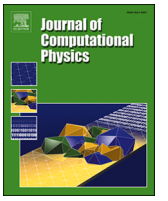
Journal of Computational Physics 561 (2026) 114975

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp



# Well-balanced high-order method for non-conservative hyperbolic PDEs with source terms: Application to one-dimensional blood flow equations with gravity

Chiara Colombo [a,b,*], Caterina Dalmaso [b], Lucas O. Müller [b], Annunziato Siviglia [a]

[a] Department of Civil, Environmental and Mechanical Engineering, University of Trento, Via Mesiano 77, Trento, 38123, Italy
[b] Laboratory of Mathematics for Biology and Medicine, Department of Mathematics, University of Trento, Via Sommarive 14, Trento, 38123, Italy

ARTICLE INFO

*Keywords:*
Well-balanced numerical method
High order
Path-conservative approach
Non-conservative hyperbolic PDEs
GRP-based reconstruction
Gravity
Blood flow

ABSTRACT

The present work proposes a well-balanced finite volume-type numerical method for the solution of non-conservative hyperbolic partial differential equations (PDEs) with source terms. The method is characterized, first, by the use of a recently introduced high-order spatial reconstruction, based on generalized Riemann problem information from the previous time level. Such reconstruction is well-balanced up to order three, compact, efficient and easy to implement. Second, the method incorporates a well-balanced space-time evolution operator, which allows for well-balanced fully explicit time evolution. The accuracy and efficiency of the method are assessed on both a scalar problem (Burgers' equation) and a nonlinear PDE system (hyperbolized one-dimensional blood flow equations with gravity and friction, and with variable mechanical and geometrical properties). The well-balanced property is verified by showing that numerically-determined stationary solutions are preserved up to machine precision. The order of accuracy in space and time is validated through empirical convergence rate studies. Additionally, the performance of the method is assessed on a network of 86 arteries, under both stationary and transient conditions.

## 1. Introduction

Several physical phenomena can be modeled through non-conservative hyperbolic PDEs with source terms. Notable examples include the shallow water equations (SWEs) with moving bottom topography [1], the multilayer SWEs with variable density [2], the Euler equations with gravity [3,4], and the characterization of blood flow within vessels having variable physical properties [5,6]. The solution of such equations in the presence of discontinuities (shock waves) poses several challenges, as the conservative form of the equations no longer holds. One of the main issues is the definition of jump relations across a discontinuity, which has been addressed by Dal Maso et al. [7] interpreting the jump relations as Borel measures dependent on the path connecting the two sides of the discontinuity.

A second issue is the description of steady-state solutions in the presence of geometric- and/or algebraic-type source terms, which can be discontinuous [8]. Such solutions should be accurately captured by the numerical methods used to solve the considered PDEs. Moreover, they should be recovered after small perturbations in the initial data, in order to avoid spurious oscillations that might cause

* Corresponding author.
*E-mail addresses:* chiara.colombo-1@unitn.it (C. Colombo), caterina.dalmaso@unitn.it (C. Dalmaso), lucas.muller@unitn.it (L.O. Müller), annunziato.siviglia@unitn.it (A. Siviglia).

https://doi.org/10.1016/j.jcp.2026.114975
Received 4 September 2025; Received in revised form 10 March 2026; Accepted 23 April 2026
Available online 25 April 2026

significant deviations from the exact solution [9,10]. A numerical scheme able to preserve the correct balance between advective and source terms was initially introduced in 1994 for the SWEs by Bermudez and Vazquez [11], paving the way for the numerous first- and high-order approaches developed in the last thirty years for different systems of balance laws [9,10,12–17], with a particular focus on variants of the SWEs [2,18–21] and Euler equations [3,4,22–25]. Nowadays, numerous methods are also available for the solution of blood flow equations (BFEs), tailored to the preservation of some or all stationary solutions of the system (well-balanced and fully well-balanced methods, respectively). Many of these methods were devised to treat space-varying geometrical and physical properties of the vasculature, friction and gravity [5,26–32].

Recently, Castro and Parés [17] and Guerrero Fernández et al. [10] proposed a generic strategy for the development of well-balanced high-order methods in the finite volume and discontinuous Galerkin frameworks, testing them on hyperbolic PDEs with continuous and discontinuous source terms. One of the main obstacles to the development of well-balanced methods is the construction of a well-balanced reconstruction operator [17], since standard reconstruction operators such as ENO, WENO and MUSCL [33–35] are not usually well-balanced. Indeed, these methods are based on standard interpolation techniques, and there is no guarantee that stationary solutions of the considered PDEs belong to the same class of reconstruction functions. A strategy to modify standard reconstruction operators so that they are well-balanced for all stationary solutions of the considered PDEs has been proposed by Castro et al. [36]. However, this approach requires the solution of a nonlinear system of equations, which might have multiple or no solutions, and might be computationally expensive [17].

In this paper, we present a numerical method that combines the novel centered reconstruction technique by Montecinos et al. [37] with the well-balancing approach by Castro and Parés [17] and Guerrero Fernández et al. [10]. The method is developed within the path-conservative framework, using the high-order solver of Müller et al. [38] and the well-balancing strategy of Guerrero Fernández et al. [10], which relies on the knowledge of families of stationary solutions. The reconstruction technique [37] requires, at each time step and in every computational cell, the cell average of the solution at the current time level and the solution of a generalized Riemann problem (GRP) at the cell interfaces from the previous time level. Provided that the GRP solver being used is well balanced, this minimal data dependence guarantees the well-balancing of the reconstruction technique by construction up to the third order, while also making the technique highly efficient and easy to implement.

In the following, we show that the resulting numerical method is well-balanced up to order three, testing it on both a scalar problem represented by the Burgers' equation, and a PDE system given by the hyperbolized one-dimensional BFEs presented in Montecinos et al. [39], including variable geometrical and mechanical vessel properties, friction and a spatially variable $C^0$ gravity term. The gravity term, in particular, is defined, depending on the considered test, either as a constant quantity, a smooth function or a polyline computed from anatomical data.

The rest of the work is structured as follows. Section 2 illustrates the proposed well-balanced numerical method, detailing the spatial reconstruction procedure [37] and our modifications to the local space-time prediction and time evolution steps described in Müller et al. [38]. Section 3 presents the numerical tests performed to assess the accuracy and efficiency of the solver. Numerical results are discussed in Section 4. Conclusions are drawn in Section 5.

## 2. Numerical method

We consider the following system

$$\begin{cases} \partial_t \mathbf{Q} + \mathbf{A}(\mathbf{Q})\partial_x \mathbf{Q} = \mathbf{S}(\mathbf{Q}), \quad x \in \Omega, \quad t \in \mathcal{T}, \\ \mathbf{Q}(x,0) = \mathbf{Q}_0(x), \end{cases} \tag{1}$$

where $t \in \mathcal{T}$ and $x \in \Omega$ are time and space independent variables, with $\Omega = [x_A, x_B] \subset \mathbb{R}$ a one-dimensional spatial domain and $\mathcal{T} = [0, t^K] \subset \mathbb{R}_0^+$ a temporal domain. $\mathbf{Q}(x,t) \in \mathcal{B}_Q$ is the state vector, with $\mathcal{B}_Q \subset \mathbb{R}^v$ the space of the admissible states, $\mathbf{Q}_0(x) \in \mathcal{B}_Q$ is the initial condition, $\mathbf{S}(\mathbf{Q}(x,t)) \in \mathbb{R}^v$ is the source vector, and $\mathbf{A}(\mathbf{Q}(x,t)) \in \mathbb{R}^{v \times v}$ is the system matrix with real and distinct eigenvalues. If $\mathbf{A}(\mathbf{Q})$ is the Jacobian of the system, then the previous equation reduces to a classical balance law.

Our goal is to construct a numerical scheme that preserves stationary solutions in the framework of path-conservative numerical methods. We begin by identifying the stationary solutions of problem (1) as described below. Let $\mathbf{Q}^*(x) \in \mathcal{B}_Q, \forall x \in \Omega$, be the stationary solution of problem (1), then it holds

$$\mathbf{A}(\mathbf{Q}^*)\partial_x \mathbf{Q}^* = \mathbf{S}(\mathbf{Q}^*). \tag{2}$$

Subtracting Eq. (2) from Eq. (1) [10], we obtain an equivalent problem given by

$$\begin{cases} \partial_t \mathbf{Q} + \mathbf{A}(\mathbf{Q})\partial_x \mathbf{Q} - \mathbf{A}(\mathbf{Q}^*)\partial_x \mathbf{Q}^* = \mathbf{S}(\mathbf{Q}) - \mathbf{S}(\mathbf{Q}^*), \quad x \in \Omega, \quad t \in \mathcal{T}, \\ \mathbf{Q}(x,0) = \mathbf{Q}_0(x). \end{cases} \tag{3}$$

Then, we discretize the spatial domain $\Omega$ into $N$ computational cells $S_i = [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$, with $i = 1, \dots, N$, and the temporal domain $\mathcal{T}$ into $K$ computational cells $T^n = [t^n, t^{n+1}]$ with $n = 0, \dots, K-1$. Integrating Eq. (3) in space and time over the control volume $V_i^n = S_i \times T^n$, we obtain the following numerical scheme

$$\mathbf{Q}_i^{n+1} = \mathbf{Q}_i^n - \frac{1}{\Delta x}\left(\mathbf{B}_i - \mathbf{B}_i^*\right) - \frac{\Delta t^n}{\Delta x}\left(\mathbf{D}^-_{i+\frac{1}{2}} + \mathbf{D}^+_{i-\frac{1}{2}}\right) + \Delta t^n\left(\mathbf{S}_i - \mathbf{S}_i^*\right), \tag{4}$$

where

$$\mathbf{Q}_i^n \approx \frac{1}{\Delta x}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \mathbf{Q}(x,t^n)dx, \tag{5}$$

$$\mathbf{B}_i \approx \int_{t^n}^{t^{n+1}}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \mathbf{A}(\mathbf{Q})\partial_x\mathbf{Q}dxdt, \quad \mathbf{B}_i^* \approx \int_{t^n}^{t^{n+1}}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \mathbf{A}(\mathbf{Q}^*)\partial_x\mathbf{Q}^*dxdt, \tag{6}$$

$$\mathbf{S}_i \approx \frac{1}{\Delta x\Delta t^n}\int_{t^n}^{t^{n+1}}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \mathbf{S}(\mathbf{Q})dxdt, \quad \mathbf{S}_i^* \approx \frac{1}{\Delta x\Delta t^n}\int_{t^n}^{t^{n+1}}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \mathbf{S}(\mathbf{Q}^*)dxdt, \tag{7}$$

and

$$\mathbf{D}^{\pm}_{i+\frac{1}{2}} \approx \frac{1}{\Delta t^n}\int_{t^n}^{t^{n+1}} D^{\pm}_{i+\frac{1}{2}}\left(\mathbf{Q}^-_{i+\frac{1}{2}}(t),\mathbf{Q}^+_{i+\frac{1}{2}}(t),\mathbf{\Psi}\left(\mathbf{Q}^-_{i+\frac{1}{2}}(t),\mathbf{Q}^+_{i+\frac{1}{2}}(t),s\right)\right)dt. \tag{8}$$

$\Delta x = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}$ is the mesh spacing, and $\Delta t^n = t^{n+1} - t^n$ is the $n$-th time step. $\mathbf{Q}_i^n$ denotes the $i$-th cell average at time $t^n$, $\mathbf{B}_i$ and $\mathbf{B}_i^*$ are the non-conservative products of the $i$-th cell, $\mathbf{S}_i$ and $\mathbf{S}_i^*$ are the numerical source terms of the $i$-th cell, while $\mathbf{D}^{\pm}_{i+\frac{1}{2}}$ are the numerical fluctuations across the interface $x_{i+\frac{1}{2}}$. $D^{\pm}_{i+\frac{1}{2}}\left(\mathbf{Q}^-_{i+\frac{1}{2}},\mathbf{Q}^+_{i+\frac{1}{2}},\mathbf{\Psi}\right)$ are the jump terms on the cells boundaries, also called fluctuations, that arise from the solution of a Riemann problem (RP) at the cell interface $x_{i+\frac{1}{2}}$. They depend on the left and right states at the cell interface $\mathbf{Q}^{\pm}_{i+\frac{1}{2}}(t)$, and on the integration path $\mathbf{\Psi}\left(\mathbf{Q}^-_{i+\frac{1}{2}},\mathbf{Q}^+_{i+\frac{1}{2}},s\right)$. Here, we compute them as described in [5]. The integration path $\mathbf{\Psi}$ is a parametric arc in the parameter $s \in [0,1]$ that is used to connect the left and right states $\mathbf{Q}^{\pm}_{i+\frac{1}{2}}(t)$. Multiple path choices are possible. Theoretical details on the choice of paths are given in Parés [40]. Here, we adopt the segment path for the Burgers' equation, and a modification of it for the BFEs [5], which is later defined in Section 3.2. The left and right states $\mathbf{Q}^{\pm}_{i+\frac{1}{2}}(t)$ are space-time reconstructed data that are extrapolated to both sides of the cell interface $x_{i+\frac{1}{2}}$. These space-time reconstructed data are space-time predictions of the sought solution $\mathbf{Q}(x,t)$, computed with a local implicit discontinuous Galerkin scheme. Here, we adopt a modified version of the Dumbser-Enoux-Toro GRP solver [41] that uses spatial reconstruction polynomials as initial data. These polynomials are computed through a reconstruction procedure that exploits GRP-based predictions from the previous time step to construct the reconstructions at the current step. Together, the GRP solver and the spatial reconstruction technique, described in detail below, provide the quantities required for evaluating the integrals in (4), thereby yielding a high-order numerical scheme in both space and time.

In order to have a fully explicit evolution of the data $\mathbf{Q}_i^{n+1}$ at each time step, we have to approximate integrals appearing in Eq. (4) with the desired order of accuracy. The numerical fluctuations $\mathbf{D}^{\pm}_{i+\frac{1}{2}}$ (8) are computed as follows. For each quadrature node $t^n + \tau_v\Delta t^n$, with $\tau_v \in [0,1]$, employed in the time integration, we solve a classical RP for system (3) with initial condition given by the boundary states $\mathbf{Q}^{\pm}_{i+\frac{1}{2}}(t)$ evaluated at the $i+\frac{1}{2}$ interface and at each time quadrature node $t^n + \tau_v\Delta t^n$, (see Section 2.3 for the computation of $\mathbf{Q}^{\pm}_{i+\frac{1}{2}}(t)$), namely

$$\mathbf{Q}(x,t^n+\tau_v\Delta t^n) = \begin{cases} \mathbf{Q}^-_{i+\frac{1}{2}}(t^n+\tau_v\Delta t^n) & \text{if } x < x_{i+\frac{1}{2}}, \\ \mathbf{Q}^+_{i+\frac{1}{2}}(t^n+\tau_v\Delta t^n) & \text{if } x > x_{i+\frac{1}{2}}. \end{cases} \tag{9}$$

In particular, the classical RP (3)–(9) is solved approximately by computing jump terms $D^{\pm}_{i+\frac{1}{2}}$ as described in Müller and Toro [5], namely

$$D^{\pm}_{i+\frac{1}{2}} \approx \frac{1}{2}\int_0^1 \left[\mathbf{A}\left(\mathbf{\Psi}(\mathbf{Q}^-_{i+\frac{1}{2}},\mathbf{Q}^+_{i+\frac{1}{2}},s)\right) \pm \left|\mathbf{A}\left(\mathbf{\Psi}(\mathbf{Q}^-_{i+\frac{1}{2}},\mathbf{Q}^+_{i+\frac{1}{2}},s)\right)\right|\right]\frac{\partial\mathbf{\Psi}}{\partial s}ds, \tag{10}$$

where $\Psi(\mathbf{Q}^-_{i+\frac{1}{2}},\mathbf{Q}^+_{i+\frac{1}{2}},s)$ is the integration path, later defined in Section 3.2, and where the absolute value operator of a matrix is defined as

$$|\mathbf{A}| = \mathbf{R}|\mathbf{\Lambda}|\mathbf{R}^{-1}, \quad |\mathbf{\Lambda}| = diag(|\lambda_1|,\dots,|\lambda_v|), \tag{11}$$

with $\lambda_i, i = 1,\dots,v$ eigenvalues of matrix $\mathbf{A}$, $\mathbf{R}$ the matrix of right eigenvectors of $\mathbf{A}$ and $\mathbf{R}^{-1}$ its inverse. The integral in phase space in (10) and the time integral in (8) are approximated using second- or third-order quadrature rules with Gaussian quadrature nodes and Lagrange interpolation polynomials defined on these nodes. In analogous manner, the numerical sources $\mathbf{S}_i$ and $\mathbf{S}_i^*$ in (7), and the two non-conservative products $\mathbf{B}_i$ and $\mathbf{B}_i^*$ in (6) are approximated with the same quadrature rule in both space and time.

### 2.1. Summary of the method

Consider a non-conservative hyperbolic PDE system with source terms written as in (1). The following list summarizes how to compute its numerical solution with the proposed methodology:

1) Discretize the spatial domain $\Omega$ into $N$ computational cells $S_i$, and the temporal domain $\mathcal{T}$ into $K$ temporal cells $T^n$.
2) At each time step $t^n$ and for each cell $S_i$, compute the steady-state solution of problem (1) using Eq. (12) and Eq. (13) described in Section 2.2.
3) At each time step $t^n$ and for each cell $S_i$, compute the spatial reconstruction polynomials using Eq. (27) or Eq. (28) described in Section 2.4.
4) At each time step $t^n$ and for each cell $S_i$, compute the space-time predictions using information from points 2) and 3), and the GRP solver presented in Section 2.3.
5) At each time step $t^n$ and for each cell $S_i$, use information from points 2) and 4) to approximate integrals (5)–(8).
6) At each time step $t^n$ and for each cell $S_i$, update the solution using Eq. (4).
7) Repeat points 2)-6) till the final simulation time is reached.

### 2.2. Stationary solution identification

In this section, we describe the procedure that, at each time step $t^n$, and for each computational cell $S_i$, we use to identify a suitable stationary solution $\mathbf{Q}^*(x)$ of problem (1) defined in each of the $P(=2,3)$ quadrature points $x_p$, $p=0,\dots,P-1$ used in the numerical scheme. To this end, we discretize each computational cell $S_i$ into $P-1$ intervals $[x_{p-1}, x_p]$ of length $h$. Then, we identify a succession of $P$ values $\{\mathbf{Q}^*_{i,p}\}_{p=0}^{P-1}$ representing the approximated stationary solution in each node $x_p$. Specifically, this succession is obtained by applying the Runge-Kutta method (RK) of order $P$ and with $P$ stages to Eq. (2). In this work, we consider the second-order RK with 2 stages, and the third-order RK with 3 stages. In both cases, the standard Butcher tableau with coefficients $a_{jk}, b_j$, and $c_j$ is employed. Therefore, we compute $\mathbf{Q}^*(x_p) \approx \mathbf{Q}^*_{i,p}$ for $p>0$ as

$$\begin{cases} \mathbf{Q}^*_{i,p+1} = \mathbf{Q}^*_{i,p} + h\sum_{j=0}^{P-1} b_j K_j, \\ K_j = \tilde{f}(x_p + c_j h, \mathbf{Q}^*_{i,p} + h\sum_{k=0}^{P-1} a_{jk} K_k). \end{cases} \tag{12}$$

Additionally, in order to identify $\mathbf{Q}^*_{i,0}$, we require that the average stationary solution in $S_i$ coincides with the cell average $\mathbf{Q}^n_i$ [10]. Particularly, we apply a Gaussian quadrature rule of order $P$, with nodes $x_p$ and weights $w_p$, to the average stationary solution integral and we enforce the equivalence at the discrete level. As a result, we compute $\mathbf{Q}^*_{i,0}$ by solving the following non-linear system of equations

$$w_0 \mathbf{Q}^*_{i,0} + \sum_{p=1}^{P-1} w_p \mathbf{Q}^*_{i,p}(\mathbf{Q}^*_{i,0}) = \mathbf{Q}^n_i. \tag{13}$$

Here, the solution to this system was found by applying the standard Newton method.

### 2.3. The well-balanced DET solver

In this section, we present the procedure for constructing a space-time polynomial $\mathbf{Q}^{ST,n}_i(x,t)$ that approximates the solution $\mathbf{Q}(x,t)$ within $V^n_i$ and is employed in the evaluation of the integrals in Eq. (4). Particularly, we apply a modified version of the Dumbser-Enaux-Toro (DET) GRP solver [41] that is based on the calculation of the so-called deviations. We define deviation $\mathbf{d}^n_i(x,t)$ as the difference between the space-time polynomial $\mathbf{Q}^{ST,n}_i(x,t)$ and the stationary solution $\mathbf{Q}^*_i(x)$, namely

$$\mathbf{d}^n_i(x,t) = \mathbf{Q}^{ST,n}_i(x,t) - \mathbf{Q}^*_i(x), \quad \forall (x,t) \in V^n_i. \tag{14}$$

In order to compute integrands in Eq. (4) at the desired time levels, we need to know the solution of the GRP for system in (3) with initial condition defined as

$$\mathbf{Q}(x,t^n) = \begin{cases} \mathbf{w}^n_{i-1}(x), & \text{if } x < x_{i-\frac{1}{2}}, \\ \mathbf{w}^n_i(x), & \text{if } x > x_{i-\frac{1}{2}}, \end{cases} \tag{15}$$

where $\mathbf{w}^n_i(x)$ is the reconstruction polynomial of order $P-1$ in $S_i$ at time $t^n$. The computation of $\mathbf{w}^n_i(x)$ is described in Section 2.4. The DET solver [41] finds the solution to this GRP at the desired time points by first performing a space-time evolution procedure, and then solving classical RPs at those time points using the space-time predictions as initial conditions. In particular, locally evolved data in space and time, the space-time predictions $\mathbf{Q}^{ST,n}_i$, are computed through a local space-time discontinuous Galerkin finite element scheme of order $P$. Given the structure of the adopted numerical scheme, these predictions $\mathbf{Q}^{ST,n}_i$ are space-time polynomials $\mathbf{Q}^{ST,n}_i(x,t)$ evaluated at spatial and temporal quadrature nodes, i.e. $x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}$ and $t^n, t^{n+1}$ for a second-order method, and $x_{i-\frac{1}{2}}, x_i, x_{i+\frac{1}{2}}$ and $t^n, t^{n+\frac{1}{2}}, t^{n+1}$ for a third-order method. The classical Riemann solver instead provides the solution of the GRP at the time points along interfaces where such solution is required for the computation of the numerical fluctuations $\mathbf{D}^{\pm}_{i\pm\frac{1}{2}}$ (8) and the reconstruction

polynomials, by solving classical RPs and using space-time predictions at the previous time level on the left and on the right of the same interface as initial data. Before proceeding with the description of the DET solver, we note that the calculation of $\mathbf{Q}_i^{ST,n}(x,t)$ will constitute the data needed for the computation of jump terms (10), in fact, we have that:

$$\mathbf{Q}^-_{i+\frac{1}{2}}(t) = \mathbf{Q}_i^{ST,n}(x_{i+\frac{1}{2}},t), \quad \mathbf{Q}^+_{i+\frac{1}{2}}(t) = \mathbf{Q}_{i+1}^{ST,n}(x_{i+\frac{1}{2}},t). \tag{16}$$

Additionally, we underline that whenever we refer to a local quantity, we drop both the temporal superscript $n$ and the spatial subscript $i$, keeping only the subscript $h$ to recall that we are in the local framework, for example $\mathbf{Q}_h^{ST} = \mathbf{Q}_i^{ST,n}$.

The local framework is defined by the transformation of the computational volume $V_i^n$ into the reference space-time element $V_h = [0,1]\times[0,1]$ with the reference coordinate $\xi$ and $\tau$ as $x = x_{i-\frac{1}{2}} + \Delta x\xi$, and $t = t^n + \Delta t^n\tau$. Hence, problem (3) becomes

$$\begin{cases} \partial_\tau \mathbf{Q}_h^{ST} + \frac{\Delta t^n}{\Delta x}\mathbf{A}(\mathbf{Q}_h^{ST})\partial_\xi \mathbf{Q}_h^{ST} - \frac{\Delta t^n}{\Delta x}\mathbf{A}(\mathbf{Q}_h^{*})\partial_\xi \mathbf{Q}_h^{*} = \Delta t^n \mathbf{S}(\mathbf{Q}_h^{ST}) - \Delta t^n \mathbf{S}(\mathbf{Q}_h^{*}), \\ \mathbf{Q}_h^{ST}(\xi,0) = \mathbf{w}_h(\xi). \end{cases} \tag{17}$$

We note that this change of coordinate does not affect the mesh spacing $\Delta x$ or the time step $\Delta t^n$, as both are constant quantities. Since $\partial_\tau \mathbf{Q}_h^* = 0$, using relation (14), we replace the term $\partial_\tau \mathbf{Q}_h^{ST}$ with $\partial_\tau \mathbf{d}_h$. Additionally, we write the space-time predictor in terms of deviations as $\mathbf{Q}_h^{ST}(\xi,\tau) = \mathbf{d}_h(\xi,\tau) + \mathbf{Q}_h^*(\xi)$. As a result, we transform the unknown of our problem from $\mathbf{Q}_h^{ST}(\xi,\tau)$ to $\mathbf{d}_h(\xi,\tau)$. We also note that if both the non-conservative product $\mathbf{A}(\mathbf{Q})\partial_\xi\mathbf{Q}$ and the source term $\mathbf{S}(\mathbf{Q})$ are linear operators, then Eq. (17) can be written only in terms of deviations $\mathbf{d}_h(\xi,\tau)$, without the stationary solution $\mathbf{Q}_h^*(\xi)$ explicitly appearing in the formulation.

Then, we multiply Eq. (17) by a space-time basis function $\theta(\xi,\tau) \in \mathbb{P}_{m,m}$ with $m$ the degree of the polynomial, and we integrate it over $V_h$. After applying integration by parts only in time, we obtain

$$[\theta,\mathbf{d}_h]^1 - \langle \partial_\tau\theta, \mathbf{d}_h\rangle_{V_h} - [\theta,\mathbf{d}_h]^0 + \frac{\Delta t^n}{\Delta x}\langle \theta, \mathbf{A}(\mathbf{Q}_h^{ST})\partial_\xi\mathbf{Q}_h^{ST}\rangle_{V_h} - \frac{\Delta t^n}{\Delta x}\langle \theta, \mathbf{A}(\mathbf{Q}_h^{*})\partial_\xi\mathbf{Q}_h^{*}\rangle_{V_h} = \Delta t^n\langle\theta, \mathbf{S}(\mathbf{Q}_h^{ST})\rangle_{Vh} - \Delta t^n\langle\theta,\mathbf{S}(\mathbf{Q}_h^*)\rangle_{Vh}, \tag{18}$$

where we have used the following notation for the two scalar products of two functions $f(\xi,\tau)$ and $g(\xi,\tau)$

$$[f,g]^\tau = \int_0^1 f(\xi,\tau)g(\xi,\tau)d\xi, \qquad \langle f,g\rangle_{V_h} = \int_0^1\int_0^1 f(\xi,\tau)g(\xi,\tau)d\xi d\tau. \tag{19}$$

We now take the element $[\theta,\mathbf{d}_h]^0$ of Eq. (18) at reference time $\tau = 0$ and we observe that it is completely defined by the initial condition. The initial condition [41] is

$$\mathbf{d}_{h,0}(\xi) = \mathbf{d}_h(\xi,0) = \mathbf{w}_h(\xi) - \mathbf{Q}_h^*(\xi), \tag{20}$$

where the stationary term $\mathbf{Q}_h^*(\xi)$ is known from previous computations.

Later, using the same basis function $\theta(\xi,\tau)$, we approximate the different quantities in (18) expanding them as

$$\mathbf{d}_h(\xi,\tau) = \sum_{l=1}^{P^2}\theta_l\widehat{\mathbf{d}}_l, \qquad \mathbf{Q}_h^*(\xi) = \sum_{l=1}^{P^2}\theta_l\widehat{\mathbf{Q}^*}_l, \tag{21}$$

$$\mathbf{A}(\mathbf{Q}_h^{ST})\partial_\xi\mathbf{Q}_h^{ST}(\xi,\tau) = \sum_{l=1}^{P^2}\theta_l\widehat{\mathbf{A}\partial_\xi\mathbf{Q}}_l, \qquad \mathbf{A}(\mathbf{Q}_h^{*})\partial_\xi\mathbf{Q}_h^{*}(\xi) = \sum_{l=1}^{P^2}\theta_l\widehat{\mathbf{A}^*\partial_\xi\mathbf{Q}^*}_l, \tag{22}$$

$$\mathbf{S}(\mathbf{Q}_h^{ST})(\xi,\tau) = \sum_{l=1}^{P^2}\theta_l\widehat{\mathbf{S}}_l = \sum_{l=1}^{P^2}\theta_l\mathbf{S}(\widehat{\mathbf{Q}}_l) = \sum_{l=1}^{P^2}\theta_l\mathbf{S}(\widehat{\mathbf{d}}_l + \widehat{\mathbf{Q}^*}_l), \tag{23}$$

$$\mathbf{S}(\mathbf{Q}_h^{*})(\xi) = \sum_{l=1}^{P^2}\theta_l\widehat{\mathbf{S}^*}_l = \sum_{l=1}^{P^2}\theta_l\mathbf{S}(\widehat{\mathbf{Q}^*}_l). \tag{24}$$

The term $\widehat{\mathbf{A}\partial_\xi\mathbf{Q}}_l$ is obtained by assuming that $\widehat{\mathbf{A}\partial_\xi\mathbf{Q}}_l = \mathbf{A}(\widehat{\mathbf{Q}}_l)\widehat{\partial_\xi\mathbf{Q}}_l$, and by computing the expansion coefficients of the spatial derivative as $\langle\theta_k,\theta_l\rangle_{V_h}\widehat{\partial_\xi\mathbf{Q}}_l = \langle\theta_k,\partial_\xi\theta_l\rangle_{V_h}\widehat{\mathbf{Q}}_l$ for $k = 1,\dots,P^2$ [38]. Similarly, $\widehat{\mathbf{A}^*\partial_\xi\mathbf{Q}^*}_l = \mathbf{A}(\widehat{\mathbf{Q}^*}_l)\widehat{\partial_\xi\mathbf{Q}^*}_l$, and the spatial derivative is computed as $\langle\theta_k,\theta_l\rangle_{V_h}\widehat{\partial_\xi\mathbf{Q}^*}_l = \langle\theta_k,\partial_\xi\theta_l\rangle_{V_h}\widehat{\mathbf{Q}^*}_l$ for $k = 1,\dots,P^2$. In general, $\forall l = 1,\dots,P^2$, the expansion coefficients $\widehat{\mathbf{Q}^*}_l$, $\widehat{\mathbf{A}^*\partial_\xi\mathbf{Q}^*}_l$, and $\widehat{\mathbf{S}^*}_l$, are known because of the knowledge of the stationary solution. The expansion coefficients $\widehat{\mathbf{Q}}_l$ instead can be written in terms of deviations as $\widehat{\mathbf{Q}}_l = \widehat{\mathbf{d}}_l + \widehat{\mathbf{Q}^*}_l$. Finally, the expansions coefficients related to the deviations $\widehat{\mathbf{d}}_l$, $\forall l = 1,\dots,P^2$, which are the resulting unknown of system (18), are found applying a fixed-point iteration procedure to the following system

$$\begin{aligned} &\big\{[\theta_k,\theta_l]^1 - \langle\partial_\tau\theta_k,\theta_l\rangle_{V_h}\big\}\widehat{\mathbf{d}}_l^{m+1} - \Delta t^n\langle\theta_k,\theta_l\rangle_{Vh}\mathbf{S}(\widehat{\mathbf{d}}_l^{m+1} + \widehat{\mathbf{Q}^*}_l) = \\ &[\theta_k,\psi_l]^0\widehat{\mathbf{d}}_{0l} - \frac{\Delta t^n}{\Delta x}\langle\theta_k,\theta_l\rangle_{V_h}\mathbf{A}(\widehat{\mathbf{d}}_l^m + \widehat{\mathbf{Q}^*}_l)\big\{\langle\theta_k,\theta_l\rangle_{V_h}\big\}^{-1}\langle\theta_k,\partial_\xi\theta_l\rangle_{V_h}(\widehat{\mathbf{d}}_l^m + \widehat{\mathbf{Q}^*}_l) \\ &+\frac{\Delta t^n}{\Delta x}\langle\theta_k,\theta_l\rangle_{V_h}\mathbf{A}(\widehat{\mathbf{Q}^*}_l)\big\{\langle\theta_k,\theta_l\rangle_{V_h}\big\}^{-1}\langle\theta_k,\partial_\xi\theta_l\rangle_{V_h}\widehat{\mathbf{Q}^*}_l - \Delta t^n\langle\theta_k,\theta_l\rangle_{Vh}\mathbf{S}(\widehat{\mathbf{Q}^*}_l), \end{aligned} \tag{25}$$

for $k = 1,\dots,P^2$, where we have expanded the initial condition as $\mathbf{d}_{h,0}(\xi) = \sum_{l=1}^{P^2}\psi_l\widehat{\mathbf{d}}_{0l} = \sum_{l=1}^{P^2}\psi_l(\widehat{\mathbf{w}}_l - \widehat{\mathbf{Q}^*}_l)$, and where $m$ represents the current iteration step. We underline that due to the non-linearity of the system we decided to evaluate the second term of the

right-hand side at the known expansion coefficients $\widehat{\mathbf{d}}_l^m$ of the current iteration. Once the expansion coefficients are known, we finally retrieve the space-time predictions $\mathbf{Q}_i^{ST,n}$ in $V_i^n$. We remark that the modified version of the DET solver here described is equivalent to the original version of the DET solver if we assume that the stationary solution $\mathbf{Q}_i^*(x)$ is always zero.

### 2.4. Spatial reconstruction

In this section, we provide a brief overview of the GRP-based reconstruction procedure used to derive second- and third-order accurate spatial reconstruction polynomials $\mathbf{w}_i^n(x)$ [37]. These polynomials serve as local reconstructed data that provide the initial condition for the GRP solver, which in turns computes high-order space-time predictions of the solution $\mathbf{Q}(x,t)$. As a result, they enable the construction of a high-order scheme in both space and time.

They are piecewise functions of degree $P-1$, with $P$ being the order of the numerical method, defined on each computational cell $S_i$ at time $t^n$. They are expressed in terms of the cell average of the solution at the current time step $\mathbf{Q}_i^n$, and in terms of left and right boundary states at the previous time step $\mathbf{Q}_{i-\frac{1}{2}}^{+,n-1}$ and $\mathbf{Q}_{i+\frac{1}{2}}^{-,n-1}$. These states arise from the solution of a classical RP at time $t^{n-1}+\Delta t^{n-1}$ at the interfaces $x_{i\pm\frac{1}{2}}$ between two neighboring cells, and are also used to compute the integrand in the time integral of the fluctuations (8). For the first time step, these states are obtained directly from the prescribed initial condition, evaluated at time $t=0$ and at the cell interfaces $x_{i\pm\frac{1}{2}}$. For the left interface $x_{i-\frac{1}{2}}$, the classical RP reads

$$\begin{cases} \partial_t \mathbf{Q} + \mathbf{A}(\mathbf{Q})\partial_x \mathbf{Q} = 0, \quad x \in \mathbb{R}, t > t^{n-1}, \\ \mathbf{Q}(x, t^{n-1}+\Delta t^{n-1}) = \begin{cases} \mathbf{Q}_{i-1}^{ST,n-1}(x_{i-\frac{1}{2}}, t^{n-1}+\Delta t^{n-1}) & \text{if } x < x_{i-\frac{1}{2}}, \\ \mathbf{Q}_{i}^{ST,n-1}(x_{i-\frac{1}{2}}, t^{n-1}+\Delta t^{n-1}) & \text{if } x > x_{i-\frac{1}{2}}, \end{cases} \end{cases} \tag{26}$$

where $\mathbf{Q}_{i-1}^{ST,n-1}$ and $\mathbf{Q}_{i}^{ST,n-1}$ are space-time predictions at time $t^{n-1}+\Delta t^{n-1}$ and related to cells $S_{i-1}$ and $S_i$, respectively. Their computation is described in the previous section. The solution to RP (26) can be obtained using various methods [42]. Here we applied the exact Riemann solver for the Burgers' equation, and a two-rarefaction approximate Riemann solver for the BFEs, which yielded to a non-linear system in the left boundary states unknowns $\mathbf{Q}_{i-\frac{1}{2}}^{\pm,n-1}$. The $-$ and $+$ superscripts on the left boundary states denote, respectively, values taken from the left and right of the interface. This distinction is essential, since at cell interfaces the solution of the considered RP can exhibit discontinuities. Details on how the non-linear system was solved are provided in Section 1 of the supplementary material. The solution to this problem internal to cell $S_i$, namely $\mathbf{Q}_{i-\frac{1}{2}}^{+,n-1}$, is then used to find the reconstruction polynomial $\mathbf{w}_i^n(x)$. We emphasize that the use of boundary states computed at time $t^{n-1}+\Delta t^{n-1}$ is essential for stability. If instead interface information evaluated solely at the current time level $t^n$ were employed, the resulting numerical scheme would lead to an unstable method.

Due to the structure of our numerical method, the reconstruction polynomials $\mathbf{w}_i^n(x)$ need to be evaluated only in the quadrature nodes, i.e $x_{i-\frac{1}{2}}$ and $x_{i+\frac{1}{2}}$ for a second-order method, and $x_{i-\frac{1}{2}}, x_i$ and $x_{i+\frac{1}{2}}$ for a third-order method. In particular, for a second-order method, a first degree polynomial evaluated in the 2 quadrature nodes $x_{i\pm\frac{1}{2}}$ of the computational cell $S_i$ at time $t^n$ reads

$$\mathbf{w}_i^n(x_{i\pm\frac{1}{2}}) = \mathbf{Q}_i^n \pm \frac{1}{2}(\mathbf{Q}_{i+\frac{1}{2}}^{-,n-1} - \mathbf{Q}_{i-\frac{1}{2}}^{+,n-1}). \tag{27}$$

Similarly, for a third-order method, a second degree polynomial evaluated in the 3 quadrature nodes $x_p \in \{x_{i\pm\frac{1}{2}}, x_i\}$ of the computational cell $S_i$ at time $t^n$ reads

$$\mathbf{w}_i^n(x_p) = a + bx_p + cx_p^2, \tag{28}$$

where

$$a = \mathbf{Q}_{i-\frac{1}{2}}^{+,n-1}, \tag{29}$$

$$b = \frac{2}{\Delta x}(-2\mathbf{Q}_{i-\frac{1}{2}}^{+,n-1} - \mathbf{Q}_{i+\frac{1}{2}}^{-,n-1} + 3\mathbf{Q}_i^n), \tag{30}$$

and

$$c = \frac{3}{(\Delta x)^2}(\mathbf{Q}_{i-\frac{1}{2}}^{+,n-1} + \mathbf{Q}_{i+\frac{1}{2}}^{-,n-1} - 2\mathbf{Q}_i^n). \tag{31}$$

Both reconstruction polynomials are fully determined by either computing a slope for a second-order method, or interpolating them through the boundary states $\mathbf{Q}_{i\pm\frac{1}{2}}^{\mp,n-1}$ for a third-order method associated with cell $S_i$, and by enforcing the conservation property, which ensures that the cell average of cell $S_i$ at time $t^n$ is $\mathbf{Q}_i^n$. Consequently, if the cell average is that of the steady-state solution on the same computational cell, and the boundary states lie on the same steady-state solution, then the resulting reconstruction polynomials evaluated in the quadrature nodes $x_p$ coincide with the steady-state solution. We underline that this feature of the GRP-based reconstruction procedure applies only to first and second degree polynomials. Higher degree polynomials require additional

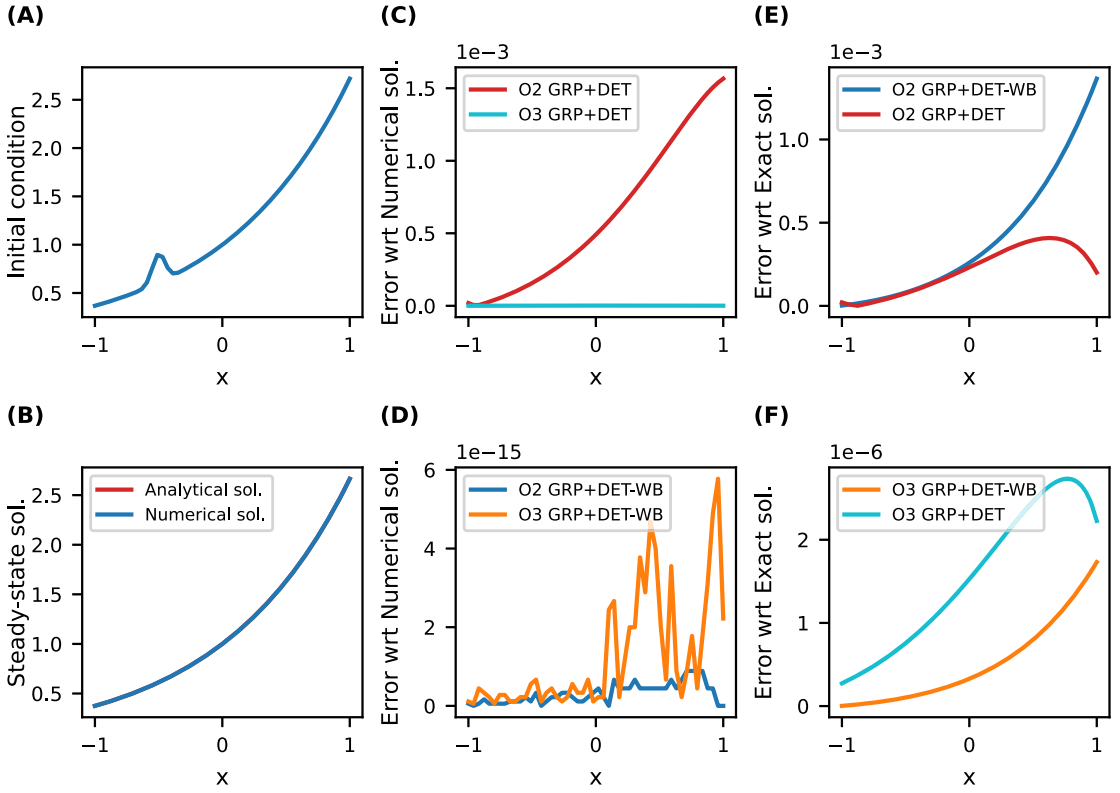



**Fig. 1.** Burgers' problem. Initial condition (A), steady-state solution (B), and errors in space between the numerical solution and the steady-state solution obtained either numerically (C and D) or analytically (E and F). Results are shown for both second- and third-order implementations of the GRP + DET and the GRP + DET-WB methods.

information coming from neighboring cells $S_{i\pm1}$ for their complete identification [37]. Thus, they may fail to coincide with the steady-state solution.

## 3. Numerical tests

This section presents the numerical tests conducted to evaluate the performance of the proposed high-order numerical method. Two representative test cases are considered: the scalar Burgers' equation and the hyperbolized BFEs system introduced in Müller et al. [38]. To carry out these tests, we employ both second- and third-order implementations of four different methods for comparison. The first is our method, which combines the GRP-based reconstruction with the well-balanced DET solver (GRP + DET-WB). The second method employs the GRP-based reconstruction, coupled with the original, non-well-balanced DET solver (GRP + DET). The third method combines the WENO reconstruction with the well-balanced DET solver (WENO + DET-WB). The fourth method is a reference approach based on the WENO reconstruction combined with the original DET solver (WENO + DET). More details about the WENO reconstruction can be found in Liu et al. [34].

### 3.1. Scalar case: Burgers' equation

Let us replicate the test proposed by Guerrero Fernández et al. [10], where we consider the following one-dimensional scalar Burgers' equation with algebraic nonlinear source term, expressed in quasi-linear form (1) as

$$\partial_t q + q\partial_x q = q^2, \quad x \in \Omega, \quad t \in \mathcal{T}, \tag{32}$$

with initial condition (see Fig. 1A) defined as

$$q(x,0) = \exp(x) + 0.3\exp\left(-200(x+0.5)^2\right), \quad x \in \Omega, \tag{33}$$

and left boundary condition as

$$q(x_A,t) = \exp(x_A), \quad t \in \mathcal{T}. \tag{34}$$

**Table 1**
$L^1$ and $L^\infty$ error norms, and corresponding empirical convergence rates for a second- and third-order implementation of the numerical scheme. $N$ is the number of computational cells.

| N | $L^1$ | $L^\infty$ | $O(L^1)$ | $O(L^\infty)$ | $L^1$ | $L^\infty$ | $O(L^1)$ | $O(L^\infty)$ |
|---|---|---|---|---|---|---|---|---|
| | Order 2 | | | | Order 3 | | | |
| 32 | 1.9e-03 | 3.2e-03 | – | – | 3.8e-06 | 6.4e-06 | – | – |
| 64 | 4.9e-04 | 8.4e-04 | 2.0 | 1.9 | 4.8e-07 | 8.3e-07 | 3.0 | 2.9 |
| 128 | 1.2e-04 | 2.2e-04 | 2.0 | 2.0 | 6.1e-08 | 1.1e-07 | 3.0 | 3.0 |
| 256 | 3.1e-05 | 5.5e-05 | 2.0 | 2.0 | 7.6e-09 | 1.3e-08 | 3.0 | 3.0 |
| 512 | 7.8e-06 | 1.4e-05 | 2.0 | 2.0 | 9.6e-10 | 1.7e-09 | 3.0 | 3.0 |

A transparent right boundary condition is also enforced. Stationary solutions of Eq. (32) can be computed either analytically as

$$q(x) = \exp\left(x\right), \quad x \in \Omega, \tag{35}$$

or numerically by applying the procedure described in Section 2.2.

The goal of this test is to demonstrate the ability of our numerical scheme to recover and preserve the stationary solution of Eq. (32) when the initial condition (33) is given by a small spatial perturbation of that stationary solution. We consider a computational domain $\Omega = [-1, 1]$, discretized through a mesh of $N = 50$ computational cells, and we stop our simulations at the final time $t^K = 40\,\mathrm{s}$. We use a CFL number of 0.9 to compute $\Delta t^n = \mathrm{CFL} \cdot \Delta x / \nu$, with

$$\nu = \max(||\mathbf{q}^n||_{L^\infty}, ||\mathcal{V}^n||_{L^\infty}), \tag{36}$$

where $\mathbf{q}$ is the vector containing the cell averages $q_i^n$ of the solution in $S_i$ at time $t^n$, and $\mathcal{V}$ is a vector containing the maximum, on each computational cell, between the left and right states $q^{+,n}_{i-\frac{1}{2}}$ and $q^{-,n}_{i+\frac{1}{2}}$ and their average $\mathcal{M} = 0.5(q^{+,n}_{i-\frac{1}{2}} + q^{-,n}_{i+\frac{1}{2}})$. The time step $\Delta t^n$ is updated at each iteration of the method and, if necessary, when computing space-time predictions. In this way, we account for the highest wave speed that arises in the presence of shocks and rarefactions. Moreover, we solve equation (25) employing a maximum number of iterations equal to the order of accuracy of the method [41].

In Fig. 1C and D, we show the absolute errors between the numerical solution of the steady-state problem obtained through an appropriate Runge-Kutta scheme, and the numerical solution of Eq. (32) obtained using the GRP + DET-WB and the GRP + DET methods. Similarly, we display in Fig. 1E and F the errors between the analytical solution (35) and the numerical solution obtained using the GRP + DET-WB and the GRP + DET methods.

We also perform an empirical convergence test to verify whether our method reaches the expected order of accuracy. We report in Table 1 $L^1$ and $L^\infty$ error norms between the numerical solution obtained with the GRP + DET-WB method and the analytical solution (35), along with the corresponding empirical convergence rates.

### 3.2. System case: Hyperbolized blood flow equations

Let us consider the hyperbolized BFEs system proposed in Müller et al. [38], including now a gravity term. This system of PDEs can be written in quasi-linear form as in Eq. (1), where we have

$$\mathbf{Q} = [A, q, \psi, A_0, h_0, E_e, E_c, p_r]^T, \tag{37}$$

$$\mathbf{A}(\mathbf{Q}) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ c^2 - u^2 & 2u & \frac{A}{\rho}\partial_\psi \zeta & \frac{A}{\rho}\partial_{A_0} \zeta & \frac{A}{\rho}\partial_{h_0} \zeta & \frac{A}{\rho}\partial_{E_e} \zeta & \frac{A}{\rho}\partial_{E_c} \zeta & \frac{A}{\rho} \\ 0 & -1/\epsilon & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \tag{38}$$

with $u = q/A$, and $c = \sqrt{\frac{A}{\rho}\partial_A \zeta}$, and where

$$\mathbf{S}(\mathbf{Q}) = \left[0, Rq/A + Ag_x, -\psi/\epsilon, 0, 0, 0, 0, 0\right]^T. \tag{39}$$

$x$ is the axial coordinate along the vessel, $t$ is the time, $A(x,t)$ represents the cross-sectional area of the vessel lumen, $q(x,t)$ is the flow rate, and $\psi(x,t)$ is an auxiliary variable used in the hyperbolization process [39] along with the relaxation time $\epsilon > 0$. $R(< 0)$ is the coefficient of the friction term, while $g_x(x)$ is the projection of gravity along the vessel's axis defining the gravity term. $A_0(x)$, $h_0(x)$, $E_e(x)$, $E_c(x)$, and $p_r(x)$ are spatial parameters, assumed to be either constant or continuously varying along the vessel's axis. These parameters characterize the vessel's wall mechanics through a nonlinear constitutive relation that links the cross-sectional

area to the blood pressure. More details about the spatial parameters, the adopted pressure-area relation, and the system itself can be found in [38]. Additionally, details of the eigenstructure of this system are given in Müller et al. [38], and more extensively in Montecinos et al. [39], Müller et al. [43]. Finally, we recall that the computation of the boundary states for RP (26), associated with the hyperbolized BFEs, is carried out as described in Section 1 of the supplementary material, while the adopted path $\boldsymbol{\Psi}$ [5,38], referring to the $i+\frac{1}{2}$ interface, reads

$$\boldsymbol{\Psi}(\mathbf{Q}^{-}_{i+\frac{1}{2}}, \mathbf{Q}^{+}_{i+\frac{1}{2}}, s) = \begin{bmatrix} \underline{A}(s) \\ \underline{q}(s) \\ \underline{\psi}(s) \\ \underline{A_0}(s) \\ \underline{h_0}(s) \\ \underline{E_e}(s) \\ \underline{E_c}(s) \\ \underline{p_r}(s) \end{bmatrix} = \begin{bmatrix} \underline{\phi}(s)^{-1} \\ q^{-}_{i+\frac{1}{2}} + s\left(q^{+}_{i+\frac{1}{2}} - q^{-}_{i+\frac{1}{2}}\right) \\ \psi^{-}_{i+\frac{1}{2}} + s\left(\psi^{+}_{i+\frac{1}{2}} - \psi^{-}_{i+\frac{1}{2}}\right) \\ A^{-}_{0,i+\frac{1}{2}} + s\left(A^{+}_{0,i+\frac{1}{2}} - A^{-}_{0,i+\frac{1}{2}}\right) \\ h^{-}_{0,i+\frac{1}{2}} + s\left(h^{+}_{0,i+\frac{1}{2}} - h^{-}_{0,i+\frac{1}{2}}\right) \\ E^{-}_{e,i+\frac{1}{2}} + s\left(E^{+}_{e,i+\frac{1}{2}} - E^{-}_{e,i+\frac{1}{2}}\right) \\ E^{-}_{c,i+\frac{1}{2}} + s\left(E^{+}_{c,i+\frac{1}{2}} - E^{-}_{c,i+\frac{1}{2}}\right) \\ p^{-}_{r,i+\frac{1}{2}} + s\left(p^{+}_{r,i+\frac{1}{2}} - p^{-}_{r,i+\frac{1}{2}}\right) \end{bmatrix}, \quad s \in [0,1]. \tag{40}$$

$\underline{\phi}(s)$ is the specific integration path selected for the cross-sectional area. It is first defined in terms of the total pressure variable, since the Riemann invariants associated with the linearly degenerate characteristic fields are naturally expressed in pressure, namely $\Gamma^{LD}_1 = p + \frac{1}{2}\rho u^2$. Particularly, we consider

$$\underline{\phi}(s) = \underline{\Gamma}(s) - \underline{p_r}(s), \tag{41}$$

where

$$\underline{\Gamma}(s) = \Gamma^{LD,-}_{1,i+\frac{1}{2}} + s\left(\Gamma^{LD,+}_{1,i+\frac{1}{2}} - \Gamma^{LD,-}_{1,i+\frac{1}{2}}\right), \tag{42}$$

and where $\Gamma^{LD,-}_{1,i+\frac{1}{2}}$ is the $\Gamma^{LD}_1$ Riemann invariant of the $i+\frac{1}{2}$ interface from the left. Once the pressure along the path $\underline{\phi}(s)$ is determined, the corresponding cross-sectional area $\underline{A}(s)$ is recovered by inverting the pressure-area relation through a globally convergent Newton method. This construction ensures that steady states with vanishing velocity ($u = 0$) are exactly preserved, so that the resulting scheme is well-balanced with respect to such equilibria.

To test our method on this PDE system, we considered two geometries. First, we used a single blood vessel to verify the empirical convergence of the method to the desired order of accuracy and to compare its efficiency to that of other numerical schemes. Later, we used an arterial network to assess the well-balanced property of the method over a complex geometry and its performance in transient scenarios. Specifically, we considered a reduced version of the anatomically detailed arterial network described in Blanco et al. [44], ADAN86 [45], which is an open network composed of 86 arteries that resemble the real human anatomy. Among all the arteries of ADAN86, we also considered the right internal carotid artery (ICA) as the single blood vessel on which run the first set of tests, due to its peculiar geometry.

The PDE system was solved in each artery of the ADAN86 network using an initial condition given by constant pressure equal to 60 mmHg, and constant zero flow rate. Additionally, three different types of boundary conditions were enforced. At the inlet of the ascending aorta, an inflow boundary condition was set. At the outlet of terminal vessels, either a Dirichlet boundary condition in cross-sectional area or an outflow boundary condition were used. Finally, at the joins between two or three arterial segments, junction coupling conditions were prescribed. Further details on the treatment of these boundary conditions and their assignment in the context of the proposed numerical scheme are given in Müller et al. [38], Müller et al. [43], Müller and Blanco [46]. We emphasize that boundary and coupling conditions computed as reported in these references are used by the implemented GRP-based reconstruction, thereby obviating the need for more sophisticated techniques, such as ghost cell filling, to achieve high-order accuracy. We also underline that, given the high computational cost of computing (8) along junctions, we adopt the local time-stepping algorithm proposed in Müller et al. [38], where, for all tests, the user-defined CFL value is 0.8.

### 3.2.1. *Efficiency analysis for a single blood vessel (ICA test)*

An efficiency analysis was performed to assess if our method was able to attain a prescribed error at low computational cost. To this end, we considered three different scenarios of increasing complexity, all defined on the same blood vessel, the ICA, and subject to identical boundary conditions. The difference among the scenarios arises solely from the choice of both the gravity term and the spatial parameters, in order to highlight the importance of combining appropriate reconstruction techniques with well-balanced solvers to achieve maximal efficiency.

The first scenario (S1) considers the parameters of the pressure-area relation to be constant along all the vessel axis. Specifically, $A_0 = 0.24\ \text{cm}^2$, $h_0 = 0.05$ cm, $E_e = 3.6 \cdot 10^6$ Pa, and $E_c = 9 \cdot 10^8$ Pa. It also assumes the gravity projection $g_x$ to be constant and equal to $981\ \text{cm/s}^2$. For this first scenario, we compared the results obtained using a second-order implementation of the GRP + DET-WB, the GRP + DET, the WENO + DET-WB, and the WENO + DET methods.

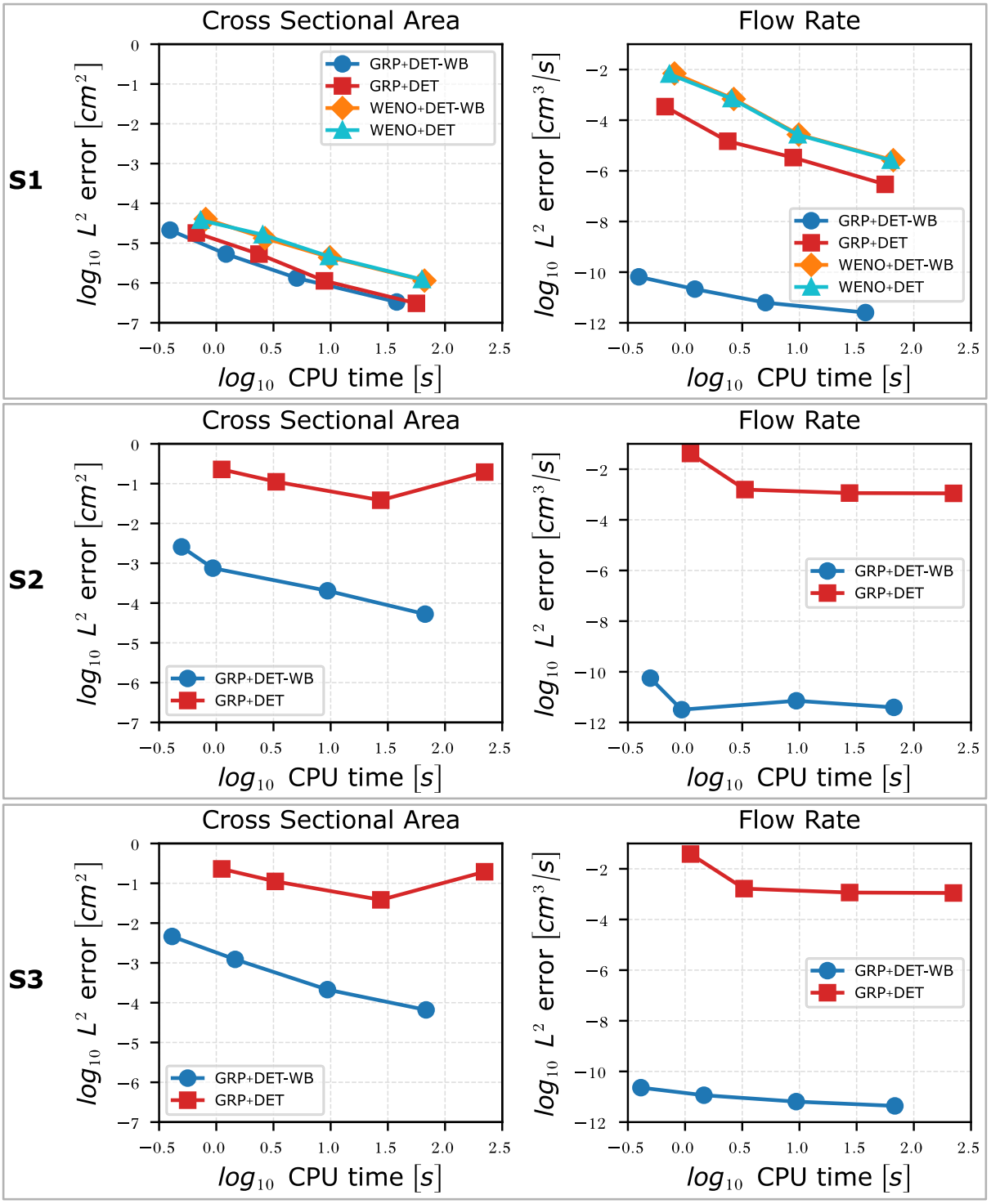



**Fig. 2.** Efficiency plots for ICA test. CPU times versus $L^2$ error norms between the numerical solution and either the exact solution (S1 and S2) or a reference solution (S3), for all the considered scenarios and for 4 consecutive mesh refinements. Each row refers to a different scenario (S1, S2, and S3). Results are shown for both cross-sectional area (left panels) and flow rate (right panels) in logarithmic scale on both axes.

**Table 2**
$L^1$ and $L^\infty$ error norms, and corresponding empirical convergence rates for the cross-sectional area $A(x,t)$ at the final simulation time and for scenarios S1, S2, and S3 for the ICA test, for a second- and a third-order implementation of the numerical scheme. $N$ is the number of computational cells.

| Variable | N | $L^1$ | $L^\infty$ | $O(L^1)$ | $O(L^\infty)$ | $L^1$ | $L^\infty$ | $O(L^1)$ | $O(L^\infty)$ |
|---|---|---|---|---|---|---|---|---|---|
| Scenario S1 | | Order 2 | | | | Order 3 | | | |
| $A$ [cm$^2$] | 4 | 7.84e-05 | 5.92e-06 | — | — | 8.49e-08 | 6.28e-09 | — | — |
| | 8 | 1.96e-05 | 1.48e-06 | 2.0 | 2.0 | 1.08e-08 | 1.17e-09 | 2.97 | 2.42 |
| | 16 | 4.90e-06 | 3.72e-07 | 2.0 | 2.0 | 1.36e-09 | 1.68e-10 | 2.99 | 2.80 |
| | 32 | 1.23e-06 | 9.30e-08 | 2.0 | 2.0 | 1.74e-10 | 2.34e-11 | 2.97 | 2.84 |
| Scenario S2 | | Order 2 | | | | Order 3 | | | |
| $A$ [cm$^2$] | 4 | 8.59e-03 | 9.59e-04 | — | — | 5.90e-03 | 4.37e-04 | — | — |
| | 8 | 2.49e-03 | 2.85e-04 | 1.78 | 1.75 | 9.72e-04 | 1.13e-04 | 2.60 | 1.96 |
| | 16 | 6.71e-04 | 7.84e-05 | 1.89 | 1.86 | 1.36e-04 | 2.03e-05 | 2.84 | 2.47 |
| | 32 | 1.74e-04 | 2.04e-05 | 1.95 | 1.94 | 1.72e-05 | 2.82e-06 | 2.99 | 2.84 |
| Scenario S3 | | Order 2 | | | | Order 3 | | | |
| $A$ [cm$^2$] | 4 | 1.64e-02 | 1.64e-03 | — | — | 9.79e-04 | 1.16e-04 | — | — |
| | 8 | 4.04e-03 | 5.15e-04 | 2.02 | 1.67 | 8.38e-04 | 8.36e-05 | 0.22 | 0.47 |
| | 16 | 6.66e-04 | 9.75e-05 | 2.60 | 2.40 | 7.48e-05 | 1.36e-05 | 3.49 | 2.62 |
| | 32 | 2.05e-04 | 3.70e-05 | 1.70 | 1.40 | 3.17e-05 | 4.34e-06 | 1.24 | 1.65 |

The second scenario (S2) considers the parameters of the pressure-area relation to be variable and continuous in space [44]. Additionally, a smooth function of the gravity projection is taken into account

$$g_x(x) = |g|\left[\exp(-x) - \exp(-L)\right], \tag{43}$$

where $L$ is the axial length of the ICA, while $|g| = 981\,\mathrm{cm/s^2}$ is the gravitational acceleration modulus. In this second case, we only compared results obtained with the second-order GRP + DET-WB method, to those obtained with the second-order GRP + DET method, to better highlight the differences among the well-balanced and the non-well-balanced solvers.

The third and final scenario (S3) considers a variation to the second case, where the gravity projection is now a polyline. Here, we took into account the real 3D geometry of the ICA and we projected the gravitational acceleration on the vessel axis.

For all the scenarios, we assumed a final simulation time of 10 s, and a minimum number of 4 spatial computational cells that are doubled for each of the 4 considered mesh refinements. Additionally, we assumed a no-flow boundary condition at the inlet of the vessel, and fixed cross-sectional area at the outlet such that the corresponding pressure was 60 mmHg. In all the cases, we run our tests on a workstation that had a Intel Core i9 processor with 16 cores and 24 threads (3.2 GHz clock speed), and 64 GiB of RAM, using 1 thread per test.

Fig. 2 shows the obtained results at the final simulation time for a second-order implementation of all the considered methods in terms of CPU time and $L^2$ error norms between the numerical solution and either the exact solution (S1 and S2), or a reference solution (S3) obtained by running a test with 4 times the number of computational cells considered in the final mesh refinement. For all tests, we show results in terms of cross-sectional area and flow rate.

### 3.2.2. Empirical convergence rates for ICA test

An empirical convergence test was carried out to verify whether the proposed method attains its expected theoretical order of accuracy. Convergence rates were computed for the ICA test cases obtained with the GRP + DET-WB method across the three scenarios (S1, S2, and S3). The errors between the computed numerical solutions and either the exact solution (S1 and S2) or a reference solution (S3) were evaluated in both $L^1$ and $L^\infty$ norms. The results, reported in Table 2, are given in terms of cross-sectional area. Errors for the flow rate are omitted, as they consistently remained below $10^{-10}\ cm^3/s$.

### 3.2.3. Well-balance property for the ADAN86 geometry (deadman test)

This test is designed to verify if our method can accurately approximate a specific steady-state solution, characterized by zero flow throughout the network and a hydrostatic pressure distribution (i.e. the pressure varies linearly along the vertical axis of the body due to gravity, with higher values in lower regions, such as in the legs, and lower values in elevated regions, such as in the head).

To this end, we considered the ADAN86 network in the upright posture, enabling the simulation of hydrostatic pressure gradients induced by gravity. Then, we assumed the parameters of the pressure-area relation to be variable and continuous in space. In this test, the gravity term $g_x(x)$ was a polyline. Specifically, we took into account the real 3D geometry of the different vessels and we projected the gravitational acceleration on the vessels' axes. All blood vessels were discretized using a maximum mesh spacing of 1 cm, and the simulations were performed over 20 s. Finally, we assumed no-flow boundary conditions at both inlet and outlets of the terminal vessels.

Results are shown in Fig. 3 (and Fig. 2 of the supplementary material) in terms of flow rate (top panel) and pressure (bottom panel). Specifically, the two networks in each panel display the errors at the final simulation time along the ADAN86 network between

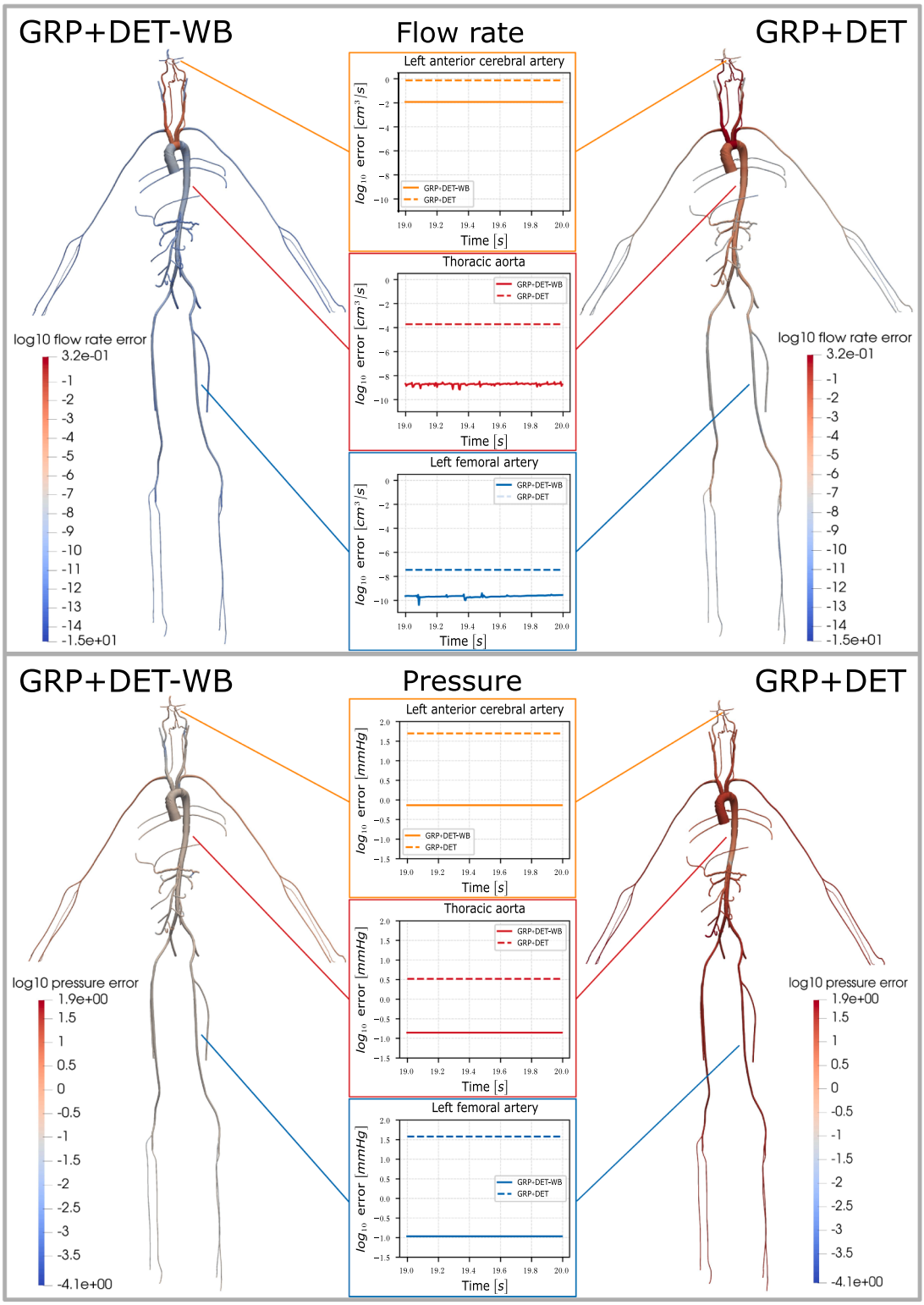



**Fig. 3.** Deadman test. A representation of the ADAN86 network is shown. In the top panel, the colors indicate the errors between zero-flow solution and the second-order numerical solution computed with either the GRP + DET-WB, or the GRP + DET. In the bottom panel, the colors indicate the errors between the reference hydrostatic pressure distribution and the numerical solution computed with either the GRP + DET-WB, or the GRP + DET. A focus on three vessels is provided in the middle of both panels. All the results are shown in logarithmic scale.

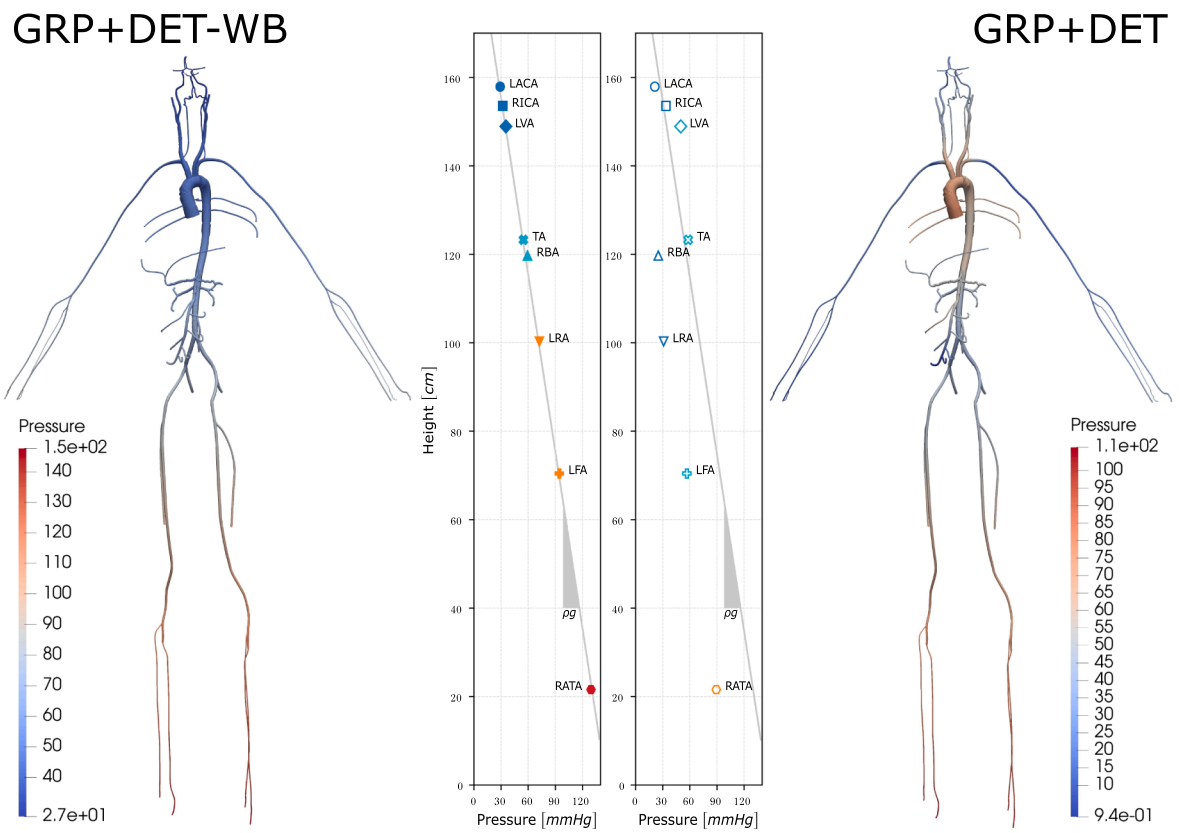



**Fig. 4.** Pressure distribution. Pressure distribution at the final simulation time along the ADAN86 network obtained with a second-order implementation of both the GRP + DET-WB method (left) and the GRP + DET method (right). A focus on eight vessels is provided in the middle of the panel, showing how the GRP + DET-WB results respect the expected hydrostatic distribution indicated in light gray. The considered vessels are: LACA: left anterior cerebral artery, RICA: right internal carotid artery, LVA: left vertebral artery, TA: thoracic aorta, RBA: right brachial artery, LRA: left radial artery, LFA: left femoral artery, RATA: right anterior tibial artery.

the deadman solution and the numerical solutions obtained with a second (and third) order implementation of the GRP + DET-WB and the GRP + DET methods. A focus on three vessels (left anterior cerebral artery, thoracic aorta, and left femoral artery) is provided in the middle of each panel, where results are reported for the midpoint of the vessels axis and for the last second of simulation. Additionally, Fig. 4 (and Fig. 3 of the supplementary material) show the obtained pressure distribution along the network for both the applied methods of order two and three at the final simulation time. A focus on different vessels at different heights of the network is provided in the middle of the panel, where results are displayed for the midpoint of the chosen vessels axis.

### *3.2.4. Transient simulation for ADAN86*

The final test we present aims to assess the capability of our method to accurately capture transient solutions in complex geometries. Specifically, we wanted to test its ability in retrieving pressure and flow rate curves that are commonly observed when modeling the cardiovascular system.

We considered the ADAN86 network in the upright posture and we assumed the parameters of the pressure-area relation to be variable and continuous in space. As in the previous test, the gravity projection $g_x(x)$ was defined as the projection of the gravitational acceleration along the vessels' axes. All blood vessels were discretized using a maximum mesh spacing of 2 cm, and the simulations were performed over 10 s. A periodic inflow boundary condition was prescribed at the aortic inlet [44] to simulate the behavior of the heart (here not present), thereby reproducing the dynamics of the human cardiac cycle. Finally, an outflow boundary condition was enforced by coupling terminal vessels to lumped parameters models that mimic the resistance and compliance of the venous system [38]. We underline that the pressure of the venous system was assumed to be hydrostatically distributed with respect to the vertical axis of the body. In particular, we considered the aortic root to be our reference level.

Results are shown in Fig. 5 (and Fig. 4 of the supplementary material) in terms of flow rate and pressure for three specific vessels (left anterior cerebral artery, thoracic aorta, and left femoral artery) at their midpoint along the final cardiac cycle of simulation. A reference solution, obtained by running the test with gravity set to zero, is also displayed.

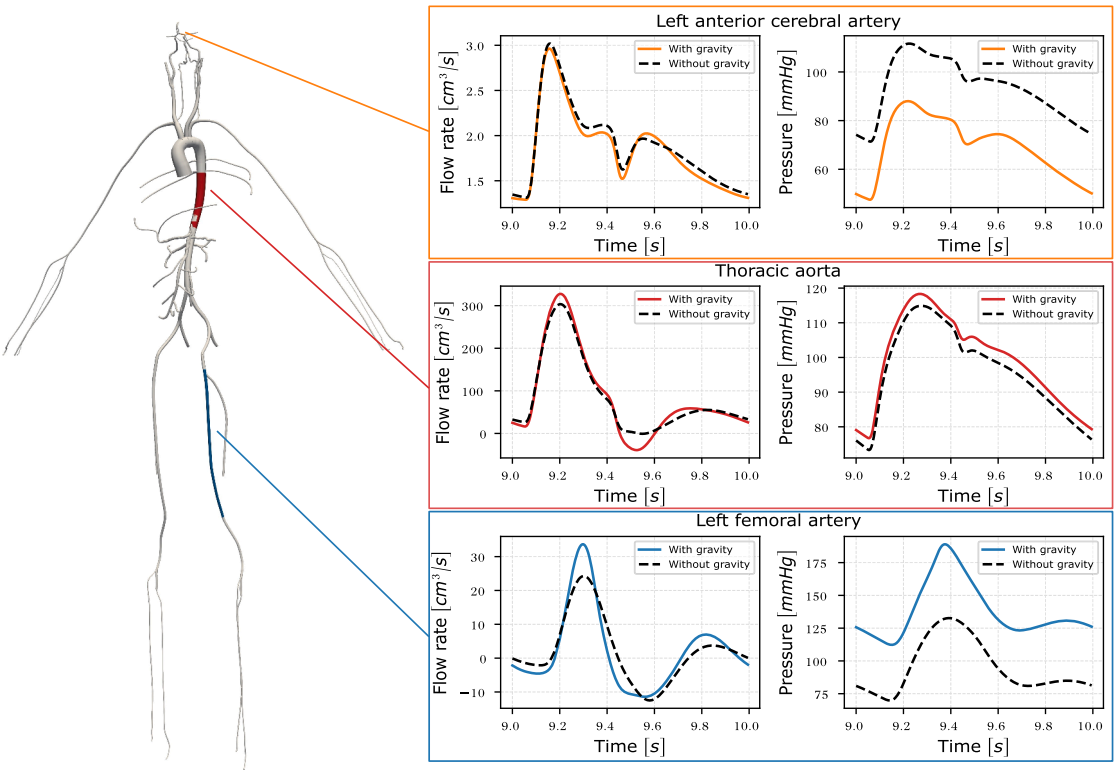



**Fig. 5.** Transient test. Flow rate (left column) and pressure (right column) curves along the final cardiac cycle of simulation at the midpoint of three selected blood vessels: left anterior cerebral artery (orange), thoracic aorta (red), and left femoral artery (blue). Indication on the location of the vessels is shown in the ADAN86 network representation on the left. Black dashed lines represent the reference solution with no gravity in the different vessels. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

## 4. Discussion

In this section, we discuss results obtained through the numerical tests outlined in Section 3, assessing the accuracy and efficiency of our numerical method, and evidencing its limitations.

Empirical convergence results reported in Tables 1 and 2 demonstrate that the GRP + DET-WB scheme converges with the expected order of accuracy to either the analytical steady-state solution or to a reference solution computed on a sufficiently fine mesh. Additionally, results reported in Fig. 1D confirm that the method is well-balanced when applied to find the steady-state solution of problem (32). This proves numerically that, as opposed to the standard WENO reconstruction, the GRP-based reconstruction is well-balanced up to the third order of accuracy by construction. Indeed, as mentioned in Section 2.4, reconstruction polynomials are computed by enforcing that the boundary states associated with each cell are used either to compute the polynomials slopes or as interpolation points, and that the conservation property holds. Consequently, if the cell average is that of the steady-state solution on the same computational cell, and the boundary states lie on the same steady-state solution, then they are not perturbed by the reconstruction procedure.

This is also confirmed from the efficiency plots reported in Fig. 2. When considering scenario S1 (top row), we can notice that the choice of the spatial reconstruction plays a significant role: results obtained through the GRP + DET method present lower errors, with the same spatial discretization, than results obtained through the WENO + DET method. Moreover, the former are comparable, when considering the cross-sectional area, to results obtained through the GRP + DET-WB scheme. As a consequence, we can conclude that the only well-balancing errors introduced within the GRP + DET framework are those associated with the DET solver.

A comparison between numerical results and either the analytical solution of the problem or the reference solution computed on a sufficiently fine mesh (see Figs. 1E and F and 2), shows that, while the GRP + DET-WB and GRP + DET methods perform similarly in simple cases such as the Burgers' equation and scenario S1 for the BFEs, there is an evident advantage in choosing a well-balanced scheme for more complex scenarios. When considering scenarios S2 and S3 for the BFEs, numerical errors for the cross-sectional area and the flow rate obtained through the GRP + DET-WB scheme are significantly lower than those obtained in the GRP + DET setup with analogous meshes, highlighting the higher efficiency of the well-balanced setup. The similar performance of the GRP + DET-WB and GRP + DET schemes in simple scenarios suggests that the approximation of steady-state solutions (see Section 2.2) has a significant impact on the overall numerical errors introduced by the scheme. This poses the problem of choosing an ordinary differential equation

solver of high enough accuracy to avoid the introduction of numerical errors that are comparable or higher than those introduced by the absence of well-balancing.

The ability of our numerical method to preserve stationary solutions is also particularly evident when the method is applied to solve the BFEs over a complex geometry like the ADAN86 network. In fact, observing Fig. 3, top panel, it is evident that the errors in flow rate between the exact solution given by zero-flow and the numerical solution are always higher for the GRP + DET case with respect to the GRP + DET-WB case. A minimum of two orders of magnitude difference between the solutions obtained with the two methods is always detected. Additionally, we observe that the maximum errors for the GRP + DET-WB case are found in the neck and head region, due to its intricate geometry. This region presents multiple vessels junctions, and blood vessels with high geometrical variability, which results in a gravity projection ranging from $-981\,\mathrm{cm/s^2}$ to $981\,\mathrm{cm/s^2}$ along the same vessel axis that can significantly influence the approximation of the solution. Similarly, observing the bottom panel of Fig. 3, we note that the errors in pressure between the deadman solution and the numerical solution are generally larger for the GRP + DET case with respect to the GRP + DET-WB case. These errors strongly affect the representation of the pressure distribution. In particular, we expect to see an hydrostatic pressure distribution along the network when a steady-state is reached, with a maximum pressure in the legs region that gradually decreases towards the cerebral region. This behavior is correctly reproduced in Fig. 4 by the GRP + DET-WB scheme, whereas the GRP-DET scheme fails to capture the hydrostatic distribution, leading to an incorrect approximation of the target solution with a maximal error of $100\,\mathrm{mmHg}$.

Finally, from the pressure and flow-rate waveforms reported in Fig. 5, we observe that the effects of the gravity term are most apparent in the pressure curves. The flow-rate waveforms obtained with gravity are in good agreement with those computed without gravity, with only slight differences in curvature and peak values. In contrast, the pressure waveforms with gravity exhibit a vertical shift relative to those without gravity. This shift reflects the position of the vessel under consideration with respect to the aortic root, which serves as the reference level for the assumed hydrostatic pressure distribution of the venous system. These findings are consistent with previous observations reported in Murillo and Garcia-Navarro [32], Fois et al. [47], confirming that the proposed method is capable of performing transient simulations in complex scenarios, allowing an accurate reproduction of the behavior of the cardiovascular system.

## 5. Conclusion

In this work, we proposed a high-order well-balanced numerical scheme for the solution of non-conservative systems of hyperbolic PDEs with source terms. The method is based on the GRP reconstruction by Montecinos et al. [37], which is well-balanced by construction up to third order of accuracy, provided that boundary states belong to the same family of the steady-state solution of the considered PDEs and the conservation property holds. In addition, it incorporates a modification of the DET solver [41] along the same lines as those proposed by Guerrero Fernández et al. [10]. Numerical tests performed on the Burgers' equation and on the hyperbolized BFEs [38] with friction, gravity and variable geometrical properties demonstrate the well-balanced property of the scheme and highlight its accuracy and efficiency in complex scenarios.

However, it is important to mention that in this work we employ the GRP-based reconstruction without making use of any limiter, which is linear in Godunov's sense. As a consequence, spurious oscillations might arise in the presence of discontinuities. A possible strategy to overcome this issue would be to employ an a-posteriori limiter such as the Multi-dimensional Optimal Order Detection (MOOD) strategy [48]. In addition, the proposed scheme does not guarantee the physical admissibility of the numerical solution. In particular, it does not belong to the class of positivity-preserving schemes and therefore provides no control on the positivity of physically relevant quantities, such as the cross-sectional area in the BFEs system. Nevertheless, in all the numerical experiments presented in this work, no loss of positivity of the cross-sectional area has been observed. This behavior is likely related to the choice of physically relevant initial and boundary conditions, as well as to the time step restrictions adopted in the simulations. The design of suitable positivity-preserving modifications of the present method represents a relevant direction for future research [49].

## CRediT authorship contribution statement

**Chiara Colombo:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Formal analysis; **Caterina Dalmaso:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology; **Lucas O. Müller:** Writing – review & editing, Supervision, Methodology, Conceptualization; **Annunziato Siviglia:** Writing – review & editing, Supervision, Methodology, Conceptualization.

## Data availability

No data was used for the research described in the article.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgments

C.C. and L.O.M acknowledge funding by the European Union under NextGenerationEU, Mission 4, Component 2 - PRIN 2022 (D.D. 104/22), project title: Immersed methods for multIscale and multiphysics problems, CUP: E53D2300592 0006. C.C. and A.S. acknowledge the Italian Ministry of Education, Universities and Research (MUR), in the framework of the project DICAM-EXC (Departments of Excellence 2023-2027, grant L232/2016). C.C., C.D., and L.O.M. are members of the "Gruppo Nazionale per il Calcolo Scientifico" dell' Istituto Nazionale di Alta Matematica (INdAM-GNCS, Italy) .

## Supplementary material

Supplementary material associated with this article can be found in the online version at 10.1016/j.jcp.2026.114975.

## References

[1] A. Siviglia, D. Vanzo, E.F. Toro, A splitting scheme for the coupled Saint-Venant-Exner model, Adv. Water Resour. 159 (2022) 104062. https://doi.org/10.1016/j.advwatres.2021.104062
[2] E.G. Fernández, M.J.C. Díaz, M. Dumbser, T.M. De Luna, An arbitrary high order well-balanced ADER-DG numerical scheme for the multilayer shallow-water model with variable density, J. Sci. Comp. 90 (1) (2022) 52. https://doi.org/10.1007/s10915-021-01734-2
[3] E. Gaburro, M.J. Castro, M. Dumbser, Well-balanced Arbitrary-Lagrangian-Eulerian finite volume schemes on moving nonconforming meshes for the Euler equations of gas dynamics with gravity, Mon. Not. R. Astro. Soc. 477(2) (2018) 2251–2275. https://doi.org/10.1093/mnras/sty542
[4] A. Bermudez, X. López, M.E. Vázquez-Cendón, Numerical solution of non-isothermal non-adiabatic flow of real gases in pipelines, J. Comp. Phys. 323 (2016) 126–148. https://doi.org/10.1016/j.jcp.2016.07.020
[5] L.O. Müller, E.F. Toro, Well-balanced high-order solver for blood flow in networks of vessels with variable properties, Int. J. Numer. Meth. Biomed. Engng. 29 (2013) 1388–1411. https://doi.org/10.1002/cnm.2580
[6] E.F. Toro, A. Siviglia, Flow in collapsible tubes with discontinuous mechanical properties: mathematical model and exact solutions, Commun. Comput. Phys. 13(2) (2013) 361–385. https://doi.org/10.4208/cicp.210611.240212a
[7] G. Dal Maso, P.G. Le Floch, F. Murat, Definition and weak stability of nonconservative products, J. Math. Pure Appl. 74(6) (1995) 483–548.
[8] J. Murillo, A. Navas-Mantilla, P. Garcia-Navarro, Formulation of exactly balanced solvers for blood flow in elastic vessels and their application to collapsed states, Comput. Fluid. 186 (2019) 74–98. https://doi.org/10.1016/j.compfluid.2019.04.008
[9] M.J. Castro Díaz, T. Chacón Rebollo, E.D. Fernández-Nieto, C. Parés, On well-balanced finite volume methods for nonconservative nonhomogeneous hyperbolic systems, SIAM J. Sci. Comput. 29(3) (2007) 1093-1126. https://doi.org/10.1137/040607642
[10] E. Guerrero Fernández, C. Escalante, M.J. Castro Díaz, Well-Balanced high-order discontinuous Galerkin methods for systems of balance laws, Mathematics 10(1) (2022) 15. https://doi.org/10.3390/math10010015
[11] A. Bermudez, M.E. Vazquez, Upwind methods for hyperbolic conservation laws with source terms, Comput. Fluid. 23(8) (1994) 1049–1071. https://doi.org/10.1016/0045-7930(94)90004-3
[12] J.P. Berberich, P. Chandrashekar, C. Klingenberg, High order well-balanced finite volume methods for multi-dimensional systems of hyperbolic balance laws, Comput. Fluid. 219 (2021) 104858. https://doi.org/10.1016/j.compfluid.2021.104858
[13] L. Gosse, A well-balanced flux-vector splitting scheme designed for hyperbolic systems of conservation laws with source terms, Comput. Math. Appl. 39 (9–10) (2000) 135–159. https://doi.org/10.1016/S0898-1221(00)00093-6
[14] J.M. Greenberg, A.Y. Leroux, R. Baraille, A. Noussair, Analysis and approximation of conservation laws with source terms, SIAM J. Numer. Anal. 34(5) (1997) 1980–2007. https://doi.org/10.1137/S0036142995286751
[15] I. Gómez-Bueno, M.J. Castro, C. Parés, High-order well-balanced methods for systems of balance laws: a control-based approach, Appl. Math. Comput. 394 (2021) 125820. https://doi.org/10.1016/j.amc.2020.125820
[16] C. Parés, C. Parés-Pulido, Well-balanced high-order finite difference methods for systems of balance laws, J. Comput. Phys. 425 (2021) 109880. https://doi.org/10.1016/j.jcp.2020.109880
[17] M.J. Castro, C. Parés, Well-balanced high-order finite volume methods for systems of balance laws, J. Sci. Comput. 82(2) (2020) 48. https://doi.org/10.1007/s10915-020-01149-5
[18] E. Guerrero Fernandez, M.J. Castro-Diaz, T. Morales de Luna, A second-order well-balanced finite volume scheme for the multilayer shallow water model with variable density, Mathematics 8(5) (2020) 848. https://doi.org/10.3390/math8050848
[19] A. Canestrelli, A. Siviglia, M. Dumbser, E.F. Toro, Well-balanced high-order centred schemes for non-conservative hyperbolic systems. applications to shallow water equations with fixed and mobile bed, Adv. Water Resour. 32(6) (6) (2009) 834–844. https://doi.org/10.1016/j.advwatres.2009.02.006
[20] T.C. Rebollo, A.D. Delgado, E.D.F. Nieto, A family of stable numerical solvers for the shallow water equations with source terms, Comput. Method. Appl. M. 192 (1–2) (2003) 203–225. https://doi.org/10.1016/S0045-7825(02)00551-0
[21] S. Noelle, N. Pankratz, G. Puppo, J.R. Natvig, Well-balanced finite volume schemes of arbitrary order of accuracy for shallow water flows, J. Comput. Phys. 213(2) (2006) 474–499. https://doi.org/10.1016/j.jcp.2005.08.019
[22] P. Chandrashekar, M. Zenk, Well-balanced nodal discontinuous Galerkin method for Euler equations with gravity, J. Sci. Comput. 71(3) (2017) 1062–1093. https://doi.org/10.1007/s10915-016-0339-x
[23] V. Desveaux, M. Zenk, C. Berthon, C. Klingenberg, A well-balanced scheme to capture non-explicit steady states in the Euler equations with gravity, Int. J. Num. Meth. Fl. 81(2) (2016) 104–127. https://doi.org/10.1002/fld.4177
[24] L. Grosheintz-Laval, R. Käppeli, High-order well-balanced finite volume schemes for the Euler equations with gravitation, J. Comput. Phys. 378 (2019) 324–343. https://doi.org/10.1016/j.jcp.2018.11.018
[25] R. Käppeli, S. Mishra, Well-balanced schemes for the Euler equations with gravitation, J. Comput. Phys. 259 (2014) 199–219. https://doi.org/10.1016/j.jcp.2013.11.028
[26] O. Delestre, P.-Y. Lagrée, A well-balanced finite volume scheme for blood flow simulation, Int. J. Numer. Meth. Fl. 72 (2) (2013) 177–205. https://doi.org/10.1002/fld.3736
[27] J. Murillo, P. Garcia-Navarro, A Roe type energy balanced solver for 1D arterial blood flow and transport, Comput. Fluid. 117 (2015) 149–167. https://doi.org/10.1016/j.compfluid.2015.05.003
[28] G. Li, O. Delestre, L. Yuan, Well-balanced discontinuous Galerkin method and finite volume WENO scheme based on hydrostatic reconstruction for blood flow model in arteries, Int. J. Numer. Meth. Fl. 86 (7) (2018) 491–508. https://doi.org/10.1002/fld.4463
[29] B. Ghitti, C. Berthon, M.H. Le, E.F. Toro, A fully well-balanced scheme for the 1D blood flow equations with friction source term, J. Comput. Phys. 421 (2020) 109750. https://doi.org/10.1016/j.jcp.2020.109750
[30] J. Britton, Y. Xing, Well-balanced discontinuous Galerkin methods for the one-dimensional blood flow through arteries model with man-at-eternal-rest and living-man equilibria, Comput. Fluid. 203 (2020) 104493. https://doi.org/10.1016/j.compfluid.2020.104493
[31] A. Spilimbergo, E.F. Toro, L.O. Müller, et al., One-dimensional blood flow with discontinuous properties and transport: mathematical analysis and numerical schemes, Commun. Comput. Phys. 2121 (3) (2021) 649–697. https://doi.org/10.4208/cicp.OA-2020-0132

[32] J. Murillo, P. Garcia-Navarro, Numerical coupling of 0D and 1D models in networks of vessels including transonic flow conditions. application to short-term transient and stationary hemodynamic simulation of postural changes, Int. J. Numer. Method Biomed. Eng. 39(11) (2023) e3751. https://doi.org/10.1002/cnm.3751
[33] A. Harten, ENO schemes with subcell resolution, J. Comp. Phys. 83 (1) (1989) 148–184. https://doi.org/10.1016/0021-9991(89)90226-X
[34] X.-D. Liu, S. Osher, T. Chan, Weighted essentially non-oscillatory schemes, J. Comput. Phys. 115(1) (1994) 200–212. https://doi.org/10.1006/jcph.1994.1187
[35] B. Van Leer, Towards the ultimate conservative difference scheme. V. A second-order sequel to Godunov's method, J. Comp. Phys. 32 (1) (1979). https://doi.org/10.1016/0021-9991(79)90145-1
[36] M. Castro, J.M. Gallardo, J.A. López-GarcÍa, C. Parés, Well-balanced high order extensions of Godunov method for linear balance laws, SIAM J. Numer. Anal. 46 (2) (2008) 1012–1039.
[37] G.I. Montecinos, E.F. Toro, L.O. Müller, A generalized Riemann problem-based compact reconstruction method for finite volume schemes, arXiv preprint (2026). https://doi.org/10.48550/arXiv.2602.21911
[38] L.O. Müller, A.J. Blanco, S.M. Watanabe, R.A. Feijóo, A high-order local time stepping finite volume solver for one-dimensional blood flow simulations: application to the ADAN model, Int. J. Numer. Meth. Biomed. Engng. 32(10) (2016) e02761. https://doi.org/10.1002/cnm.2761
[39] G.I. Montecinos, L.O. Müller, E.F. Toro, Hyperbolic reformulation of a 1D viscoelastic blood flow model and ADER finite volume schemes, J. Comput. Phys. 266 (2014) 101–123. https://doi.org/10.1016/j.jcp.2014.02.013
[40] C. Parés, Numerical methods for nonconservative hyperbolic systems: a theoretical framework, SIAM J. Numer. Anal. 44 (1) (2006) 300–321. https://doi.org/10.1137/050628052
[41] M. Dumbser, C. Enaux, E.F. Toro, Finite volume schemes of very high order of accuracy for stiff hyperbolic balance laws, J. Comput. Phys. 227 (2008) 3971-4001. https://doi.org/10.1016/j.jcp.2007.12.005
[42] E.F. Toro, Riemann Solvers and Numerical Methods for Fluid Dynamics, Springer, 2009.
[43] L.O. Müller, G. Leugering, P.J. Blanco, Consistent treatment of viscoelastic effects at junctions in one-dimensional blood flow models, J. Comput. Phys. 314 (2016) 167–193. https://doi.org/10.1016/j.jcp.2016.03.012
[44] P.J. Blanco, S.M. Watanabe, M.A.R.F. Passos, P.A. Lemos, R.A. Feijóo, An anatomically detailed arterial network model for one-dimensional computational hemodynamics, IEEE Trans. Biomed. Eng. 62(2) (2015) 736–753.
[45] P.J. Blanco, L.O. Müller, S.M. Watanabe, R.A. Feijóo, On the anatomical definition of arterial networks in blood flow simulations: comparison of detailed and simplified models, Biomech. Model Mechan. 19 (2020) 663-1678. https://doi.org/10.1007/s10237-020-01298-4
[46] L.O. Müller, P.J. Blanco, A high order approximation of hyperbolic conservation laws in networks: application to one-dimensional blood flow, J. Comp. Phys. 300 (2015) 423–437. https://doi.org/10.1016/j.jcp.2015.07.056
[47] M. Fois, S.V. Maule, M. Giudici, M. Valente, L. Ridolfi, S. Scarsoglio, Cardiovascular response to posture changes: multiscale modeling and in vivo validation during head-up tilt, Front. Physiol. 13 (2022) 826989. https://doi.org/10.3389/fphys.2022.826989
[48] Clain, S. Diot, R. Loubère, A high-order finite volume method for systems of conservation laws-multi-dimensional optimal order detection (MOOD), J. Comp. Phys. 230 (10) (2011) 4028–4050. https://doi.org/10.1016/j.jcp.2011.02.026
[49] Q.-M. Huang, H. Zhou, Y.-X. Ren, Q. Wang, A general positivity-preserving algorithm for implicit high-order finite volume schemes solving the Euler and Navier-Stokes equations, J. Comp. Phys. 508 (2024) 112999. https://doi.org/10.1016/j.jcp.2024.112999