurements are all displayed in Fig 13. Results obtained with the methods proposed in this paper compare satisfactorily to measurements. The flow is supercritical, so the PSFP method did not run for this test.



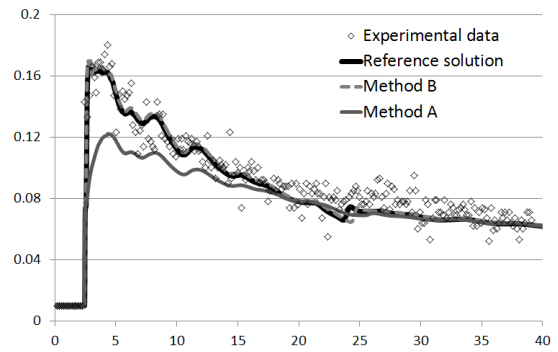
(a) Gauge 2



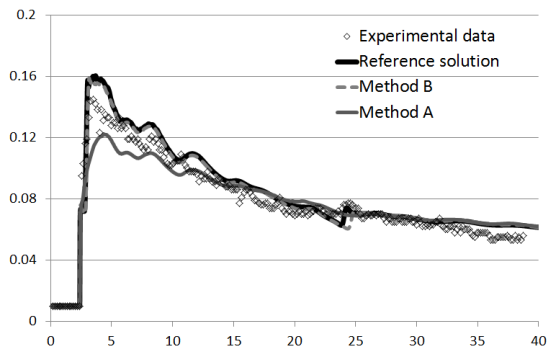
(b) Gauge 3



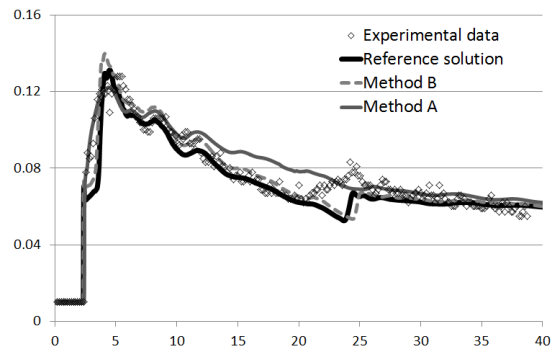
(c) Gauge 4



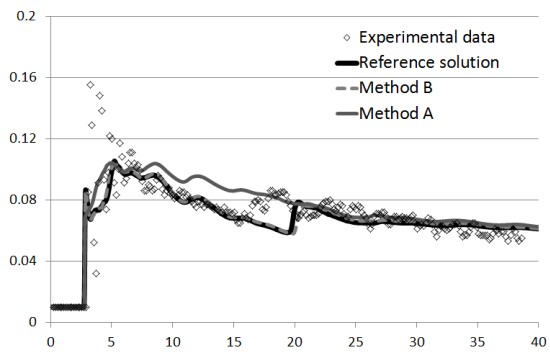
(d) Gauge 5



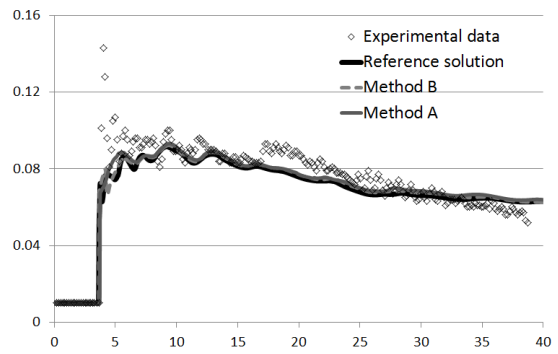
(e) Gauge 6



(f) Gauge 7



(g) Gauge 8



(h) Gauge 9

Figure 13: Test 5: the CADAM test problem. Computed free-surface elevation [meters] in time [seconds] and experimental measurements. Gauges 2 to 9 are the points of measurement used in the experimental test [27].

4.3. Test 6: A multiple-channel network

In this section we assess the performance of the various methods for the case of a multiple-channel network involving 16 junctions and 25 branches; see Figs. 14-16. We considered two cases, an incident subcritical wave and an incident supercritical shock. For the sake of simplicity we set the bed slope and the friction to zero. Solutions are computed with all three approximate junction methods considered, except for the supercritical shock case for which only methods A and B are used. For this test, due to the complexity of the situation with many shock wave reflections and wave interaction, for method A we use a 2D coarse grid inside the four junctions in the corners (see Fig. 15), where the flow is very complex due to large variations in angles and the large space occupied by the junction. Results will be shown at the eight positions shown in Fig. 16.

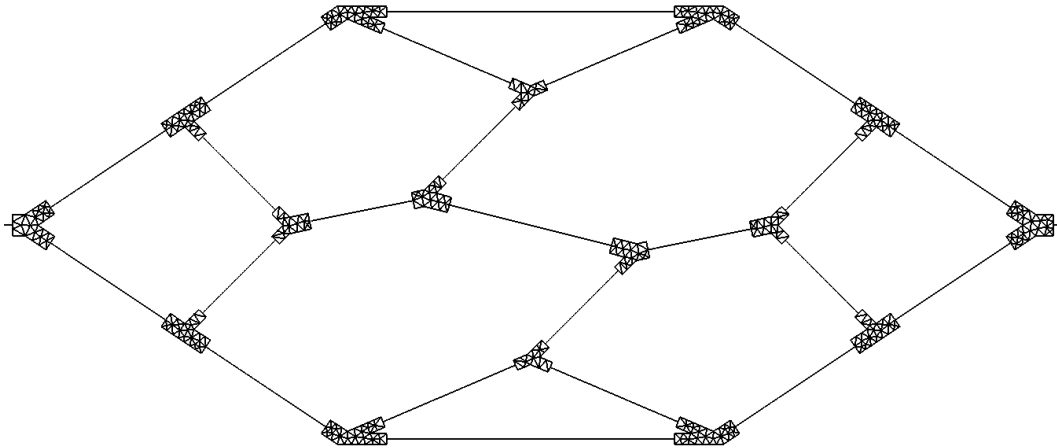


Figure 14: Test 6: A multiple-channel network. Configuration for method B. Two-dimensional grids in the vicinity of junctions.

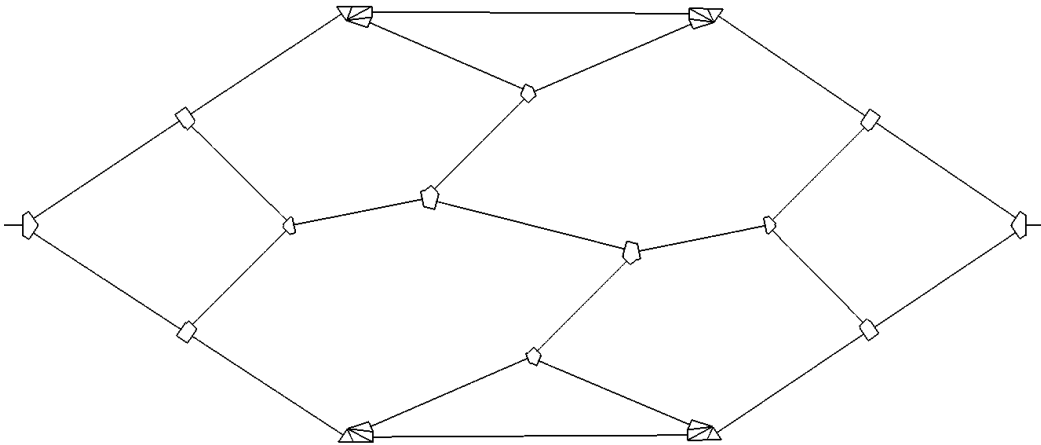


Figure 15: Test 6: A multiple-channel network. Method A. 2D single elements in most junctions, modified at shown four corner junctions.

For the subcritical wave case, computed results are displayed in Figs. 17 to 20. All three approximate junction methods run and are compared to the reference 2D solution. Methods A and B are seen to be very accurate; all three methods give very similar results for the arrival phase of the wave but differ at later times. For the supercritical shock wave case, computed results are displayed in Figs. 21 to 24. For this case the PSFP method did not run. Not surprisingly, it is seen that method B hardly differs from the reference 2D

solution but the simpler method A is also seen to be very accurate. As expected, the larger discrepancies between method A and the reference solution are seen in wave arrival times. Results at point 8 were expected to show the largest errors, as waves must transverse the full complex network, with multiples shock waves and complex interactions, and yet the end results at position 8 are satisfactory.

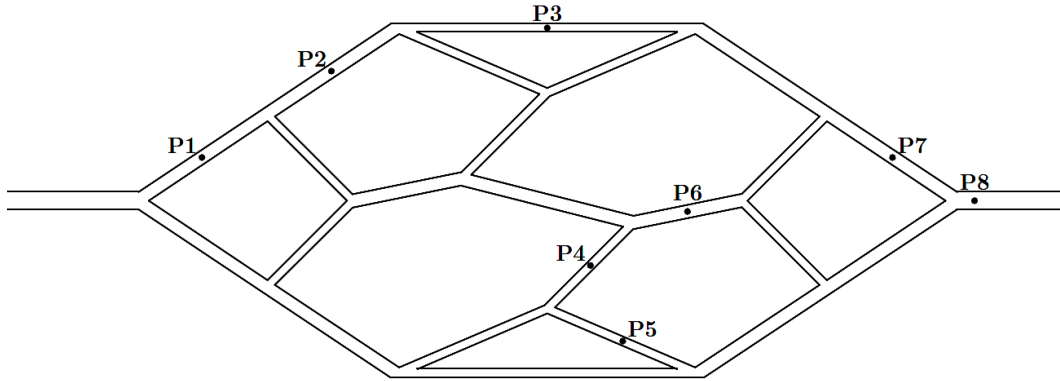


Figure 16: Test 6: A multiple-channel network. Points of the network where the free surface elevation is recorded and then reported in figures 17 to 24.

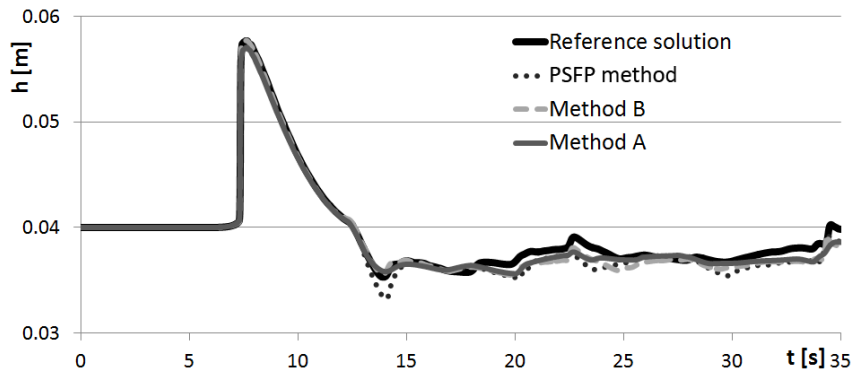


Figure 17: Test 6 (subcritical wave): A multiple-channel network. Computed free-surface elevation [m] in time [s] for Point 1.

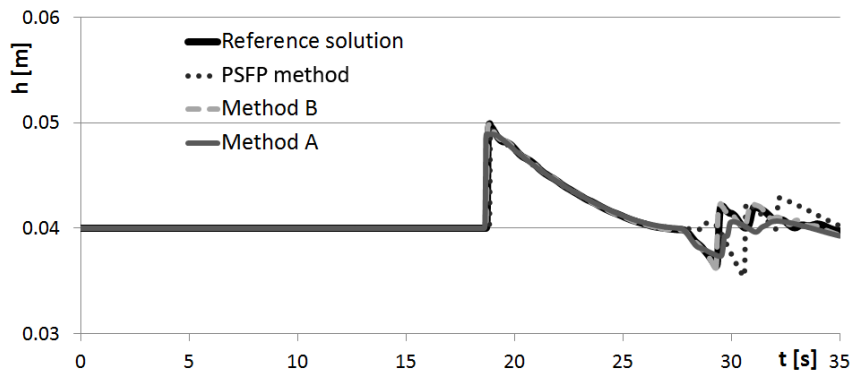


Figure 18: Test 6 (subcritical wave): Computed free-surface elevation [m] in time [s] for Point 3.

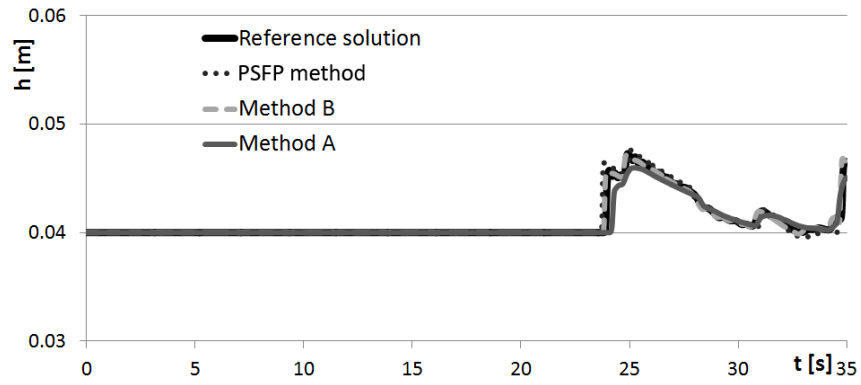


Figure 19: Test 6 (subcritical wave): Computed free-surface elevation [m] in time [s] for Point 6.

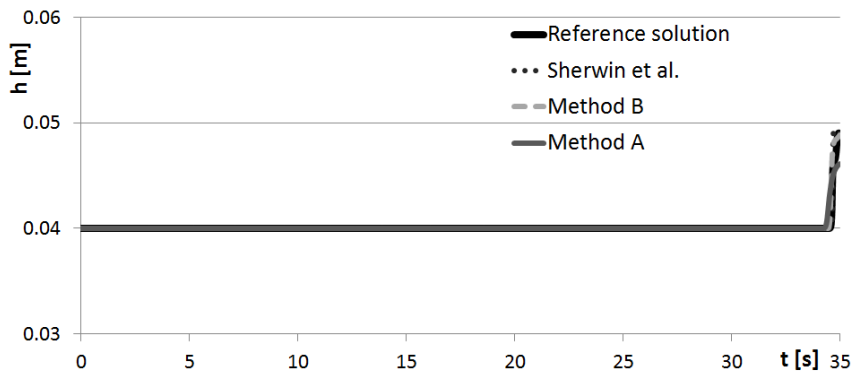


Figure 20: Test 6 (subcritical wave): Computed free-surface elevation [m] in time [s] for Point 8.

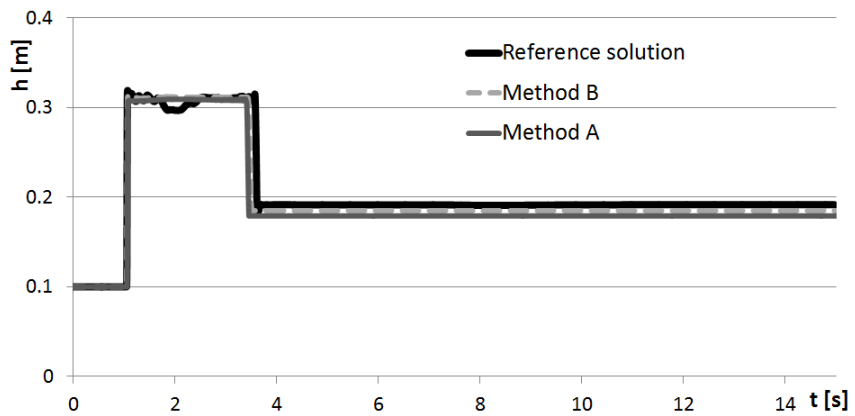


Figure 21: Test 6 (supercritical shock): Computed free-surface elevation [m] in time [s] for Point 1.

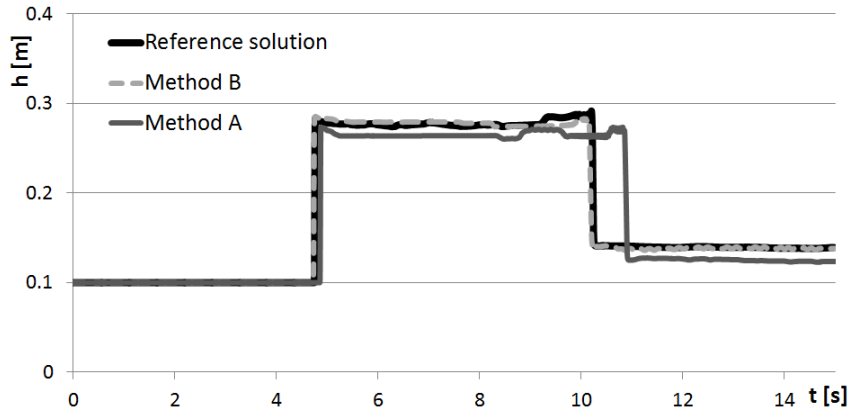


Figure 22: Computed free-surface elevation [m] in time [s] for Point 3. Supercritical shock.

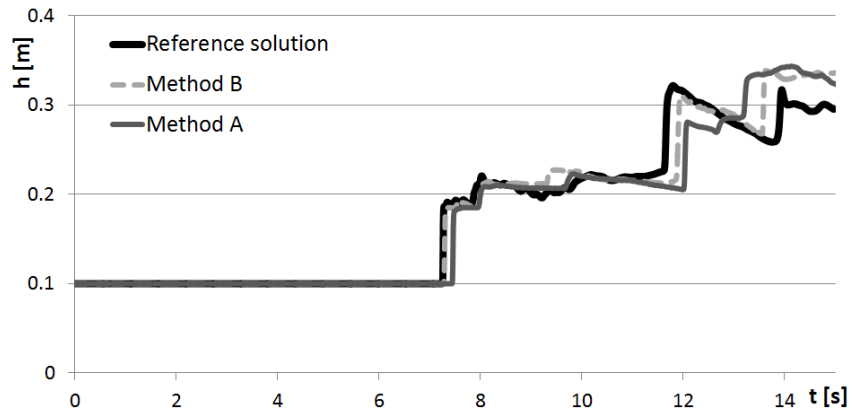


Figure 23: Test 6 (supercritical shock): Computed free-surface elevation [m] in time [s] for Point 6.

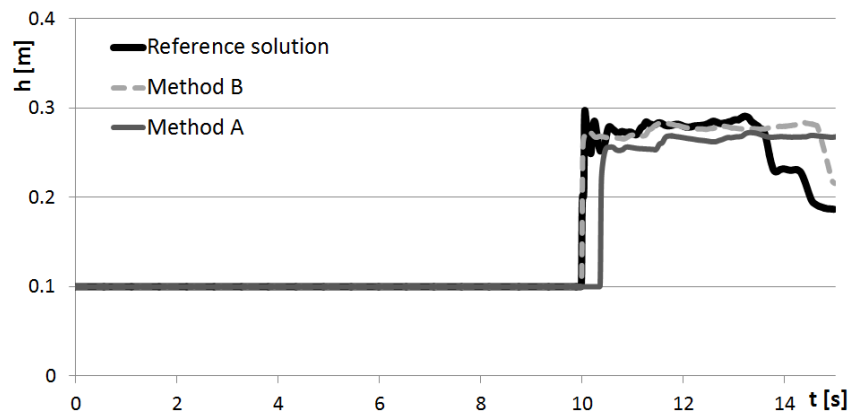


Figure 24: Test 6 (supercritical shock): Computed free-surface elevation [m] in time [s] for Point 8.

4.4. Computational times

Here we show the computational times involved to solve each one of the six problems previously presented. Table 1 shows the tests on the left columns and the CPU times in seconds in the subsequent columns, for the various methods used. Missing values for the PSFP method regard tests for which this method did not work.

Test	2D Reference	PSFP method	Method B	Method A
Test 1	392.1	3.31	34.8	1.08
Test 2	547.2	3.28	31.7	2.06
Test 3	1215	-	68.9	3.53
Test 4	1684	-	128.3	4.69
Test 6 (subcritical wave)	5787	70.3	1413	19.2
Test 6 (supercritical shock)	13775	-	3091	51.5

Table 1: Computational times [s] for all numerical methods reported in this paper, for six test problems.

As expected, the largest CPU times are those for the full 2D solver used to produce reference solutions. In terms of cost, there follows method B. The next one in CPU time cost is the PSFP method, with Method A being the fastest, even faster than the simplest of all methods, namely the PSFP method, which is based entirely on 1D assumptions. It appears as if the method of choice is our method A, since it runs for all very demanding test problems, while giving reasonably accurate solutions as compared to the full 2D solver and at lowest computational cost. Computational saving factors for method A, relative to the full 2D solver, are of the order of 300, making the method a realistic option for complex applications.

5. Concluding remarks

We have presented a novel method to treat junctions in networks of 1D shallow water channels. The method, called method A, inserts a single 2D, junction-shaped finite volume right at the junction, taking care that the element protrudes into the 1D channels. In this manner, the geometrical information, such as bifurcation angles and reflective boundaries is accounted for, locally. Method B results from generalising method A by inserting a local 2D unstructured grid in the vicinity of the junction. In addition, we briefly reviewed the existing junction method due to Peiró, Sherwin, Formaggia and Parker [35, 36], which we termed PSFP method. All three approximate junction methods are assessed through a carefully selected suite of demanding test problems. No exact solutions to these problems exist to test the accuracy of approximate junction methods. We therefore use a fully 2D unstructured-mesh, second order method of the ADER type to compute accurate numerical solutions. Method A is the preferred one, since it is simple and sufficiently accurate for all test problems. Method B is the more accurate of all three approximate methods tested, but also the most expensive, as shown by our computational efficiency test. Method A is the fastest, about three times faster than the PSFP method and about 70 times faster than method B for the more realistic test problem involving a reasonably complex network. Methods A and B work well for all test problem, while the PSFP method only works for 3 of the 6 test problems. An attractive feature of method A, shared by method B, is that it can successfully cope with problems involving high subcritical, transcritical and supercritical flows at the junctions. We note that due to the single-element of method A, accuracy may deteriorate, depending on the mesh dimensions involved. This shortcoming is most evident in the first-order version of the methodology. Higher-order versions can ameliorate this deficiency. In fact, second order accuracy is found to be satisfactory, though we found a test problem for which only the third-order scheme produced fully satisfactory solutions, not shown here. Potential users of the schemes may have to assess this aspect of the methods before embarking on practical applications. For practical applications, both methods A and B may benefit from using local time-stepping, for example, following the methodology proposed in [16, 30]. This may be required by the disparity of spatial mesh sizes at the junctions and the 1D domains, which potentially implies disparity in time step sizes.

The methods presented in this paper can be applied to any problem involving networks of nearly straight 1D domains, provided the multidimensional version of the equations, 2D or 3D, are available.

Acknowledgements

The authors are indebted to Prof. Dr. M. Dumbser, University of Trento, for useful discussions on the subject.

Appendix A. Existing method for junctions: the PSFP method

Here we briefly present an existing method for dealing with junctions, which we attribute to the work of J. Peiró, S. Sherwin, L. Formaggia and K. Parker, see [35] and [36]. The methodology has become exceedingly popular and has been heavily applied in cardiovascular mathematics to deal with junctions of blood vessels. Here we adapt the method to deal with junctions in networks of water channels

Appendix A.1. Brief description of the method for shallow water

Consider the model bifurcation configuration shown in figure 25(a), where we denote the parent channel by index 1 and the daughter channels by indices 2 and 3. We divide the domain composed by the three 1D channels into three sub-domains joined at the junction. In order to solve the junction problem and connect the 3 channels, a purely one-dimensional approach is taken to compute the needed 1D intercell numerical fluxes.

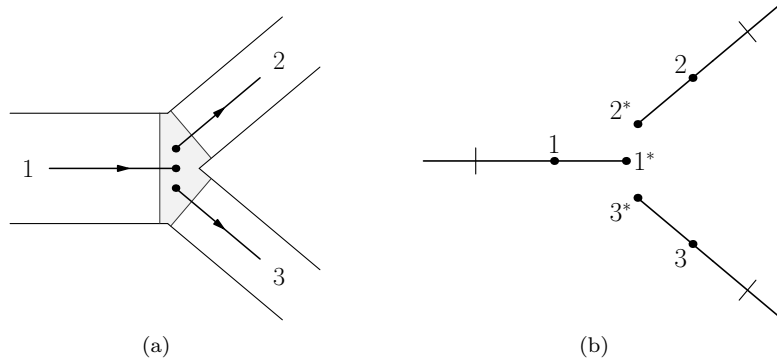


Figure A.25: Junction problem as dealt with by the PSFP method. Frame (a): Idealised junction connecting a parent channel 1 to two daughter channels 1 and 2. Frame (b): notation setup to solve the problem via 3 Riemann problems.

At the bifurcation we have six unknowns, namely (h_1^*, u_1^*) , (h_2^*, u_2^*) and (h_3^*, u_3^*) . Once these are determined, corresponding numerical fluxes at the appropriate boundaries are computed, see figure 25(b). Such fluxes will be used to update the 1D elements 1, 2 and 3.

The method assumes subcritical conditions and that the Riemann problem solutions involve only rarefaction waves. This allows the application of Riemann invariants, leading to 3 algebraic equations

$$\begin{aligned} u_1^* + 2\sqrt{gh_1^*} &= u_1 + 2\sqrt{gh_1} , \\ u_2^* - 2\sqrt{gh_2^*} &= u_2 - 2\sqrt{gh_2} , \\ u_3^* \mp 2\sqrt{gh_3^*} &= u_3 \mp 2\sqrt{gh_3} , \end{aligned} \tag{A.1}$$

where the minus or plus sign in the third equation depends on the type of junction (diverging or merging).

Three more algebraic equations are

$$\begin{aligned} h_1^* u_1^* b_1 &= h_2^* u_2^* b_2 + h_3^* u_3^* b_3 , \\ h_1^* + \frac{u_1^{*2}}{2g} &= h_2^* + \frac{u_2^{*2}}{2g} , \\ h_1^* + \frac{u_1^{*2}}{2g} &= h_3^* + \frac{u_3^{*2}}{2g} . \end{aligned} \tag{A.2}$$

The first of these represents mass conservation through the bifurcation, while the last two conditions are obtained from the requirement of continuity of total energy at the bifurcation. Conditions (A.1) and (A.2) constitute a system of six non-linear algebraic equations; these equations are solved by a Newton method.

The described PSFP method requires the solution of a 6×6 non-linear algebraic system at each junction at each time step and represents a substantial computational effort in a complex network. However the method is indeed very simple and performs well under subcritical flow conditions with relatively small Froude numbers. Sharing our experience, a word of caution is due. The method does not acknowledge the geometrical features of the problem and consequently, works better for small bifurcation angles and symmetric cases. For supercritical and subcritical flows with relatively high Froude numbers the method may actually fail. Below we report a subcritical example with the following data:

$$\begin{aligned} b_1 &= 0.4 \text{ m} , & h_1 &= 0.2 \text{ m} , & u_1 &= 0.96 \text{ m/s} , \\ b_2 &= 0.3 \text{ m} , & h_2 &= 0.1 \text{ m} , & u_2 &= 0.08 \text{ m/s} , \\ b_3 &= 0.3 \text{ m} , & h_3 &= 0.1 \text{ m} , & u_3 &= 0.08 \text{ m/s} . \end{aligned} \tag{A.3}$$

Froude number in channel 1 is $Fr_1 = 0.685$ and in channel 2 and 3 is $Fr_2 = Fr_3 = 0.081$, so the regime is clearly subcritical in every channel. Through the use of MAPLE we found two analytical solutions of system (A.1)-(A.2), namely:

$$\begin{aligned} h_1^* &= 0.155 - 0.0660I \text{ m} , \\ h_2^* &= 0.186 - 0.0006I \text{ m} , \\ h_3^* &= 0.186 - 0.0006I \text{ m} , \\ u_1^* &= 1.240 + 0.5138I \text{ m/s} , \\ u_2^* &= 0.807 - 0.0047I \text{ m/s} , \\ u_3^* &= 0.807 - 0.0047I \text{ m/s} , \end{aligned} \tag{A.4}$$

and

$$\begin{aligned} h_1^* &= 0.155 + 0.0660I \text{ m} , \\ h_2^* &= 0.186 + 0.0006I \text{ m} , \\ h_3^* &= 0.186 + 0.0006I \text{ m} , \\ u_1^* &= 1.240 - 0.5138I \text{ m/s} , \\ u_2^* &= 0.807 + 0.0047I \text{ m/s} , \\ u_3^* &= 0.807 + 0.0047I \text{ m/s} . \end{aligned} \tag{A.5}$$

Both solutions are complex; needless to say, the Newton method did not converge.

Appendix A.2. Test cases for the PSFP method

In order to test the PSFP method we performed various tests consisting of a subcritical wave propagating across junctions of different geometry. We set initial conditions of zero velocity everywhere and water height of 16 cm . Boundary conditions at the outflow of daughter channels are transparent and velocity was assigned at the inflow of the parent channel by the relation

$$u(t) = 0.4 \exp(-0.5(t - 3)^2) . \tag{A.6}$$

To guarantee no wave reflection at the inflow, we used the method applied in [2]. Note that the flow regime of this test is subcritical, the maximum Froude number is about 0.5.

First we performed a test with a straight channel of width 0.4 m which bifurcates into two channels of width 0.2 m each placed at an angle of 0° , as displayed in figure 26(a). Then we tested the method with the same wave as above propagating through a channel which bifurcates into two larger channels placed at an angle of 0° , as displayed in figure 26(b). Finally we performed a test with a situation similar to the previous one, but asymmetrical, as displayed in figure 26(c). Figure A.26 shows computed results, comparing results with the reference 2D solution; water height profiles obtained in the secondary channels at time $t = 6 \text{ s}$ are shown. Results worsen as the test problem depart from symmetry.

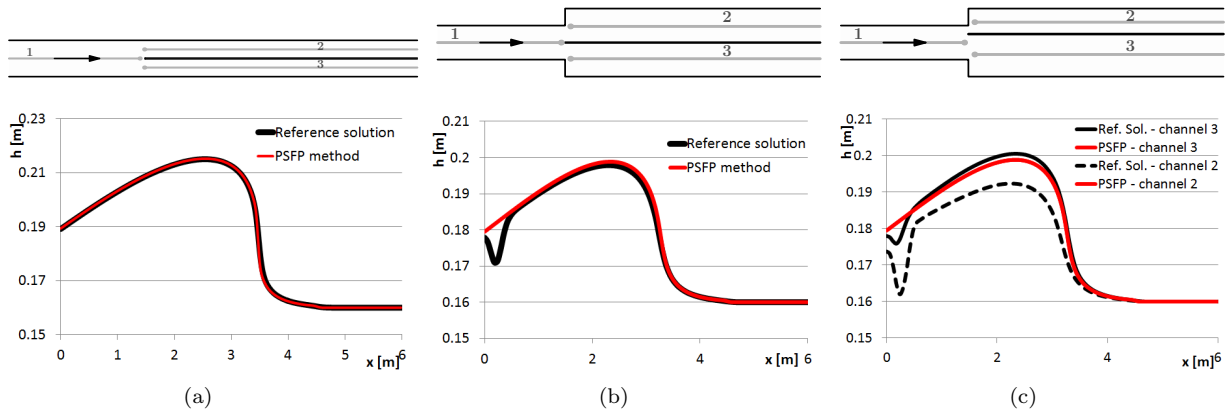


Figure A.26: Performance of the PSFP junction method applied to problems of a subcritical wave propagating across straight channels. The PSFP-based 1D solutions are compared to the 2D reference solutions.

We also tested the method for tests with bifurcation angles different from zero (figure A.27). Initial conditions are the same as for the previous problems.

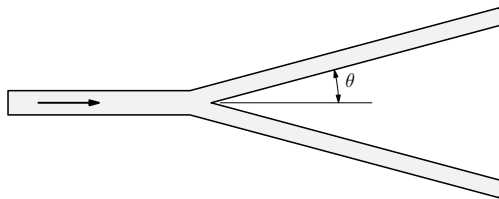


Figure A.27: Example of a symmetric bifurcation with bifurcation angle θ .

We tested three bifurcations, with angles of 15° , 45° and 90° . Figures 26(a), 26(b) and 26(c) compare the water height profiles obtained in the primary channels at a time $t = 8$ s. Figures 26(a), 26(b) and 26(c) compare the water height profiles in the secondary channels to the reference 2D solution.

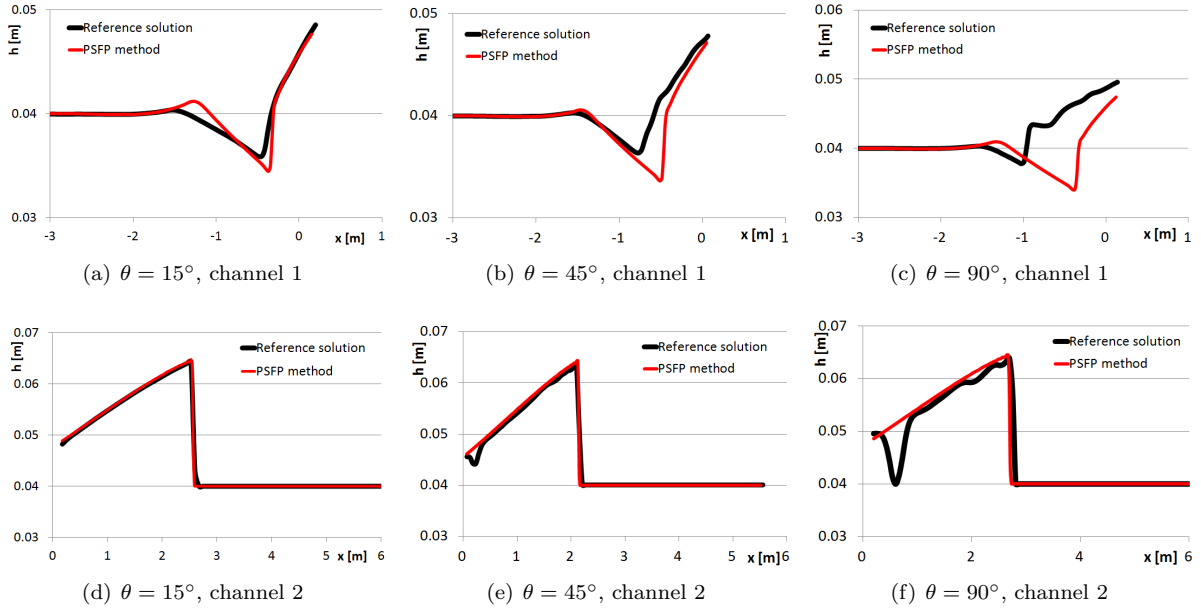


Figure A.28: Performance of the PSFP junction method applied to a problem of a subcritical wave propagating across junctions of different bifurcation angles θ . PSFP-based 1D solutions are compared to 2D reference solutions.

As the bifurcation angle increases, the solution in channel 1 obtained with the one-dimensional model based on the PSFP method, differs more and more from the reference two-dimensional solution. We note that solutions in channels 2 and 3 seem to be less sensitive to increased angle values.

Appendix B. A reference two-dimensional solver

Here we give some details of our 2D shallow water solver used to compute reference solutions to assess the junction models. Departing from equations (1-2) and making use of the rotational invariance we construct a finite volume method on unstructured triangular meshes, written in the form

$$\mathbf{Q}_k^{n+1} = \mathbf{Q}_k^n - \frac{\Delta t}{|V_k|} \sum_{e=1}^N \mathcal{L}_e \mathbf{T}_e^{-1} \hat{\mathbf{F}}_e . \quad (\text{B.1})$$

where the flux $\hat{\mathbf{F}}_e$ for edge e is computed as discussed in Sect. 1. The scheme is second-order accurate in space and time and its construction follows the ADER approach [38]. The first-order component of the method uses the HLLC flux [37]; the high-order version requires (i) a spatial nonlinear reconstruction procedure and (ii) the solution of the generalised Riemann problem. These steps are described in more detail below.

Spatial reconstruction. For the spatial reconstruction procedure on triangular meshes we follow [7]; as a matter of fact we also use this procedure for 1D domains. For the 2D case, for the three triangles surrounding a general element we write the equation

$$\mathbf{Q}_{J(k)}^n = a + b(x_{J(k)} - x_j) + c(y_{J(k)} - y_j) , \quad (\text{B.2})$$

where j is the global index and k the local index of each neighbour; $x_{J(k)}$ and $y_{J(k)}$ the coordinates of the centroids of the neighbours. Once this three-equation system is solved, in order to respect the conservation property we have to force the coefficient a to satisfy $a = \mathbf{Q}_i^n$. Coefficients b and c represent the best estimates of the solution gradient. We have to impose a limiter on this gradient in order to respect the non-oscillatory property. Barth and Jespersen [7] suggest to find the largest admissible Φ so that the values of the linearly reconstructed function do not exceed the maximum and minimum of neighbouring centroid values (including the centroid value in the element). Then, keeping in mind that for linear reconstructions extrema in a triangle occur at the vertices, we can calculate the limiter Φ as the minimum ϕ_k found among all vertices, and finally multiply b and c by Φ . We perform this procedure for every conserved variable.

Generalised Riemann problem. There exist several methods to solve the Generalised Riemann Problem (GRP), some of which are described in the paper by Castro and Toro [14]. Here we have implemented the method proposed by Harten and collaborators in [20], reinterpreted in [14] as a solver for the GRP. The method evolves the Riemann problem data \mathbf{Q}_L and \mathbf{Q}_R considering power series expansions in time on each side of the interface. For a second order scheme we need first order time series expansions, as follows:

$$\begin{aligned} \tilde{\mathbf{Q}}_L\left(\frac{\Delta t}{2}\right) &= \mathbf{Q}_L(0_-) + \frac{\Delta t}{2} \partial_t \mathbf{Q}(0_-, 0) , \\ \tilde{\mathbf{Q}}_R\left(\frac{\Delta t}{2}\right) &= \mathbf{Q}_R(0_+) + \frac{\Delta t}{2} \partial_t \mathbf{Q}(0_+, 0) . \end{aligned} \quad (\text{B.3})$$

Using the Cauchy-Kowalewskaya procedure to express time derivatives as functions of space derivatives, and given the slopes resulting from second order reconstruction, we have:

$$\begin{aligned} \partial_t \mathbf{Q}(0_-, 0) &= -\mathbf{A}(\mathbf{Q}(0_-, 0)) \cdot \mathbf{b}(\mathbf{Q}(0_-, 0)) - \mathbf{B}(\mathbf{Q}(0_-, 0)) \cdot \mathbf{c}(\mathbf{Q}(0_-, 0)) , \\ \partial_t \mathbf{Q}(0_+, 0) &= -\mathbf{A}(\mathbf{Q}(0_+, 0)) \cdot \mathbf{b}(\mathbf{Q}(0_+, 0)) - \mathbf{B}(\mathbf{Q}(0_+, 0)) \cdot \mathbf{c}(\mathbf{Q}(0_+, 0)) . \end{aligned} \quad (\text{B.4})$$

Then we apply the rotational matrix \mathbf{T}_s on $\tilde{\mathbf{Q}}_L$ and $\tilde{\mathbf{Q}}_R$, obtaining $\hat{\tilde{\mathbf{Q}}}_L$ and $\hat{\tilde{\mathbf{Q}}}_R$, and we compute the numerical flux from the solution of a classical Riemann problem, for which we use the HLLC approximation [37]. The resulting method is second-order accurate in space and time.

Grid-independence test. In order to choose a mesh size that gives a reliable reference solution, we performed a grid-independence test. We used a typical problem of a shock wave propagating across a junction.

For this problem we stored the computed free-surface elevation in time for a representative point placed where the two-dimensional character of the problem is more evident. Then we run the simulation with five different mesh sizes, each time halving the element size. We calculated the integral in time for each mesh and then we computed the change relative to the previous mesh. Results are shown in table B.2. We consider the difference obtained for the mesh of size 0.02 as being satisfactory. For all the computations to provide reference 2D solutions we used this mesh size.

Grid	Integrals	Relative differences	CPU times
0.16	0.3767	0.0502	2.8 s
0.08	0.3966	0.1216	22.3 s
0.04	0.4515	0.0046	129.1 s
0.02	0.4536	0.0015	1215 s
0.01	0.4543		9528 s

Table B.2: Variation of integrals of water height as the mesh is refined.

References

- [1] Akan, A. O. and Yen, B. C. Diffusion-wave flood routing in channel networks. *Journal of the Hydraulics Division*, 107(6): 719–732, 1981.
- [2] Alastruey, J. *Numerical modelling of pulse wave propagation in the cardiovascular system: development, validation and clinical applications*. PhD thesis, Departments of Aeronautics and Bioengineering, Imperial College London, 2006.
- [3] Alastruey, J., Khir, A. W., Matthys, K. S., Segers, P., Sherwin, S. J., Verdonck, P. R., Parker, K. H., and Peiró, J. Pulse wave propagation in a model human arterial network: Assessment of 1-d visco-elastic simulations against in vitro measurements. *Journal of Biomechanics*, 44(12):2250–2258, 2011. doi: 10.1016/j.jbiomech.2011.05.041. URL <http://dx.doi.org/10.1016/j.jbiomech.2011.05.041>.
- [4] Aral, M. M., Zhang, Y., and Jin, S. Application of relaxation scheme to wave-propagation simulation in open-channel networks. *Journal of Hydraulic Engineering*, 124(11):1125–1133, 1998.
- [5] Bales, P., Kolb, O., and Lang, J. Hierarchical modelling and model adaptivity for gas flow on networks. In *Lecture Notes in Computer Science*, pages 337–346. Springer, 2009. doi: 10.1007/978-3-642-01970-8_33. URL http://dx.doi.org/10.1007/978-3-642-01970-8_33.
- [6] Banda, M., Herty, M., and Klar, A. Coupling conditions for gas networks governed by the isothermal euler equations. *Networks and Heterogeneous Media*, 1(2):295–314, 2006. doi: 10.3934/nhm.2006.1.295. URL <http://dx.doi.org/10.3934/nhm.2006.1.295>.
- [7] Barth, T. J. and Jespersen, D. C. The design and application of upwind schemes on unstructured meshes. In *27th Aerospace Sciences Meeting*, 1989.
- [8] Bellamoli, F. Computational methods for coupling 1d and 2d channel geometries for the shallow water equations. Master’s thesis, Università degli studi di Trento, 2012.
- [9] Bermúdez, A., López, X., and Vázquez-Cendon, M. Treating network junctions in finite volume solution of transient gas flow models. Pre-print submitted to *Journal of Computational Physics* 2016.
- [10] Borsche, R. and Klar, A. Flooding in urban drainage systems: coupling hyperbolic conservation laws for sewer systems and surface flow. *International Journal for Numerical Methods in Fluids*, 76(11):789–810, 2014. ISSN 0271-2091. doi: 10.1002/flid.3957. URL <http://dx.doi.org/10.1002/flid.3957>.
- [11] Borsche, R., Klar, A., Kühn, S., and Meurer, A. Coupling traffic flow networks to pedestrian motion. *Mathematical Models and Methods in Applied Sciences*, 24(02):359–380, feb 2014. doi: 10.1142/s0218202513400113. URL <http://dx.doi.org/10.1142/s0218202513400113>.
- [12] Bretti, G., Natalini, R., and Piccoli, B. Numerical algorithms for simulations of a traffic model on road networks. *Journal of Computational and Applied Mathematics*, 210(1-2):71–77, dec 2007. doi: 10.1016/j.cam.2006.10.057. URL <http://dx.doi.org/10.1016/j.cam.2006.10.057>.
- [13] Brouwer, J., Gasser, I., and Herty, M. Gas pipeline models revisited: Model hierarchies, nonisothermal models, and simulations of networks. *Multiscale Modeling & Simulation*, 9(2):601–623, apr 2011. doi: 10.1137/100813580. URL <http://dx.doi.org/10.1137/100813580>.
- [14] Castro, C. and Toro, E. Solvers for the high-order Riemann problem for hyperbolic balance laws. *Journal of Computational Physics*, 227:2481–2513, 2008.
- [15] Coclite, G. M., Garavello, M., and Piccoli, B. Traffic flow on a road network. *SIAM Journal on Mathematical Analysis*, 36(6):1862–1886, jan 2005. doi: 10.1137/s0036141004402683. URL <http://dx.doi.org/10.1137/s0036141004402683>.
- [16] Dumbser, M., Käser, M., and Toro, E. F. An Arbitrary High Order Discontinuous Galerkin Method for Elastic Waves on Unstructured Meshes V: Local Time Stepping and p -Adaptivity. *Geophysical Journal International*, 171:695–717, 2007.
- [17] Formaggia, L., Nobile, F., Quarteroni, A., and Veneziani, A. Multiscale modelling of the circulatory system: a preliminary analysis. *Computing and Visualization in Science*, 2:75–83, 1999.
- [18] Formaggia, L., Nobile, F., Quarteroni, A., and Veneziani, A. Multiscale modelling of the circulatory system: a preliminary analysis. *Computing and Visualization in Science*, 2(2-3):75–83, 1999. doi: 10.1007/s007910050030. URL <http://dx.doi.org/10.1007/s007910050030>.
- [19] Fullana, J.-M. and Zaleski, S. A branched one-dimensional model of vessel networks. *Journal of Fluid Mechanics*, 621: 183–204., 2009.
- [20] Harten, A., Engquist, B., Osher, S., and Chakravarty, S. Uniformly high order accuracy essentially non-oscillatory schemes III. *Journal of Computational Physics*, 71:231–303, 1987.
- [21] Kesserwani, G., Ghostine, R., Vazquez, J., Mosé, R., Abdallah, M., and Ghenaïm, A. Simulation of subcritical flow at open-channel junction. *Advances in Water Resources*, 31:287–297, 2008.
- [22] Kesserwani, G., Ghostine, R., Vazquez, J., Mosé, R., Abdallah, M., and Ghenaïm, A. Simulation of subcritical flow at open-channel junction. *Advances in Water Resources*, 31(2):287–297, feb 2008. doi: 10.1016/j.advwatres.2007.08.007. URL <http://dx.doi.org/10.1016/j.advwatres.2007.08.007>.
- [23] Liang, F., Senzaki, H., Kurishima, C., Sugimoto, K., Inuzuka, R., and Liu, H. Hemodynamic performance of the fontan circulation compared with a normal biventricular circulation: a computational model study. *American Journal of Physiology: Heart and Circulatory Physiology*, 307(7):H1056–H1072, jul 2014. doi: 10.1152/ajpheart.00245.2014. URL <http://dx.doi.org/10.1152/ajpheart.00245.2014>.
- [24] Liang, F., Takagi, S., Himeno, R., and Liu, H. Multi-scale modeling of the human cardiovascular system with applications to aortic valvular and arterial stenoses. *Medical & Biological Engineering & Computing*, 47(7):743–755, feb 2009. doi: 10.1007/s11517-009-0449-9. URL <http://dx.doi.org/10.1007/s11517-009-0449-9>.
- [25] Liang, F., Takagi, S., Himeno, R., and Liu, H. Biomechanical characterization of ventricular–arterial coupling during aging:

- A multi-scale model study. *Journal of Biomechanics*, 42(6):692–704, apr 2009. doi: 10.1016/j.jbiomech.2009.01.010. URL <http://dx.doi.org/10.1016/j.jbiomech.2009.01.010>.
- [26] Miglio, E., Perotto, S., and Saleri, F. Model coupling techniques for free-surface flow problems: Part I. *Nonlinear Analysis: Theory, Methods and Applications*, 63:1885–1896, 2005. ISSN 0362-546X. doi: 10.1016/j.na.2005.03.083. URL <http://www.sciencedirect.com/science/article/pii/S0362546X05003378>.
- [27] Morris, M., Galland, J. C., and Balabanis, P. Concerted Action on Dam-Break Modelling. Proceedings of the CADAM Meeting at Wallingford, UK, 2 to 3 march 1999.
- [28] Mueller, L. O. and Toro, E. F. A global multiscale mathematical model for the human circulation with emphasis on the venous system. *International Journal for Numerical Methods in Biomedical Engineering*, 30(7):681–725, 2014. doi: 10.1002/cnm.2622. URL <http://dx.doi.org/10.1002/cnm.2622>.
- [29] Mueller, L. O. and Toro, E. F. Enhanced global mathematical model for studying cerebral venous blood flow. *Journal of Biomechanics*, 47(13):3361–3372, 2014. doi: 10.1016/j.jbiomech.2014.08.005. URL <http://dx.doi.org/10.1016/j.jbiomech.2014.08.005>.
- [30] Müller, L. O., Blanco, P. J., Watanabe, S. M., and Feijóo, R. A. A high-order local time stepping finite volume solver for one-dimensional blood flow simulations: application to the ADAN model. *International Journal for Numerical Methods in Biomedical Engineering*, 32(10):e02761, October 2016. ISSN 20407939. doi: 10.1002/cnm.2761. URL <http://doi.wiley.com/10.1002/cnm.2761>.
- [31] Olufsen, M. S., Peskin, C. S., Kim, W. Y., Pedersen, E. M., Nadim, A., and Larsen, J. Numerical simulation and experimental validation of blood flow in arteries with structured-tree outflow conditions. *Annals of Biomedical Engineering*, 28(11):1281–1299, nov 2000. doi: 10.1114/1.1326031. URL <http://dx.doi.org/10.1114/1.1326031>.
- [32] Quarteroni, A., Tuveri, M., and Veneziani, A. Computational vascular fluid dynamics: problems, models and methods. *Comput. Visual. Sci.*, 2:163–197, 2000.
- [33] Quarteroni, A. and Formaggia, L. Mathematical Modelling and Numerical Simulation of the Cardiovascular System. In *Modelling of Living Systems*, Handbook of Numerical Analysis Series. 2003.
- [34] Reigstad, G. A., Flåtten, T., Haugen, N. E., and Ytrehus, T. Coupling constants and the generalized Riemann problem for isothermal junction flow. *Journal of Hyperbolic Differential Equations*, 12(01):37–59, mar 2015. doi: 10.1142/s0219891615500022. URL <http://dx.doi.org/10.1142/s0219891615500022>.
- [35] Sherwin, S., Formaggia, L., Peiró, J., and Franke, V. Computational modelling of 1D blood flow with variable mechanical properties and its application to the simulation of wave propagation in the human arterial system. *International Journal for Numerical Methods in Fluids*, 43:673–700, 2003.
- [36] Sherwin, S., Franke, V., Peiró, J., and Parker, K. One-dimensional modelling of a vascular network in space-time variables. *Journal of Engineering Mathematics*, 47:217–250, 2003.
- [37] Toro, E. F., M.Spruce, and Speares, W. Restoration of the Contact Surface in the HLL-Riemann Solver. *Shock Waves*, 4: 25–34, 1994.
- [38] Toro, E. F., Millington, R. C., and Nejad, L. A. M. *Godunov Methods: Theory and Applications*, chapter Towards Very High Order Godunov Schemes., pages 905–937. Kluwer Academic/Plenum Publishers, 2001.
- [39] Toro, E. F. *Shock-capturing methods for free-surface shallow flows*. Wiley and Sons, 2001.
- [40] Toro, E. F. *Riemann Solvers and Numerical Methods for Fluid Dynamics*. Springer, third edition, 2009. doi: 10.1007/b79761. URL <http://dx.doi.org/10.1007/b79761>.
- [41] Toro, E. F. Brain Venous Haemodynamics, Neurological Diseases and Mathematical Modelling. A Review. *Applied Mathematics and Computation*. Vol. 272, pp 542-579. doi:10.1016/j.amc.2015.06.066, 2016.
- [42] Zhang, Y. Simulation of open channel network flows using finite element approach. *Communications in Nonlinear Science and Numerical Simulation*, 10:467–478, 2005.