POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

High-order discontinuous Galerkin methods for 1D vascular flows: from single-scale models to 3D-1D multiscale haemodynamics

Tesi di Laurea Magistrale in
Aeronautical Engineering - Ingegneria Aeronautica

Author: **Pietro Businaro**

Student ID: 10738998
Advisor: Prof. Luca Dede'
Co-advisor: Dr. Michele Bucelli
Academic Year: 2024-2025

Alla mia famiglia,
ai miei amici e compagni di vita.

«Non si vede bene che col cuore.
L'essenziale è invisibile agli occhi.»

(Antoine de Saint-Exupéry, *Il piccolo principe*)

Abstract

One-dimensional (1D) modeling of vascular flows belongs to the family of reduced-order models for haemodynamics; it is well established in literature to study the wave propagation along arteries, while remarkably reducing the computational burden with respect to three-dimensional (3D) fluid-structure interaction (FSI) models. The objective of this work is to present the underlying mathematical formulation of the 1D haemodynamics equations and to devise a suitable numerical approximation, capable of capturing the waveforms with a minimum amount of diffusion and dispersion errors. For this purpose, we adopt discontinuous Galerkin (DG) finite element methods for the space discretization and exploit high-order polynomials in order to describe the waveforms accurately. The time discretization is achieved by means of Adams-Bashforth (AB) schemes. In this context, the literature assumes a flat velocity profile within the vessel cross section to approximate the non-linear momentum transport term, thus leading to considerable mathematical simplifications. However, in this work, we avoid this hypothesis by deriving a proper expression for Roe's linearization applied to the 1D haemodynamics equations, meaning that the numerical method can be readily applied to an arbitrary velocity profile. After that, we extend the formulation to discontinuous material properties (e.g. representing the installation of a stent) and to complex blood vessel networks (i.e. accounting for junction points). In addition, a possible strategy to couple the current 1D solver with 3D FSI solvers is discussed. Relevant numerical experiments appearing in this work include a verification against benchmark solutions from literature (which demonstrates the correctness of the numerical method and its implementation), the study of a single pressure pulse in a 3D-1D coupled simulation (to highlight possible numerical issues arising at the coupling interface) and a detailed convergence analysis (where we show the sub-optimality of AB schemes for the numerical approximation at hand).

Keywords: reduced-order models for haemodynamics, multiscale simulations for haemodynamics, cardiovascular modeling, blood vessel networks, discontinuous Galerkin methods, spectral/*hp* element methods.

Abstract in lingua italiana

La modellazione monodimensionale (1D) del flusso sanguigno appartiene alla famiglia di modelli di ordine ridotto per l'emodinamica; questo modello è consolidato in letteratura per studiare la propagazione di onde nelle arterie, riducendo drasticamente l'onere computazionale rispetto a modelli tridimensionali (3D) di interazione fluido-struttura (FSI). L'obiettivo di questo lavoro è quello di presentare la formulazione matematica delle equazioni dell'emodinamica 1D e di ideare un'opportuna approssimazione numerica capace di catturare le forme d'onda, introducendo un errore minimo di diffusione e dispersione. A tal proposito, adottiamo metodi agli elementi finiti di Galerkin discontinui (DG) per la discretizzazione nello spazio e sfruttiamo polinomi di ordine elevato al fine di descrivere le forme d'onda accuratamente. La discretizzazione nel tempo è realizzata tramite schemi di Adams-Bashforth (AB). In questo contesto, le letteratura assume un profilo di velocità piatto nella sezione trasversale del vaso, al fine di approssimare il termine non-lineare di trasporto della quantità di moto, portando così a semplificazioni matematiche notevoli. Tuttavia, in questo lavoro, evitiamo questa assunzione e deriviamo un'espressione opportuna per la linearizzazione di Roe applicata alle equazioni dell'emodinamica 1D, il che implica che il metodo può essere applicato a un profilo di velocità arbitrario. Dopodiché, estendiamo la formulazione a proprietà discontinue del materiale (rappresentando, ad esempio, l'installazione di uno stent) e ad una rete complessa di vasi sanguigni (ossia, teniamo in considerazione la presenza di punti di giunzione). In aggiunta, viene discussa una possibile strategia per accoppiare questo modello 1D con modelli FSI 3D. Gli esperimenti numerici rilevanti riportati in questo lavoro includono una verifica rispetto a risultati noti in letteratura (volta a dimostrare la correttezza del metodo numerico e della sua implementazione), lo studio di un singolo impulso di pressione in una simulazione accoppiata 3D-1D (per evidenziare possibili problemi numerici all'interfaccia di accoppiamento) e un'analisi di convergenza dettagliata (dove mostriamo la subottimalità degli schemi AB per la presente approssimazione numerica).

Parole chiave: modelli di ordine ridotto per l'emodinamica, simulazioni multiscala per l'emodinamica, modellazione cardiovascolare, reti di vasi sanguigni, metodi di Galerkin discontinui, metodi agli elementi spettrali/*hp*.

Contents

1 | Introduction

One-dimensional (1D) modeling of the vascular flow in compliant vessels [1] is extensively employed in literature for the analysis of pressure and flow waveforms within arteries. Compared to three-dimensional (3D) fluid-structure interaction (FSI) models, 1D models offer a substantial reduction in the computational burden, while describing the wave propagation phenomena associated with pressure and flow rate. At the same time, 1D haemodynamics models overcome the limitations of zero-dimensional (0D) lumped-parameter Windkessel models, which are not capable of providing the spatial distribution of the physical quantities. Some 1D simulations of complex networks of arteries have been performed in e.g. [2, 3, 4, 5, 6], showing a considerable consistency with 3D FSI models and experimental data. Recent developments and applications to clinical test cases are further analyzed in [7, 8].

However, for numerical modeling purposes, the literature assumes a flat velocity profile to describe the blood flow within vessels, in order to approximate the momentum transport term and obtain considerable mathematical simplifications. This hypothesis is adopted in the existing literature regarding 1D haemodynamics modeling, see e.g. [2, 3, 5, 6, 9, 10, 11], and is justified by physical considerations [10]. In this work, we devise a numerical technique that allows to avoid this assumption and express the 1D blood flow equations in the most general case, with an arbitrary velocity profile – more specifically, we consider a parabolic velocity profile for lower-Womersley flows and a blunt velocity profile for higher-Womersley flows. From a numerical standpoint, this is achieved by applying a discontinuous Galerkin (DG) finite element method, in its spectral/*hp* element variant (also known as DG-SEM), coupled with an Adams-Bashforth (AB) time-advancing scheme, as proposed in [2, 11]. To avoid assuming a flat velocity profile, we extend Roe's linearization [12] for the problem at hand, in order to compute the intercell fluxes appearing in the DG-SEM discretization. In hyperbolic problems, DG-SEM schemes are particularly attractive, since they are able to capture wave propagation with limited dispersion and diffusion errors. Moreover, spectral/*hp* element methods exhibit an exponential convergence with respect to the degree of the polynomials chosen as basis functions.

The objective of this work is to formulate the numerical approximation for the 1D haemodynamics equations with a general velocity profile, extend it to a network of vessels and discuss a possible technique coupling this 1D blood flow model to a 3D FSI model. The standalone 1D model is verified against the results reported in [5], while we offer some relevant numerical experiments about the 3D-1D coupling, in order to study possible issues related to wave reflections at the coupling interface.

The entire implementation has been carried out within the `life`^{x} project[1.1]. `life`^{x} is a C++ high-performance library focused on numerical simulations with cardiovascular applications [13, 14]. It relies on `deal.II` open-source library[1.2] for the finite element implementation [15].

The numerical experiments conducted in this work include:

- a verification against benchmark solutions (Section 7.1);
- a comparison between the exact Riemann solver and Roe's approximate Riemann solver, where we analyze the error induced by Roe's linearization (Section 7.2);
- a test case on a stented vessel (Section 7.3), to show the relevance of these 1D models when discontinuous material properties are introduced;
- the propagation of a single pressure pulse in a 3D-1D coupled simulation (Section 7.4), to underline possible numerical artifacts arising at the coupling interface;
- convergence analysis (Section 7.5), where we assess the convergence rate in terms of mesh spacing (h-convergence), polynomial degree (p-convergence) and time step (time convergence); in particular, h-convergence exhibits super-convergence (as it may occur in DG schemes [16]), while p-convergence shows an exponential decay of the error (as predicted by spectral/hp element theory); on the other hand, we are not able to prove an optimal rate for the time convergence on the current numerical setup, and further insights are discussed in Section 7.5.3;
- parallel scalability (Section 7.6), where it is not possible to reach a perfect scalability, due to the excessive overhead in exchanging the ghost entries.

The rest of this thesis is organized as follows. Chapter 2 formulates the 1D haemodynamics equations and introduces the corresponding characteristic analysis. Chapter 3 shows the DG-SEM formulation, the time stepping procedure and the way we handle boundary conditions; in this section, we also present Roe's linearization applied to the 1D

[1.1] https://lifex.gitlab.io/lifex-public/
[1.2] https://dealii.org/

haemodynamics equations, which has been derived specifically for this work. Chapter 4 extends the mathematical and numerical formulations to complex networks of blood vessels, and Chapter 5 proposes a possible 3D-1D coupling strategy. Finally, the code design is discussed in Chapter 6, while numerical results are reported in Chapter 7.

2 | Mathematical modeling

In this chapter, the mathematical formulation of 1D vascular flow equations is introduced, together with the simplifying hypotheses, the hyperbolic nature of the problem and its characteristic analysis. Finally, we discuss the common approach in literature, which assumes a flat velocity profile when writing the momentum transport term, and analyze the reasons why it is physically inconsistent but yet accepted for numerical modeling purposes.

2.1. 1D haemodynamics model

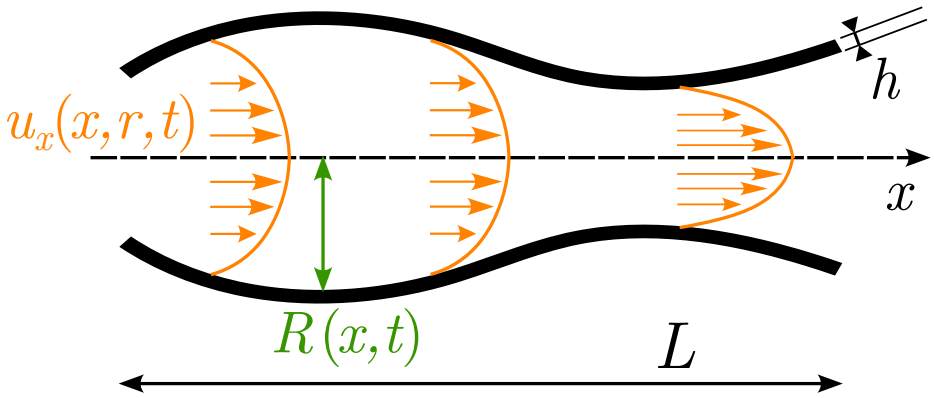


Figure 2.1: Straight blood vessel. x denotes the axial coordinate, $u_x(x,r,t)$ is the axial component of the velocity, $R(x,t)$ the vessel radius, L the vessel length, h the wall thickness.

In haemodynamic context, the blood is usually assumed to be an incompressible Newtonian fluid. In addition, by introducing a cylindrical reference frame whose axial coordinate x is aligned along the vessel axis (as illustrated in Figure 2.1), we can formulate the following hypotheses, needed to derive the 1D haemodynamics model:

- *axial symmetry:* all the quantities introduced hereafter are independent of the angular coordinate θ;
- *material properties:* each blood vessel is modeled as a straight, deformable, imper-

meable and perfectly circular tube, whose properties (i.e. Young modulus E, wall thickness h and Poisson ratio ξ) vary only along the axial coordinate x; the vessel radius R depends on both the axial coordinate x and the time t;

- *radial displacement:* the displacement of each cross section occurs only along the radial direction;
- *constant pressure:* the pressure P is assumed to be constant within the same cross section, meaning that it depends solely on the axial coordinate x and on the time t;
- *absence of body forces:* the contribution of the body forces is neglected (e.g. gravity effects are not taken into consideration);
- *dominance of axial velocity:* the axial component u_x of the velocity is dominant with respect to the other components, yielding the following approximation for the velocity:

$$\mathbf{u}(x,r,t) \approx u_x(x,r,t)\ \mathbf{e}_x = u(x,t)\ s\left(\frac{r}{R(x,t)}\right)\mathbf{e}_x\ , \tag{2.1}$$

where $\mathbf{e}_x$ represents the unit vector aligned with x-direction, $u(x,t)$ is the average velocity over the cross section, $s\left(\frac{r}{R}\right)$ is a suitable velocity profile, r the radial coordinate and $R(x,t)$ the vessel radius; a possible choice for $s\left(\frac{r}{R}\right)$ is reported in the relation (2.6).

By applying these hypotheses and integrating the 3D incompressible Navier-Stokes equations over the cross-sectional lumen area of the blood vessel, we can derive the following mass and momentum balance equations for the 1D vascular flow [1]:

$$\begin{cases} \dfrac{\partial A}{\partial t} + \dfrac{\partial Q}{\partial x} = 0 & \forall x \in \Omega = (x_L, x_R),\ \forall t \in (0, t_f] \\ \dfrac{\partial Q}{\partial t} + \dfrac{\partial (\alpha Q^2/A)}{\partial x} + \dfrac{A}{\rho}\dfrac{\partial P}{\partial x} + K_R \dfrac{Q}{A} = 0 & \forall x \in \Omega = (x_L, x_R),\ \forall t \in (0, t_f] \end{cases} \tag{2.2}$$

where:

- x_L and x_R represent the left endpoint and the right endpoint of the 1D domain Ω, respectively, while t_f is the final time;
- ρ is the blood density (which is constant under the incompressibility hypothesis);
- $A(x,t)$ is the cross-sectional lumen area;
- $Q(x,t)$ is the volumetric flow rate, equal to $Q(x,t) = A(x,t)\ u(x,t)$;
- the pressure $P(x,t)$ is linked to the cross-sectional area A through a constitutive

law, usually represented by the following tube law:

$$P(x,t) = P_{\text{ref}} + \psi(A, A_{\text{ref}}, \beta) = P_{\text{ref}} + \beta \, \frac{\sqrt{A} - \sqrt{A_{\text{ref}}}}{A_{\text{ref}}} \, , \tag{2.3}$$

where P_{ref} and A_{ref} are the reference pressure and area, while $\beta := \dfrac{\sqrt{\pi} h E}{1 - \xi^2}$ represents the wall properties (h is the wall thickness, E the Young modulus and ξ the Poisson ratio); we refer to [1] for the derivation of this constitutive law;

- α is the momentum flux correction factor, defined as follows:

$$\alpha := \frac{\int_S \rho u_x^2 \mathrm{d}S}{\rho u^2 A} = \frac{1}{A} \int_S \left[s\Big(\frac{r}{R}\Big) \right]^2 \mathrm{d}S \, , \tag{2.4}$$

  from which it is clear that α depends on the selected velocity profile $s\Big(\dfrac{r}{R}\Big)$; however, the literature assumes $\alpha = 1$ independently of the velocity profile, since it leads to considerable simplifications (as later discussed in Sections 2.4–2.5 and in Appendix A), but the physical meaning is partially lost; the objective of this work is to devise a numerical technique applicable also to the case $\alpha \neq 1$, in order to avoid this physical inconsistency;

- the viscous coefficient K_R is derived by integration of the viscous term from the 3D incompressible Navier-Stokes equations:

$$K_R = -2\pi\nu \left. \frac{\mathrm{d}s}{\mathrm{d}r} \right|_{r=R} \, , \tag{2.5}$$

  with $\nu = \dfrac{\mu}{\rho}$ being the kinematic viscosity; more specifically, K_R represents the wall shear stress and thus depends on the velocity gradient evaluated at the wall, i.e. $\left. \dfrac{\mathrm{d}s}{\mathrm{d}r} \right|_{r=R}$.

In literature, the following polynomial function of order γ is well established to describe the velocity profile $s\Big(\dfrac{r}{R}\Big)$:

$$s\Big(\frac{r}{R}\Big) = \frac{\gamma + 2}{\gamma} \left[1 - \Big(\frac{r}{R}\Big)^{\gamma} \right] \, , \tag{2.6}$$

as depicted in Figure 2.2. Therefore, the momentum flux correction factor (2.4) and the

viscous coefficient (2.5) can be computed as functions of γ:

$$\alpha = \frac{\gamma + 2}{\gamma + 1} , \qquad K_R = 2\pi\nu \, (\gamma + 2) . \tag{2.7}$$

On the other hand, a flat velocity profile implies:

$$s\left(\frac{r}{R}\right) = 1 , \qquad \alpha = 1 , \qquad K_R = 0 , \tag{2.8}$$

where we remark that the velocity gradient in radial direction is null, so the flow is considered essentially inviscid (i.e. $K_R = 0$).

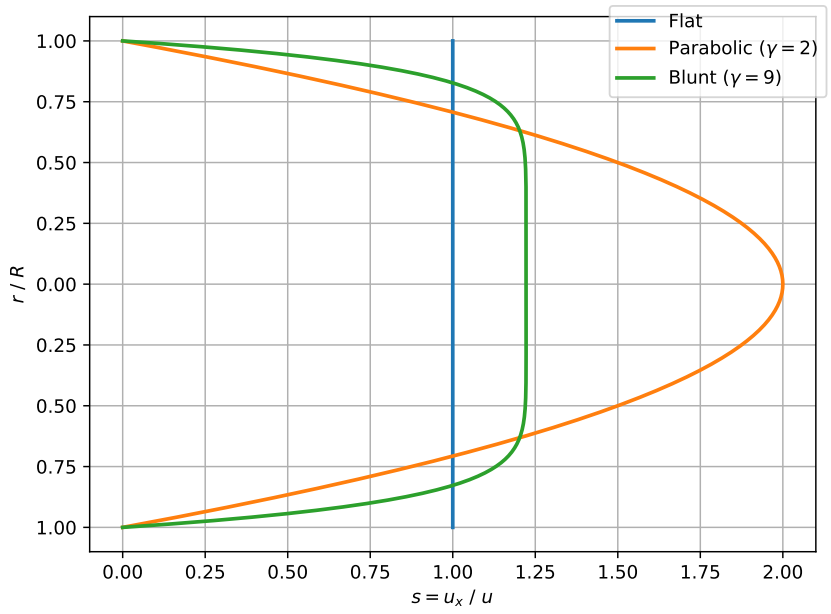


Figure 2.2: Flat, parabolic and blunt velocity profiles.

The choice of γ is driven by experimental data. Common values are $\gamma = 2$ for a parabolic velocity profile and $\gamma = 9$ for a blunt velocity profile [5], as plotted in Figure 2.2. The choice between parabolic and blunt velocity profiles depends on the Womersley number Wo, which in turn can be expressed as function of the Reynolds number $\mathrm{Re} = \dfrac{UD}{\nu}$ and the Strouhal number $\mathrm{St} = \dfrac{D}{UT}$:

$$\mathrm{Wo} = R\sqrt{\frac{2\pi}{\nu T}} = \sqrt{\frac{\pi}{2}\,\mathrm{Re}\,\mathrm{St}} , \tag{2.9}$$

with R being the blood vessel radius, ν the kinematic viscosity and T the pulse period. Wo is proportional to the radius of the blood vessel over the laminar boundary layer

growth within a pulse period T (see [17] for a more detailed analysis). In other words, Wo describes two relevant features of the vascular flow:

1. the ratio between inertial and viscous forces, as provided by the Reynolds number Re;
2. the oscillatory motion, as represented by the Strouhal number St.

Since the kinematic viscosity ν and the pulse period T are approximately constant within the cardiovascular system, Wo depends mainly on the radius of the vessel R. That is to say, in larger vessels (e.g. the aorta or the vena cava), the Womersley number is larger ($\mathrm{Wo} \gtrsim 10$) and this implies a blunt velocity profile (i.e. $\gamma = 9$); on the contrary, smaller vessels exhibits a small Womersley number ($\mathrm{Wo} \approx 1$), with a parabolic profile (i.e. $\gamma = 2$), as discussed in [6].

2.2. Quasi-linear form

By properly grouping the terms in (2.2), the 1D haemodynamics equations can be rewritten in their *quasi-linear form*:

$$\frac{\partial \mathbf{U}}{\partial t} + \mathrm{H}(\mathbf{U})\frac{\partial \mathbf{U}}{\partial x} + \mathbf{B}(\mathbf{U}) = \mathbf{0} \qquad \forall x \in \Omega = (x_L, x_R),\ \forall t \in (0, t_f]\ , \tag{2.10}$$

where:

$$\begin{aligned} \mathbf{U} &= (A, Q)^\top\ , \\ \mathrm{H}(\mathbf{U}) &= \begin{bmatrix} 0 & 1 \\ c^2 - \alpha\left(\dfrac{Q}{A}\right)^2 & 2\alpha\dfrac{Q}{A} \end{bmatrix}\ , \\ \mathbf{B}(\mathbf{U}) &= \begin{pmatrix} 0 \\ K_R\dfrac{Q}{A} + \dfrac{A}{\rho}\dfrac{\partial \psi}{\partial A_{\mathrm{ref}}}\dfrac{\partial A_{\mathrm{ref}}}{\partial x} + \dfrac{A}{\rho}\dfrac{\partial \psi}{\partial \beta}\dfrac{\partial \beta}{\partial x} \end{pmatrix}\ , \end{aligned} \tag{2.11}$$

with $c := \sqrt{\dfrac{A}{\rho}\dfrac{\partial \psi}{\partial A}}$. It is worth noting that c is the analytical wave propagation speed only in a purely inviscid case (i.e. without pressure losses, as proven in [18]). For this reason, we will indicate c as the *wave speed* hereafter, even though it just approximates the real wave speed in viscous flows. Thus, by introducing the tube law (2.3), the wave speed c

becomes:

$$c = \sqrt{\frac{\beta}{2\rho A_{\text{ref}}}} A^{\frac{1}{4}} \ . \tag{2.12}$$

2.3. Conservative form

It is possible to further manipulate the system of equations (2.2) and to rewrite it in its *conservative form*:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}(\mathbf{U})}{\partial x} + \mathbf{S}(\mathbf{U}) = \mathbf{0} \qquad \forall x \in \Omega = (x_L, x_R),\ \forall t \in (0, t_f] \ , \tag{2.13}$$

where:

$$\begin{aligned} \mathbf{U} &= (A, Q)^\top \ , \\ \mathbf{F}(\mathbf{U}) &= \begin{pmatrix} Q \\ \alpha \dfrac{Q^2}{A} + C \end{pmatrix} \ , \\ \mathbf{S}(\mathbf{U}) &= \begin{pmatrix} 0 \\ K_R \dfrac{Q}{A} + \dfrac{A}{\rho} \dfrac{\partial \psi}{\partial A_{\text{ref}}} \dfrac{\partial A_{\text{ref}}}{\partial x} + \dfrac{A}{\rho} \dfrac{\partial \psi}{\partial \beta} \dfrac{\partial \beta}{\partial x} - \dfrac{\partial C}{\partial A_{\text{ref}}} \dfrac{\partial A_{\text{ref}}}{\partial x} - \dfrac{\partial C}{\partial \beta} \dfrac{\partial \beta}{\partial x} \end{pmatrix} \ , \end{aligned} \tag{2.14}$$

with $\mathbf{F}(\mathbf{U})$ being the non-linear flux vector and $\mathbf{S}(\mathbf{U})$ the non-linear reaction term. Moreover, we have indicated by C a primitive function of c^2, i.e. $C := \displaystyle\int_0^A c^2 \mathrm{d}A$, which, in the case of the tube law (2.3), is equal to:

$$C = \frac{\beta}{3\rho A_{\text{ref}}} A^{\frac{3}{2}} \ . \tag{2.15}$$

Finally, we highlight that, in our implementation, the material properties are assumed to be constant within the same blood vessel, i.e. A_{ref} and β are constant along the same blood vessel, thus yielding the following reaction term:

$$\mathbf{S}(\mathbf{U}) = \mathbf{B}(\mathbf{U}) = \begin{pmatrix} 0 \\ K_R \dfrac{Q}{A} \end{pmatrix} \ . \tag{2.16}$$

However, it is possible to interface vessels with different material properties by introducing suitable matching conditions, as discussed in Sections 4.1–4.2.

2.4. Characteristic analysis

In this section, we perform the characteristic analysis of the hyperbolic system (2.10), as discussed in [1, 19]. If the cross-sectional area is strictly positive (i.e. $A > 0$, which is physically meaningful), the system (2.10) admits the following pair of eigenvalues:

$$\lambda_{1,2} = \alpha\frac{Q}{A} \pm c_\alpha \tag{2.17}$$

with:

$$c_\alpha := \sqrt{c^2 + \alpha(\alpha-1)\left(\frac{Q}{A}\right)^2}\,. \tag{2.18}$$

Under physiological conditions, the wave speed is larger than the average velocity over the cross section, i.e. $c \gg \alpha\dfrac{Q}{A}$. Consequently, the problem at hand is considered *strictly hyperbolic* and *subcritical*, since the eigenvalues $\lambda_{1,2} \in \mathbb{R}$ are real and distinct, and do not change their sign (i.e. $\lambda_1 > 0,\ \lambda_2 < 0,\ \forall A \in \mathbb{R}^+,\ \forall Q \in \mathbb{R}$). This means that the solution of the hyperbolic problem (2.10) is characterized by two waves traveling in opposite directions.

The left and right eigenvectors of $\mathrm{H}(\mathbf{U})$ are mutually orthogonal and defined up to a constant. Hence, we choose them to be also orthonormal, i.e. by requiring $\mathbf{l}_i \cdot \mathbf{r}_j = \delta_{ij}$ (with δ_{ij} being the Kronecker delta). This yields:

$$\mathbf{l}_{1,2}(\mathbf{U}) = \zeta(A,Q)\begin{pmatrix} \pm c_\alpha - \alpha\dfrac{Q}{A} \\ 1 \end{pmatrix}\,, \qquad \mathbf{r}_{1,2}(\mathbf{U}) = \frac{1}{2c_\alpha\zeta(A,Q)}\begin{pmatrix} \pm 1 \\ c_\alpha \pm \alpha\dfrac{Q}{A} \end{pmatrix}\,, \tag{2.19}$$

with $\zeta(A,Q) > 0$ being an arbitrary function. Then, we define the following matrices, collecting the left eigenvectors, the right eigenvectors and the eigenvalues, respectively:

$$\mathrm{L}(\mathbf{U}) := \begin{bmatrix} \mathbf{l}_1^\top \\ \mathbf{l}_2^\top \end{bmatrix}\,, \qquad \mathrm{R}(\mathbf{U}) := \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 \end{bmatrix}\,, \qquad \Lambda(\mathbf{U}) := \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}\,. \tag{2.20}$$

In this way, $\mathrm{H}(\mathbf{U})$ can be diagonalized as: $\Lambda(\mathbf{U}) = \mathrm{L}(\mathbf{U})\,\mathrm{H}(\mathbf{U})\,\mathrm{R}(\mathbf{U})$. Finally, we can assume to have two characteristic variables $W_{1,2}$ satisfying:

$$\frac{\partial W_{1,2}}{\partial \mathbf{U}} = \mathbf{l}_{1,2}(\mathbf{U}) \tag{2.21}$$

such that the system (2.10) can be rewritten into its *characteristic form*:

$$\frac{\partial \mathbf{W}}{\partial t} + \Lambda(\mathbf{U}) \frac{\partial \mathbf{W}}{\partial x} + \mathrm{L}(\mathbf{U})\, \mathbf{B}(\mathbf{U}) = \mathbf{0} \tag{2.22}$$

with $\mathbf{W} := (W_1, W_2)^\top$. For $W_{1,2}$ to be derived in an analytical and global form, we must require, in addition to (2.21):

$$\frac{\partial^2 W_1}{\partial Q \partial A} = \frac{\partial^2 W_1}{\partial A \partial Q}\,, \qquad \frac{\partial^2 W_2}{\partial Q \partial A} = \frac{\partial^2 W_2}{\partial A \partial Q}\,. \tag{2.23}$$

It can be shown that one possible choice of $\zeta(A, Q)$ satisfying (2.23) when $\alpha = 1$ (i.e. for a flat velocity profile) is given by: $\zeta(A, Q) = A^{-1}$. The intermediate steps can be found in [1] and are reported in detail in Appendix A. The resulting characteristic variables can be computed by integrating (2.21) and by introducing the tube law (2.3):

$$\begin{aligned} W_{1,2} &= \frac{Q}{A} \pm \int_0^A \frac{c}{A} \mathrm{d}A \\ &= \frac{Q}{A} \pm 4c \\ &= \frac{Q}{A} \pm 4\sqrt{\frac{\beta}{2\rho A_{\mathrm{ref}}}} A^{\frac{1}{4}}\,, \end{aligned} \tag{2.24}$$

where $c = c(A)$ is provided by the relation (2.12).

We remark that the expression (2.24) holds only for a flat velocity profile. On the other hand, for a generic velocity profile (i.e. with $\alpha \neq 1$), the integration (2.21) cannot be carried out analytically, so the expression (2.24) does not hold anymore and $W_{1,2}$ cannot be derived in a global form. Further details are discussed in Appendix A.

2.5. A common (yet inconsistent) approach in literature

It is worth mentioning that a common approach in literature (e.g. employed in [2, 5, 6, 10, 11]) is to consider a unit momentum flux correction factor (i.e. $\alpha = 1$ for any velocity profile, instead of using the expression given in (2.7)), while the viscous effects are introduced by means of the usual viscous coefficient (i.e. $K_R = 2\pi\nu\,(\gamma + 2)$, as already provided in (2.7)). These assumptions yield the following mass and momentum balance

equations for the 1D blood flow model (equivalent to (2.2) but with $\alpha = 1$):

$$\begin{cases} \dfrac{\partial A}{\partial t} + \dfrac{\partial Q}{\partial x} = 0 & \forall x \in \Omega = (x_L, x_R),\ \forall t \in (0, t_f] \\ \dfrac{\partial Q}{\partial t} + \dfrac{\partial (Q^2/A)}{\partial x} + \dfrac{A}{\rho}\dfrac{\partial P}{\partial x} + K_R \dfrac{Q}{A} = 0 & \forall x \in \Omega = (x_L, x_R),\ \forall t \in (0, t_f] \end{cases} \tag{2.25}$$

In other words, a flat velocity profile is assumed when writing the non-linear momentum transport term (i.e. the second term in the momentum balance equation), whereas the viscosity is taken into consideration by assuming an arbitrary non-flat velocity profile. This simplification is physically inconsistent, because a flat velocity profile occurs only for an inviscid flow (i.e. K_R should be null when $\alpha = 1$). However, it is commonly accepted, since, when $\alpha = 1$, we have at our disposal the analytical expression (2.24) for the characteristic variables $W_{1,2}$, which can be employed to compute the intercell fluxes exactly (Section 3.4.1), without resorting to (more complex) numerical approximations (Section 3.4.2).

Moreover, the mathematical model (2.25) can be further manipulated by substituting $Q = Au$, thus passing from a $(A, Q)^\top$ formulation to a $(A, u)^\top$ formulation, yielding:

$$\begin{cases} \dfrac{\partial A}{\partial t} + \dfrac{\partial (Au)}{\partial x} = 0 & \forall x \in \Omega = (x_L, x_R),\ \forall t \in (0, t_f] \\ \dfrac{\partial u}{\partial t} + \dfrac{\partial (u^2/2)}{\partial x} + \dfrac{1}{\rho}\dfrac{\partial P}{\partial x} + K_R \dfrac{u}{A} = 0 & \forall x \in \Omega = (x_L, x_R),\ \forall t \in (0, t_f] \end{cases} \tag{2.26}$$

That is to say, only when $\alpha = 1$, we can rearrange (2.26) into its *conservative form*, which reads (with a little abuse of notation):

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}(\mathbf{U})}{\partial x} + \mathbf{S}(\mathbf{U}) = \mathbf{0} \tag{2.27}$$

with $\mathbf{U} = (A, u)^\top$. In this case, the flux term is given by:

$$\mathbf{F}(\mathbf{U}) = \mathbf{F}(A, u) = \begin{pmatrix} Au \\ \dfrac{u^2}{2} + \dfrac{P}{\rho} \end{pmatrix} = \begin{pmatrix} Q \\ \dfrac{P_{\text{tot}}}{\rho} \end{pmatrix} , \tag{2.28}$$

where the total pressure P_{tot} is defined as follows:

$$P_{\text{tot}} := P + \frac{1}{2}\rho u^2 = P + \frac{1}{2}\rho \left(\frac{Q}{A}\right)^2 . \tag{2.29}$$

This implies that enforcing the continuity of the fluxes is equivalent to requiring the conti-

nuity of the flow rate Q and the total pressure P_{tot} for a flat velocity profile, which serves as a mathematical justification for the matching conditions at discontinuous material points (Section 4.1) and bifurcation points (Section 4.2).

As stated in [10], the choice $\alpha = 1$ represents a physiologically reasonable approximation in large arteries (e.g. within the aorta), where the Womersley number is sufficiently large (i.e. Wo ≈ 10) and thus the velocity profile can be considered mostly flat. One the other hand, one may argue that this simplification invalidates the model for lower-Womersley flows. The objective of this work is to devise a numerical strategy allowing to choose the momentum flux correction factor α depending on the selected velocity profile (i.e. $\alpha = \dfrac{\gamma+2}{\gamma+1}$, as already derived in (2.7)), avoiding any inconsistency.

3 | Numerical modeling

The objective of this chapter is to provide a suitable numerical approximation of the mathematical model introduced in Chapter 2. The space discretization is achieved by means of a high-order discontinuous Galerkin (DG) method, while the problem is advanced in time using explicit Adams-Bashforth (AB) schemes. Indeed, since the problem is inherently subcritical and thus free of shocks (as shown in Section 2.4), high-order methods are particularly attractive. Indeed, these schemes achieve exponential convergence with respect to the polynomial degree for sufficiently smooth solutions, yielding significantly higher accuracy for a given number of degrees of freedom compared to conventional low-order finite element discretizations. Finally, in this chapter, we deal with the treatment of the intercell fluxes, and the handling of the boundary conditions in the context of DG methods.

3.1. Space discretization with DG method

In order to obtain a numerical discretization of problem (2.2), we partition the 1D domain $\Omega = (x_L, x_R)$ into a mesh of M non-overlapping elements and $M+1$ nodes. Each element is denoted by Ω_m, $\forall m \in \{1, \dots, M\}$, with nodes (x_L^m, x_R^m) such that:

- $x_R^m \equiv x_L^{m+1} \quad \forall m = \{1, \dots, M-1\}$,
- $x_L^1 \equiv x_L$,
- $x_R^M \equiv x_R$.

Formally, this is equivalent to:

$$\bigcup_{m=1}^{M} \overline{\Omega}_m = \overline{\Omega} \, . \tag{3.1}$$

Hence, we can discretize the 1D blood flow model in space by applying a DG approximation to the conservative form (2.13). Consequently, the weak formulation reads: find

$\mathbf{U} \in V$ such that:

$$\sum_{m=1}^{M} \left\{ \left(\frac{\partial \mathbf{U}}{\partial t}, \mathbf{v} \right)_{\Omega_m} + \left(\frac{\partial \mathbf{F}(\mathbf{U})}{\partial x}, \mathbf{v} \right)_{\Omega_m} + \Big(\mathbf{S}(\mathbf{U}), \mathbf{v} \Big)_{\Omega_m} \right\} = 0 \qquad \forall \mathbf{v} \in V \,, \tag{3.2}$$

where:

- V is a suitable Sobolev space;
- $\mathbf{v}(x) \in V$ is a suitable test function;
- $\big(\mathbf{u}, \mathbf{v}\big)_{\Omega_m} := \displaystyle\int_{\Omega_m} \mathbf{u} \cdot \mathbf{v} \, \mathrm{d}x$ represents the L^2-inner product over the m-th cell.

By integrating the flux term (i.e. the second term in (3.2)) by parts, we obtain:

$$\sum_{m=1}^{M} \left\{ \left(\frac{\partial \mathbf{U}}{\partial t}, \mathbf{v} \right)_{\Omega_m} - \left(\mathbf{F}(\mathbf{U}), \frac{\mathrm{d}\mathbf{v}}{\mathrm{d}x} \right)_{\Omega_m} + \Big[\mathbf{F}(\mathbf{U}) \cdot \mathbf{v} \Big]_{x_L^m}^{x_R^m} + \Big(\mathbf{S}(\mathbf{U}), \mathbf{v} \Big)_{\Omega_m} \right\} = 0 \qquad \forall \mathbf{v} \in V \,. \tag{3.3}$$

In order to guarantee the stability and consistency of the method, we need to rework the third term in (3.3) (containing the interface flux). In the DG framework, the solution is approximated by functions that are allowed to be discontinuous across element interfaces, resulting in a jump discontinuity at each interface between two consecutive cells. Consequently, the interface flux cannot be evaluated unambiguously from the local solution and must be replaced by a single-valued *numerical flux*, denoted by $\mathbf{F}^*$, which depends on the solution evaluated on both sides of the interface. Moreover, the choice of $\mathbf{F}^*$ must ensure the stability and consistency of the numerical approximation. A rigorous derivation and analysis of the numerical flux, based on the solution of the Riemann problem at each interface, is discussed in Section 3.4.

We denote by V_h the discrete space of piecewise polynomials of degree p:

$$V_h = \Big\{ \mathbf{v}_h \in V : \mathbf{v}_h\big|_{\Omega_m} \in \big[\mathbb{P}_p(\Omega_m)\big]^2, \quad m = 1, \ldots, M \Big\} \tag{3.4}$$

with $\mathbb{P}_p(\Omega_m)$ being the space of polynomials up to order p defined on the m-th element. In this way, each computational cell is characterized by $p+1$ degrees of freedom per component.

After that, we rewrite (3.3) into the discrete finite elements space $V_h \subset V$ (3.4) and introduce the numerical flux $\mathbf{F}^*$ into the third term of (3.3), yielding the following discrete

weak formulation: find $\mathbf{U}_h \in V_h$ such that:

$$\sum_{m=1}^{M} \left\{ \left(\frac{\partial \mathbf{U}_h}{\partial t}, \mathbf{v}_h \right)_{\Omega_m} - \left(\mathbf{F}(\mathbf{U}_h), \frac{\mathrm{d}\mathbf{v}_h}{\mathrm{d}x} \right)_{\Omega_m} + \left[\mathbf{F}^*(\mathbf{U}_h) \cdot \mathbf{v}_h \right]_{x_L^m}^{x_R^m} + \left(\mathbf{S}(\mathbf{U}_h), \mathbf{v}_h \right)_{\Omega_m} \right\} = 0 \qquad \forall \mathbf{v}_h \in V_h \, . \tag{3.5}$$

Let $E = M + 1$ be the total number of nodes of the 1D computational domain. Hence, the third term in (3.5) (containing the numerical flux) can be rewritten as a summation over the e-th node, formally:

$$\sum_{m=1}^{M} \left[\mathbf{F}^*(\mathbf{U}_h) \cdot \mathbf{v}_h \right]_{x_L^m}^{x_R^m} = \sum_{e=1}^{E} \mathbf{F}^*(\mathbf{U}_{h,L}^e, \mathbf{U}_{h,R}^e) \cdot (\mathbf{v}_{h,L}^e - \mathbf{v}_{h,R}^e) \qquad \forall \mathbf{v}_h \in V_h \, , \tag{3.6}$$

where $\mathbf{v}_{h,L}^e$ and $\mathbf{v}_{h,R}^e$ represent the discontinuous test functions in V_h evaluated at the left-hand side and right-hand side of the e-th node, respectively. Similar definitions follow for $\mathbf{U}_{h,L}^e$ and $\mathbf{U}_{h,R}^e$. The way we handle these interface terms, together with the numerical flux $\mathbf{F}^*$, will be clarified in Section 3.4 for interior nodes, and in Section 3.5 for boundary nodes. By substituting these interface terms (3.6) into (3.5), we obtain the discrete weak formulation: find $\mathbf{U}_h \in V_h$ such that:

$$\begin{aligned} \sum_{m=1}^{M} \left\{ \left(\frac{\partial \mathbf{U}_h}{\partial t}, \mathbf{v}_h \right)_{\Omega_m} - \left(\mathbf{F}(\mathbf{U}_h), \frac{\mathrm{d}\mathbf{v}_h}{\mathrm{d}x} \right)_{\Omega_m} + \left(\mathbf{S}(\mathbf{U}_h), \mathbf{v}_h \right)_{\Omega_m} \right\} + \\ + \sum_{e=1}^{E} \mathbf{F}^*(\mathbf{U}_{h,L}^e, \mathbf{U}_{h,R}^e) \cdot (\mathbf{v}_{h,L}^e - \mathbf{v}_{h,R}^e) = 0 \\ \forall \mathbf{v}_h \in V_h \, . \end{aligned} \tag{3.7}$$

Finally, let $\{\varphi_i^m\}_{i=0}^p$ be a basis functions of $\mathbb{P}_p(\Omega_m)$ defined on the m-th cell, $\forall m \in \{1, \ldots, M\}$, such that $\mathbf{v}_h \in V_h$ can be expressed as a linear combination of those basis functions:

$$\mathbf{v}_h(x,t) = \sum_{m=1}^{M} \sum_{i=0}^{p} \widehat{\mathbf{v}}_i^m(t)\, \varphi_i^m(x) \qquad \forall \mathbf{v}_h \in V_h \, . \tag{3.8}$$

For the sake of simplicity, we collect into $\{\varphi_i\}_{i=1}^N$ the basis functions defined on all the cells, with N being the total number of degrees of freedom per component (i.e. $N = M\,(p+1)$).

Hence, (3.8) can be rewritten in a more compact way:

$$\mathbf{v}_h(x,t) = \sum_{i=1}^{N} \widehat{\mathbf{v}}_i(t)\,\varphi_i(x) \qquad \forall \mathbf{v}_h \in V_h\ . \tag{3.9}$$

By introducing the expression (3.9) into (3.7), we can define the algebraic counterpart of the weak formulation:

$$\mathrm{M}_h \frac{\mathrm{d}\widehat{\mathbf{U}}}{\mathrm{d}t} = \mathbf{f}_h(\widehat{\mathbf{U}})\ , \tag{3.10}$$

where M_h is the mass matrix:

$$(\mathrm{M}_h)_{ij} := \sum_{m=1}^{M} \Big(\varphi_j, \varphi_i\Big)_{\Omega_m}\ , \tag{3.11}$$

while $\mathbf{f}_h(\widehat{\mathbf{U}})$ collects the non-linear flux and reaction contributions. In this context, we would like to have a diagonal (and thus trivially-invertible) mass matrix M_h, in order to decouple the system of ordinary differential equations (ODEs) given by (3.10). One common approach for DG methods is to use Legendre polynomials as the basis functions for the discrete finite element space V_h. Legendre polynomials are a system of polynomials orthogonal within the interval $[-1, 1]$, thus satisfying:

$$\Big(\phi_n, \phi_m\Big)_{[-1,1]} = \int_{-1}^{1} \phi_n\,\phi_m\,\mathrm{d}x = 0 \qquad \forall n \neq m\ . \tag{3.12}$$

In addition, if we also require $\phi_n(1) = 1$, we can obtain the Legendre polynomials illustrated in Figure 3.1. We refer to [20, 21] for a more theoretical perspective about Legendre polynomials.

This class of numerical schemes is known in literature as spectral/*hp* element methods. We refer to [2, 11] for a more in-depth analysis of these numerical methods applied to 1D vascular flow modeling.

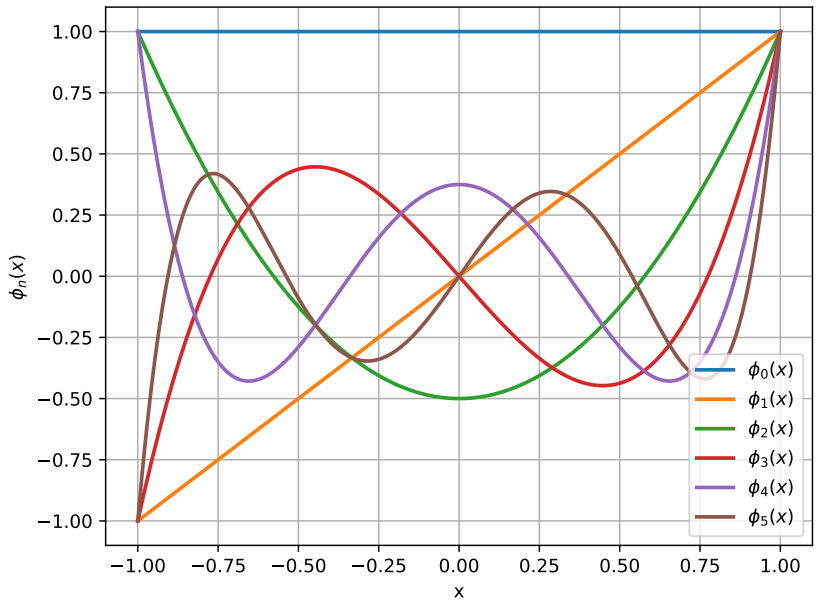


Figure 3.1: Legendre polynomials $\phi_n(x)$, up to 5^{th} order.

3.2. Time discretization with AB scheme

Given the choice of Legendre polynomials as basis functions for V_h, M_h is diagonal and thus can be trivially inverted, leading to the following system of ODEs:

$$\frac{\mathrm{d}\widehat{\mathbf{U}}}{\mathrm{d}t} = \mathrm{M}_h^{-1}\,\mathbf{f}_h(\widehat{\mathbf{U}}) := \mathbf{f}(\widehat{\mathbf{U}})\ , \tag{3.13}$$

which is discretized using an explicit AB time-advancing scheme, as proposed in [2, 11]. The core idea behind AB schemes is to find a suitable interpolation of the the right-hand side $\mathbf{f}(\widehat{\mathbf{U}})$, identified by $\mathbf{L}(t)$:

$$\mathbf{f}(\widehat{\mathbf{U}}) \approx \mathbf{L}(t) := \sum_{i=0}^{s-1} \mathbf{f}(\widehat{\mathbf{U}}^{n+i})\,\ell_i(t)\ , \tag{3.14}$$

where $\widehat{\mathbf{U}}^{n+i} := \widehat{\mathbf{U}}(t^{n+i})$, while $\ell_i(t)$ represents the i-th Lagrange polynomial of order $(s-1)$, defined as follows:

$$\ell_i(t) := \prod_{\substack{j=0 \\ j\neq i}}^{s-1} \frac{t - t_{n+j}}{t_{n+i} - t_{n+j}} \qquad \forall i \in \{0, \dots, s-1\}\ . \tag{3.15}$$

Thus, we perform the integration of (3.13) over the time step $[t^{n+s-1}, t^{n+s}]$, yielding the following update rule for AB time-stepping schemes:

$$\begin{aligned}\widehat{\mathbf{U}}^{n+s} &= \widehat{\mathbf{U}}^{n+s-1} + \int_{t^{n+s-1}}^{t^{n+s}} \mathbf{f}(\widehat{\mathbf{U}})\,\mathrm{d}t \\ &\approx \widehat{\mathbf{U}}^{n+s-1} + \int_{t^{n+s-1}}^{t^{n+s}} \mathbf{L}(t)\,\mathrm{d}t \\ &= \widehat{\mathbf{U}}^{n+s-1} + \Delta t \sum_{i=0}^{s-1} c_i\,\mathbf{f}(\widehat{\mathbf{U}}^{n+i})\ ,\end{aligned} \tag{3.16}$$

where s represents the order of the AB scheme, Δt the time step and $\{c_i\}_{i=0}^{s-1}$ the coefficients resulting from the integration of $\ell_i(t)$ over a single time step, formally:

$$\int_{t^{n+s-1}}^{t^{n+s}} \ell_i(t)\,\mathrm{d}t = \Delta t\, c_i \qquad \forall i \in \{0, \dots, s-1\}\ . \tag{3.17}$$

It should be noticed that the values of c_i depend on the selected order s. For completeness, we report the update rule (3.16) up to 4^{th} order, where we highlight that the case $s = 1$ corresponds to a forward Euler scheme:

$$\begin{aligned}
s = 1 \quad &\Rightarrow \quad \widehat{\mathbf{U}}^{n+1} = \widehat{\mathbf{U}}^{n} + \Delta t\,\mathbf{f}(\widehat{\mathbf{U}}^{n}) \\
s = 2 \quad &\Rightarrow \quad \widehat{\mathbf{U}}^{n+2} = \widehat{\mathbf{U}}^{n+1} + \Delta t \left(\frac{3}{2}\mathbf{f}(\widehat{\mathbf{U}}^{n+1}) - \frac{1}{2}\mathbf{f}(\widehat{\mathbf{U}}^{n})\right) \\
s = 3 \quad &\Rightarrow \quad \widehat{\mathbf{U}}^{n+3} = \widehat{\mathbf{U}}^{n+2} + \Delta t \left(\frac{23}{12}\mathbf{f}(\widehat{\mathbf{U}}^{n+2}) - \frac{16}{12}\mathbf{f}(\widehat{\mathbf{U}}^{n+1}) + \frac{5}{12}\mathbf{f}(\widehat{\mathbf{U}}^{n})\right) \\
s = 4 \quad &\Rightarrow \quad \widehat{\mathbf{U}}^{n+4} = \widehat{\mathbf{U}}^{n+3} + \Delta t \left(\frac{55}{24}\mathbf{f}(\widehat{\mathbf{U}}^{n+3}) - \frac{59}{24}\mathbf{f}(\widehat{\mathbf{U}}^{n+2}) + \frac{37}{24}\mathbf{f}(\widehat{\mathbf{U}}^{n+1}) - \frac{9}{24}\mathbf{f}(\widehat{\mathbf{U}}^{n})\right)\ .
\end{aligned} \tag{3.18}$$

We refer to [20] for a more theoretical perspective about AB time-advancing schemes.

Given the hyperbolic nature of the problem at hand, a Courant-Friedrichs-Levy (CFL) restriction is necessary to ensure the stability of the method. The semi-empirical study

conducted in [22] proposes a CFL criterion in the form:

$$\Delta t < \Delta t_{\max} = \kappa(s)\,\frac{h}{\lambda_{\max}\,p^2}\ , \tag{3.19}$$

where $\kappa(s)$ is a suitable multiplying factor (depending on the AB order s), h the characteristic mesh size, $\lambda_{\max}$ the maximum eigenvalue of the Jacobian matrix (i.e. $\lambda_{\max} = \max_{i\in\{1,2\}} |\lambda_i|$ for the problem at hand, see (2.17)) and p the polynomial degree. It is worth mentioning that $\kappa(s)$ is strictly related to the chosen time stepper and is proportional to the intersection of the stability region of the time-advancing scheme with the imaginary axis (we refer to [22] for the complete derivation). That is to say, $\kappa(s)$ decreases as the order s of the AB scheme is increased, so a smaller time step Δt is needed for high-order AB schemes. Moreover, given $\Delta t_{\max} = \mathcal{O}(p^{-2})$, also higher-degree polynomials lead to stricter time steps.

3.3. L^2-projection of the initial conditions

The time-stepping scheme defined in Section 3.2 must be provided with suitable initial conditions, obtained via L^2-projection of the initial data.

Let $\mathbf{U}_0(x) = (A_0(x), Q_0(x))^\top$ be the vector collecting the initial data for the cross-sectional area and the flow rate, and $\mathbf{U}_{0,h}(x)$ the L^2-projection of $\mathbf{U}_h$ onto V_h. This implies that $\mathbf{U}_{0,h}$ is chosen such that the L^2-error norm between $\mathbf{U}_0$ and $\mathbf{U}_{0,h}$ is minimum, i.e.:

$$\mathbf{U}_{0,h} := \arg\min_{\mathbf{v}_h\in V_h} \|\mathbf{U}_0 - \mathbf{v}_h\|^2_{L^2}\ , \tag{3.20}$$

which is equivalent to requiring:

$$\sum_{m=1}^{M} \Big(\mathbf{U}_{0,h}, \mathbf{v}_h\Big)_{\Omega_m} = \sum_{m=1}^{M} \Big(\mathbf{U}_0, \mathbf{v}_h\Big)_{\Omega_m} \qquad \forall \mathbf{v}_h \in V_h\ . \tag{3.21}$$

By introducing $\{\varphi_i\}_{i=1}^N$ as basis functions defined on all the cells (as already proposed in (3.9)), we obtain the algebraic counterpart of this variational problem:

$$\mathrm{M}_h\,\widehat{\mathbf{U}}_0 = \mathbf{f}_{0,h} \tag{3.22}$$

with M_h being the mass matrix (as expressed in (3.11)), while $(\mathbf{f}_{0,h})_i := \sum_{m=1}^{M} (\mathbf{U}_0, \varphi_i)_{\Omega_m}$, $\forall i \in \{1, \dots, N\}$. The resulting coefficients of the Legendre polynomials for the initial

conditions can be finally computed as:

$$\widehat{\mathbf{U}}_0 = \mathrm{M}_h^{-1}\,\mathbf{f}_{0,h}\ . \tag{3.23}$$

We recall that M_h is diagonal and thus trivially invertible in the case of Legendre polynomials.

3.4. Numerical flux and Riemann problem

The objective of this section is to find a suitable expression for the numerical flux $\mathbf{F}^*$, which has been already introduced in Section 3.1.

As already mentioned in Section 3.1, given the discontinuous nature of the functions belonging to V_h (defined in (3.4)), the solution at the interface between two adjacent cells exhibits a jump discontinuity, i.e. the value of the solution lying on the left-hand side of the e-th interior node (denoted by $\mathbf{U}_L^e = (A_L^e, Q_L^e)^\top$) differs from the value lying on the right-hand side (denoted by $\mathbf{U}_R^e = (A_R^e, Q_R^e)^\top$), as illustrated in Figure 3.2. More rigorously, we consider the e-th interface between the m-th and $(m+1)$-th cells, thus defining:

$$\mathbf{U}_L^e(t) := \lim_{x\to x_R^m} \mathbf{U}_h(x,t)\ , \qquad \mathbf{U}_R^e(t) := \lim_{x\to x_L^{m+1}} \mathbf{U}_h(x,t)\ . \tag{3.24}$$

Then, we choose a sufficiently small $\varepsilon > 0$, such that $\mathbf{U}_h(x,t)$ can be approximated as constant in the neighborhoods $(x^e-\varepsilon, x^e)$ and $(x^e, x^e+\varepsilon)$ at a fixed time $t = t^n$, yielding:

$$\begin{aligned} \mathbf{U}_h(x,t^n) &\approx \mathbf{U}_L^e(t^n) = \text{const} \qquad && \forall x \in (x^e-\varepsilon, x^e)\ , \\ \mathbf{U}_h(x,t^n) &\approx \mathbf{U}_R^e(t^n) = \text{const} && \forall x \in (x^e, x^e+\varepsilon)\ . \end{aligned} \tag{3.25}$$

Similarly, we select a sufficiently small $\delta > 0$, such that $\mathbf{U}_h(x^e \pm \varepsilon, t)$ can be approximated as constant in the neighborhood $(t^n, t^n+\delta)$, namely:

$$\begin{aligned} \mathbf{U}_h(x^e-\varepsilon,t) &\approx \mathbf{U}_L^e(t^n) = \text{const} \qquad && \forall t \in (t^n, t^n+\delta)\ , \\ \mathbf{U}_h(x^e+\varepsilon,t) &\approx \mathbf{U}_R^e(t^n) = \text{const} && \forall t \in (t^n, t^n+\delta)\ . \end{aligned} \tag{3.26}$$

With a little abuse of notation, we denote $\mathbf{U}_L^e \equiv \mathbf{U}_L^e(t^n)$ and $\mathbf{U}_R^e \equiv \mathbf{U}_R^e(t^n)$ hereafter, as depicted in Figure 3.2.

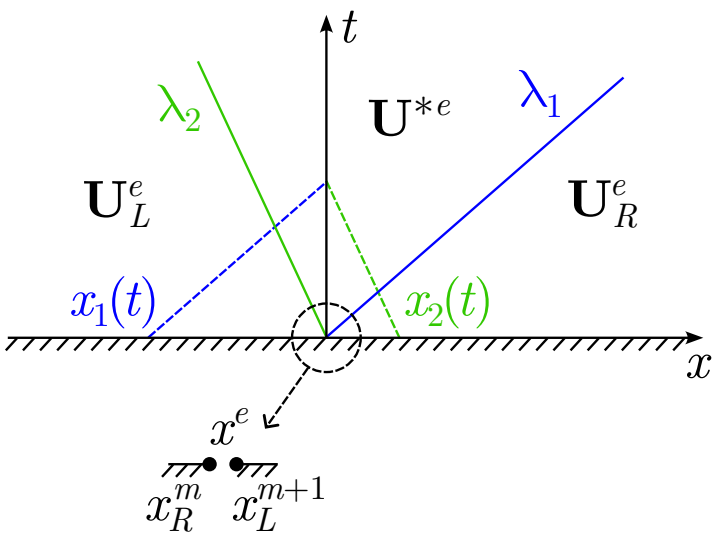


Figure 3.2: Graphical representation of the Riemann problem. The domain is identified by the dashed oblique line. $\mathbf{U}_L^e$ and $\mathbf{U}_R^e$ represent the left and right states. $\mathbf{U}^{*e}$ is the solution within the intermediate (or *star*) region. As already defined in (2.17), λ_1 and λ_2 are the eigenvalues of the Jacobian matrix $\mathrm{H}(\mathbf{U})$ associated with the right- and left-running waves, respectively. $x_1(t)$ and $x_2(t)$ are the characteristic lines along which W_1 and W_2 propagate unchanged.

This leads to an initial-boundary value problem to be solved at each e-th interior node, whose initial data are piecewise constant, with the only jump discontinuity appearing at $x = x^e$, formally:

$$\begin{cases} \dfrac{\partial \mathbf{U}}{\partial t} + \mathrm{H}(\mathbf{U})\dfrac{\partial \mathbf{U}}{\partial x} + \mathbf{B}(\mathbf{U}) = \mathbf{0} & \forall x \in (x^e - \varepsilon, x^e + \varepsilon),\ \forall t \in (t^n, t^n + \delta) \\ \mathbf{U}(x^e - \varepsilon, t) = \mathbf{U}_L^e & \forall t \in (t^n, t^n + \delta) \\ \mathbf{U}(x^e + \varepsilon, t) = \mathbf{U}_R^e & \forall t \in (t^n, t^n + \delta) \\ \mathbf{U}(x, t^n) = \begin{cases} \mathbf{U}_L^e & \text{if } x < x^e \\ \mathbf{U}_R^e & \text{if } x > x^e \end{cases} & \forall x \in (x^e - \varepsilon, x^e + \varepsilon) \end{cases} \tag{3.27}$$

Moreover, it is important to recall that the problem (3.27) arises at the interface between two cells and we are seeking a suitable expression for the numerical flux $\mathbf{F}^*$, appearing in the last term of (3.7). Since the reaction contribution is null at the interface between two cells (that is, $\mathbf{S}(\mathbf{U})$ does not appear in the last term of (3.7)), we drop the reaction term

from the first equation of (3.27), yielding:

$$\begin{cases} \dfrac{\partial \mathbf{U}}{\partial t} + \mathrm{H}(\mathbf{U})\dfrac{\partial \mathbf{U}}{\partial x} = \mathbf{0} & \forall x \in (x^e - \varepsilon, x^e + \varepsilon),\ \forall t \in (t^n, t^n + \delta) \\ \mathbf{U}(x^e - \varepsilon, t) = \mathbf{U}_L^e & \forall t \in (t^n, t^n + \delta) \\ \mathbf{U}(x^e + \varepsilon, t) = \mathbf{U}_R^e & \forall t \in (t^n, t^n + \delta) \\ \mathbf{U}(x, t^n) = \begin{cases} \mathbf{U}_L^e & \text{if } x < x^e \\ \mathbf{U}_R^e & \text{if } x > x^e \end{cases} & \forall x \in (x^e - \varepsilon, x^e + \varepsilon) \end{cases} \tag{3.28}$$

The hyperbolic system (3.28) is referred to as *Riemann problem* in hyperbolic problems theory and we refer to [23] for a more in-depth theoretical perspective. This problem is illustrated graphically in Figure 3.2.

Given the definition of characteristic variables $W_{1,2}$ provided in (2.21), we can diagonalize the first equation of (3.28) as follows:

$$\frac{\partial W_i}{\partial t} + \lambda_i \frac{\partial W_i}{\partial x} = 0 \qquad \forall i \in \{1, 2\} \tag{3.29}$$

with λ_i being the eigenvalues of $\mathrm{H}(\mathbf{U})$ defined in (2.17). The relation above is equivalent to (2.22), except for the reaction term. After that, it is possible to rewrite (3.29) along a suitable path $x_i(t)$ defined by $\dfrac{\mathrm{d}x_i}{\mathrm{d}t} = \lambda_i$, hence converting (3.29) into a system of ODEs:

$$\begin{cases} \left.\dfrac{\mathrm{d}W_i}{\mathrm{d}t}\right|_{x=x_i(t)} = 0 \\ \dfrac{\mathrm{d}x_i}{\mathrm{d}t} = \lambda_i \end{cases} \qquad \forall i = \{1, 2\}\ . \tag{3.30}$$

It is noteworthy that the characteristic variables $W_{1,2}$ are preserved along the lines described by $x_{1,2}(t)$, which are referred to as *characteristic lines* in hyperbolic problems theory (see Figure 3.2). Given the chosen ordering for the eigenvalues and the subcritical nature of the problem under physiological conditions, the subscript "1" always identifies right-running waves, while the subscript "2" corresponds to left-running waves.

In hyperbolic problems theory, solving the Riemann problem (3.28) entails computing the solution within the intermediate (or *star*) region, denoted by $\mathbf{U}^{*e}$ in Figure 3.2, and the corresponding numerical flux $\mathbf{F}^*$. As discussed in [23], if the characteristic variables $W_{1,2}$ are known in a closed analytical form, the Riemann problem can be solved by applying the methods of characteristics, i.e. we can obtain $\mathbf{U}^{*e}$ exactly and compute the numerical flux as: $\mathbf{F}^*(\mathbf{U}_L^e, \mathbf{U}_R^e) = \mathbf{F}(\mathbf{U}^{*e})$ (further details follow in Section 3.4.1). On

the other hand, when the characteristic variables are not available analytically, only a numerical approximation of $\mathbf{U}^{*e}$ and $\mathbf{F}^*$ can be found; we remark that, in the latter case, $\mathbf{F}^*(\mathbf{U}_L^e, \mathbf{U}_R^e) \neq \mathbf{F}(\mathbf{U}^{*e})$, since $\mathbf{U}^{*e}$ represents an approximation of the real intermediate state and cannot be used to evaluate the numerical flux $\mathbf{F}^*$ (as rigorously analyzed in Section 3.4.2).

3.4.1. Solution of the Riemann problem for a flat velocity profile

Only for a flat velocity profile (i.e. with $\alpha = 1$), $W_{1,2}$ are given analytically by the expression (2.24). By recalling that the characteristic variables $W_{1,2}$ are preserved along the corresponding characteristic lines $x_{1,2}(t)$ (as formally written in (3.30)), we proceed as follows:

- we allow W_1 to propagate unchanged along $x_1(t)$, i.e. from the left state $\mathbf{U}_L^e$ towards the star region, as depicted in Figure 3.2;
- we allow W_2 to propagate unchanged along $x_2(t)$, i.e. from the right state $\mathbf{U}_R^e$ towards the star region, as depicted in Figure 3.2.

This is known as *method of characteristics* in hyperbolic problems theory [23]. In formulas, the following non-linear system arises at each e-th interface, in the unknowns $\mathbf{U}^{*e} = (A^{*e}, Q^{*e})^\top$:

$$\begin{cases} W_1(\mathbf{U}^{*e}) = W_1(\mathbf{U}_L^e) \\ W_2(\mathbf{U}^{*e}) = W_2(\mathbf{U}_R^e) \end{cases} . \tag{3.31}$$

We can solve (3.31) by introducing the analytical expression of the characteristic variables (2.24), thus finding the following expression for the intermediate state $\mathbf{U}^{*e}$:

$$\mathbf{U}^{*e} = \begin{pmatrix} A^{*e} \\ Q^{*e} \end{pmatrix} = \begin{pmatrix} \left(\frac{2\rho A_{\text{ref}}}{\beta}\right)^2 \left(\frac{W_1 - W_2}{8}\right)^4 \\ A^{*e}\,\frac{W_1 + W_2}{2} \end{pmatrix} . \tag{3.32}$$

Thus, to address the issue of having a jump discontinuity at the e-th interior node, the corresponding numerical flux can be defined as:

$$\mathbf{F}^*(\mathbf{U}_L^e, \mathbf{U}_R^e) = \mathbf{F}(\mathbf{U}^{*e}) , \tag{3.33}$$

where the expression for the generic flux term $\mathbf{F}(\mathbf{U})$ has been already provided in (2.14). This method is successfully devised and employed in [2, 11].

3.4.2. Solution of the Riemann problem for a non-flat velocity profile

For a generic velocity profile with $\alpha \neq 1$ (with α being the momentum flux correction factor (2.4)), a closed analytical form for the characteristic variables $W_{1,2}$ is not available (as discussed in Appendix A). This limitation forces us to use approximate Riemann solvers, i.e. a class of numerical methods solving for $\mathbf{U}^{*e}$ and $\mathbf{F}^{*}$ only approximately.

Among the approximate Riemann solvers available in literature, we choose Roe's linearization, as described in [12, 24]. The objective of this method is to find a suitable linearization of the system (3.28), denoted by:

$$\frac{\partial \mathbf{U}}{\partial t} + \widetilde{\mathrm{H}}\,\frac{\partial \mathbf{U}}{\partial x} = \mathbf{0}\ , \tag{3.34}$$

where $\widetilde{\mathrm{H}} = \widetilde{\mathrm{H}}(\mathbf{U}_L^e, \mathbf{U}_R^e)$ is the linearized Jacobian matrix. It is worth noting that $\widetilde{\mathrm{H}}$ depends on the left state $\mathbf{U}_L^e$ and right state $\mathbf{U}_R^e$ only (which represent the constant boundary conditions for the Riemann problem, as explained in Section 3.4), thus $\widetilde{\mathrm{H}}$ is a constant-coefficient matrix. As proposed in the original work of Roe [12], we choose $\widetilde{\mathrm{H}}$ such that it satisfies the following properties:

1. *consistency:* $\widetilde{\mathrm{H}}(\mathbf{U}_L^e, \mathbf{U}_R^e) \to \mathrm{H}(\overline{\mathbf{U}}) \quad \text{as} \quad \mathbf{U}_{L,R}^e \to \overline{\mathbf{U}}$;
2. *conservation:* $\widetilde{\mathrm{H}}\ (\mathbf{U}_R^e - \mathbf{U}_L^e) = \mathbf{F}(\mathbf{U}_R^e) - \mathbf{F}(\mathbf{U}_L^e)$;
3. *local hyperbolicity:* $\widetilde{\mathrm{H}}$ is diagonalizable with real eigenvalues.

Then, we consider the method applied in [12, 24] to the Euler equations for the compressible fluid dynamics and extend it to the 1D haemodynamics equations (2.2), thus finding the following expression for $\widetilde{\mathrm{H}}$:

$$\widetilde{\mathrm{H}} = \mathrm{H}(\widetilde{A}, \widetilde{Q}, \widetilde{c}) = \begin{bmatrix} 0 & 1 \\ \widetilde{c}^2 - \alpha \left(\dfrac{\widetilde{Q}}{\widetilde{A}}\right)^2 & 2\alpha \dfrac{\widetilde{Q}}{\widetilde{A}} \end{bmatrix}\ , \tag{3.35}$$

where $\widetilde{A}$, $\widetilde{Q}$, $\widetilde{c}$ are referred to as *Roe's averages*:

$$\begin{aligned}
\widetilde{A} &= \left(\frac{\sqrt{A_L^e} + \sqrt{A_R^e}}{2} \right)^2 , \\
\widetilde{Q} &= \frac{\sqrt{\widetilde{A}}}{2} \left(\frac{Q_L^e}{\sqrt{A_L^e}} + \frac{Q_R^e}{\sqrt{A_R^e}} \right) , \\
\widetilde{c} &= \sqrt{\frac{\beta}{2\rho A_{\text{ref}}} \frac{A_L^e + \sqrt{A_L^e A_R^e} + A_R^e}{3\sqrt{\widetilde{A}}}} \neq c(\widetilde{A}) .
\end{aligned} \tag{3.36}$$

The derivation of Roe's averages consists in enforcing the conservation property by a suitable variable change, while the consistency and the local hyperbolicity follow as a consequence. We refer to Appendix B for the complete derivation, carried out specifically for this work.

After that, we solve the Riemann problem for the linearized system (3.34), in order to find a suitable approximation of the intermediate state $\mathbf{U}^{*e}$ and the numerical flux $\mathbf{F}^*$. For this purpose, we apply the definition of characteristic variables (2.21) to the linearized case (3.34), thus finding the following approximation of the characteristic variables $\widetilde{W}_{1,2}$:

$$\widetilde{W}_{1,2} = \widetilde{\mathbf{l}}_{1,2} \cdot \mathbf{U} . \tag{3.37}$$

By employing the method of characteristic for hyperbolic systems, we obtain the following approximation for the intermediate state $\mathbf{U}^{*e}$:

$$\mathbf{U}^{*e} = \mathbf{U}_L^e + \left[\widetilde{\mathbf{l}}_2 \cdot (\mathbf{U}_R^e - \mathbf{U}_L^e) \right] \widetilde{\mathbf{r}}_2 . \tag{3.38}$$

The intermediate steps can be found in [24] and are reported in detail in Appendix C.

The corresponding numerical flux $\mathbf{F}^*$ reads:

$$\mathbf{F}^*(\mathbf{U}_L^e, \mathbf{U}_R^e) = \mathbf{F}(\mathbf{U}_L^e) + \widetilde{\lambda}_2 \left[\widetilde{\mathbf{l}}_2 \cdot (\mathbf{U}_R^e - \mathbf{U}_L^e) \right] \widetilde{\mathbf{r}}_2 , \tag{3.39}$$

where $\widetilde{\lambda}_2$ represents the second eigenvalue of $\widetilde{\mathrm{H}}$ and is equivalent to (2.17) evaluated at Roe's averages. Again, the intermediate steps can be found in [24] and are reported in detail in Appendix C. We remark that, in this case, $\mathbf{F}^*(\mathbf{U}_L^e, \mathbf{U}_R^e) \neq \mathbf{F}(\mathbf{U}^{*e})$ since the intermediate state (3.38) is just an approximation of the real intermediate state (due to Roe's linearization), so the expression for $\mathbf{F}^*$ must be derived by considering the integral

formulation of the conservative form (2.13), as shown in Appendix C.

Hereafter, for the sake of conciseness, we will denote by $\mathbf{R}(\mathbf{U}_L^e, \mathbf{U}_R^e)$ the solution of the Riemann problem, which is formally given by:

$$\mathbf{U}^{*e} = \mathbf{R}(\mathbf{U}_L^e, \mathbf{U}_R^e) = \begin{cases} \mathbf{U}_{\text{exact}}^{*e} & \text{if } \alpha = 1 \text{ (i.e. for flat velocity profiles)} \\ \mathbf{U}_{\text{Roe}}^{*e} & \text{if } \alpha \neq 1 \text{ (i.e. for non-flat velocity profiles)} \end{cases} , \tag{3.40}$$

where $\mathbf{U}_{\text{exact}}^{*e}$ is given by the exact solution to the Riemann problem (3.32), while $\mathbf{U}_{\text{Roe}}^{*e}$ is obtained through Roe's linearization (3.38). We will refer to (3.40) as *direct Riemann problem* hereafter.

3.5. Boundary conditions and inverse Riemann problem

In this section, we extend the Riemann problem (3.28) to boundary nodes in order to deal with boundary conditions. Given the discontinuous nature of the Riemann problem, we have to address the issue of having a jump discontinuity in the solution also at boundary nodes. In this work, we choose to prescribe the boundary conditions directly on the intermediate state $\mathbf{U}^{*e}$, which represents the value of the solution at the e-th boundary, i.e. $\lim_{t \to t^n} \mathbf{U}_h(x^e, t) = \mathbf{U}^{*e}$ (as illustrated in Figure 3.3). However, this is not the only technique to enforce boundary conditions in the DG framework and other methods are reported in [25].

Let $\mathbf{U}^{-e}(t)$ be the *internal state*, representing the state known by interpolation of the latest solution at the e-th boundary node, formally:

$$\mathbf{U}^{-e}(t) := \lim_{x \to x^{-e}} \mathbf{U}_h(x, t) , \tag{3.41}$$

where x^{-e} represents the interior side of the node x^e, as illustrated in Figure 3.3. Let $\mathbf{U}^{+e}(t)$ be the *external state*, which is a virtual state introduced to guarantee that the boundary conditions are properly prescribed on $\mathbf{U}^{*e}$, formally:

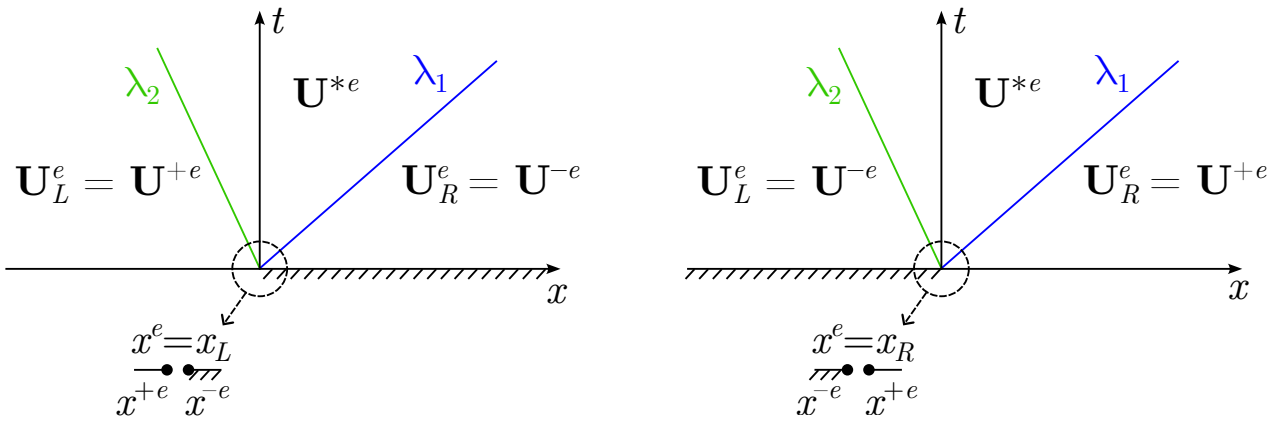
$$\mathbf{U}^{+e}(t) := \lim_{x \to x^{+e}} \mathbf{U}_h(x, t) , \tag{3.42}$$

where x^{+e} represents the exterior side of the node x^e, as illustrated in Figure 3.3. Then, we apply the same procedure shown in (3.25)–(3.26) and, with a little abuse of notation, we denote $\mathbf{U}^{-e} \equiv \mathbf{U}^{-e}(t^n)$ and $\mathbf{U}^{+e} \equiv \mathbf{U}^{+e}(t^n)$ hereafter.

Therefore, it can be noticed that the internal and external states ($\mathbf{U}^{-e}$ and $\mathbf{U}^{+e}$) are linked to the left and right states ($\mathbf{U}_L^e$ and $\mathbf{U}_R^e$) as follows:

1. $\mathbf{U}^{-e} = \mathbf{U}_R^e$ when the e-th boundary node represents the left endpoint of the 1D domain, i.e. $x^e = x_L$ (as shown in Figure 3.3a);
2. $\mathbf{U}^{-e} = \mathbf{U}_L^e$ when the e-th boundary node represents the right endpoint of the 1D domain, i.e. $x^e = x_R$ (as shown in Figure 3.3b).

The converse is true for $\mathbf{U}^{+e}$, as shown in Figure 3.3.



(a) e-th node is located at the left endpoint of the 1D domain, i.e. $x^e = x_L$.

(b) e-th node is located at the right endpoint of the 1D domain, i.e. $x^e = x_R$.

Figure 3.3: Graphical representation of the inverse Riemann problem. The domain is identified by the dashed oblique line. $\mathbf{U}_L^e$ and $\mathbf{U}_R^e$ represent the left and right states. $\mathbf{U}^{-e}$ and $\mathbf{U}^{+e}$ represent the internal and external states. $\mathbf{U}^{*e}$ is the solution within the intermediate (or *star*) region. As already defined in (2.17), λ_1 and λ_2 are the eigenvalues of the Jacobian matrix $\mathrm{H}(\mathbf{U})$ associated with the right- and left-running waves, respectively.

It is worth noting that, given the hyperbolic nature of (2.2), we can enforce only one scalar equation per boundary node, since only one characteristic enters the computational domain under physiological (subcritical) conditions. We will denote this boundary condition enforced at the e-th boundary node by a suitable residual function, namely $g(A^{*e}, Q^{*e}) = 0$. Since the external state $\mathbf{U}^{+e}$ is not known a priori, we need to solve the following non-linear system at the e-th boundary node, in the four unknowns $(A^{*e}, Q^{*e}, A^{+e}, Q^{+e})^\top$:

$$\begin{cases} g(A^{*e}, Q^{*e}) = 0 \\ \mathbf{U}^{*e} = \begin{pmatrix} A^{*e} \\ Q^{*e} \end{pmatrix} = \mathbf{R}(\mathbf{U}_L^e, \mathbf{U}_R^e) \end{cases} \tag{3.43}$$

where the direct Riemann problem $\mathbf{R}(\mathbf{U}_L^e, \mathbf{U}_R^e)$ is given by the expression (3.40). It is important to notice that we still miss one equation to close the system (3.43) (since only 3 equations are provided for 4 unknowns). By recalling that the external state $\mathbf{U}^{+e}$ is just virtual and does not have any physical meaning, we choose arbitrarily to enforce the continuity of the cross-sectional area between the internal and the external states, namely $A^{+e} = A^{-e}$ (or, equivalently, $A_L^e = A_R^e$), thus yielding:

$$\begin{cases} g(A^{*e}, Q^{*e}) = 0 \\ \mathbf{U}^{*e} = \begin{pmatrix} A^{*e} \\ Q^{*e} \end{pmatrix} = \mathbf{R}(\mathbf{U}_L^e, \mathbf{U}_R^e) \\ A_L^e = A_R^e \end{cases} \tag{3.44}$$

We will refer to (3.44) as *inverse Riemann problem* hereafter. From a numerical standpoint, given the non-linearity of (3.44), Newton's method is employed to solve this system iteratively. A similar approach is presented in the appendix of [11], but, instead of this general framework, an analytical expression for $g(A^{*e}, Q^{*e})$ is derived for each boundary condition type.

A complete list of the implemented boundary conditions, together with the corresponding residual function $g(A^{*e}, Q^{*e})$, is reported below. We highlight that the boundary residuals are all dimensionless in order to simplify the tolerance setup of Newton's solver for the non-linear system (3.44).

3.5.1. Inlet boundary condition

At the inlet of the blood vessel, we usually prescribe a known value of a certain quantity over time, which can be either the cross-sectional area (i.e. $A^{*e} = A_{\text{in}}(t)$), the inflow (i.e. $Q^{*e} = Q_{\text{in}}(t)$) or the pressure (i.e. $P^{*e} = P_{\text{in}}(t)$). These boundary conditions result in the following residual function for either the cross-sectional area:

$$g(A^{*e}, Q^{*e}) = \frac{A^{*e} - A_{\text{in}}(t)}{A_{\text{ref}}} = 0 \,, \tag{3.45}$$

or the flow rate:

$$g(A^{*e}, Q^{*e}) = \frac{Q^{*e} - Q_{\text{in}}(t)}{Q_{\text{ref}}} = 0 \,, \tag{3.46}$$

or the pressure:

$$g(A^{*e}, Q^{*e}) = \frac{P^{*e} - P_{\text{in}}(t)}{P_{\text{tot,ref}}} = 0 \,, \tag{3.47}$$

where A_{ref}, Q_{ref} and $P_{\text{tot,ref}}$ are arbitrary reference values, needed to make the expressions above dimensionless. We remark that the relation $P^{*e} = P(A^{*e})$ is provided by the tube law (2.3), while we employed the total pressure $P_{\text{tot,ref}}$ (2.29) as reference value for (3.47), since the reference pressure P_{ref} appearing in the tube law (2.3) is usually set to 0 Pa.

3.5.2. Non-reflecting boundary condition

This boundary condition requires that the outgoing waves must leave the computational domain without any wave reflection. When no reaction term appears (i.e. $\mathbf{S}(\mathbf{U}) = \mathbf{B}(\mathbf{U}) = \mathbf{0}$, as given by the definitions (2.14)–(2.11)), this is equivalent to requiring that the incoming characteristic variable is constant over time, namely:

$$\left.\frac{\partial W_{\text{in}}}{\partial t}\right|_{x=x^e} = 0 \,, \tag{3.48}$$

where the choice of the incoming characteristic variable depends on whether x^e represents the left endpoint x_L or the right endpoint x_R of the 1D domain, formally:

$$W_{\text{in}} = \begin{cases} W_1 & \text{if } x^e = x_L \\ W_2 & \text{if } x^e = x_R \end{cases} \,. \tag{3.49}$$

On the other hand, when $\mathbf{S}(\mathbf{U}) \neq \mathbf{0}$, we should also introduce the variation of the characteristic variables due to reaction contributions [1], yielding the following condition on the incoming characteristic:

$$\mathbf{l}_{\text{in}}(\mathbf{U}^{*e}) \cdot \left[\frac{\partial \mathbf{U}^{*e}}{\partial t} + \mathbf{S}(\mathbf{U}^{*e})\right] = 0 \tag{3.50}$$

with $\mathbf{l}_{\text{in}}$ being the left eigenvector associated with the incoming characteristic, whose definition is provided in (2.19). By employing a forward Euler time discretization, we can find the following expression for the dimensionless boundary residual:

$$g((A^{*e})^{n+1}, (Q^{*e})^{n+1}) = \frac{\mathbf{l}_{\text{in}}(\mathbf{U}^{*e})^n}{W_{\text{in,ref}}} \cdot \left[(\mathbf{U}^{*e})^{n+1} - (\mathbf{U}^{*e})^n + \Delta t\ \mathbf{S}(\mathbf{U}^{*e})^n\right] = 0 \,, \tag{3.51}$$

where $W_{\text{in,ref}}$ is an arbitrary reference value for the incoming characteristic variable, needed to make the residual (3.51) dimensionless. One possible choice of $W_{\text{in,ref}}$, consistent with the analytical expression (2.24) for flat velocity profiles, is given by:

$$W_{\text{in,ref}} := \frac{Q_{\text{ref}}}{A_{\text{ref}}} \pm 4c(A_{\text{ref}}) \,, \tag{3.52}$$

where A_{ref} and Q_{ref} are arbitrary reference values for the cross-sectional area and the flow rate, respectively.

3.5.3. 3-element Windkessel boundary condition

As discussed in [19], the peripheral circulation is often captured by means of a 3-element Windkessel model (Figure 3.4), composed of:

- a proximal resistance R_1, representing the viscous effects within the proximal compartment;
- a compliance C, representing the mass storage due to the vessels deformation;
- a distal resistance R_2, representing the viscous effects within the distal compartment.

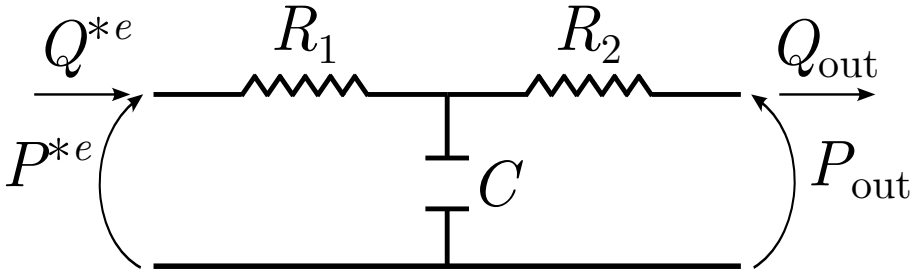


Figure 3.4: 3-element Windkessel lumped-parameter model. P^{*e} and Q^{*e} represent the input pressure and flow rate. P_{out} and Q_{out} represent the output pressure and flow rate. R_1, C and R_2 are the proximal resistance, the vessels compliance and the distal resistance, respectively.

The relation between the input flow rate Q^{*e} and the input pressure P^{*e} is provided by the following ODE:

$$R_2 C \frac{\mathrm{d}P^{*e}}{\mathrm{d}t} + P^{*e} - P_{\text{out}} = R_1 R_2 C \frac{\mathrm{d}Q^{*e}}{\mathrm{d}t} + (R_1 + R_2)\, Q^{*e}\,, \tag{3.53}$$

where P_{out} is the outflow pressure, i.e. the pressure at which flow to the micro-circulation ceases (usually set to 0 Pa), while the relation $P^{*e} = P(A^{*e})$ is provided by the tube law (2.3). The coupling between this 0D model and the 1D blood flow model is achieved by solving the inverse Riemann problem (3.44), with the following boundary residual function, obtained by discretizing (3.53) in time with a backward Euler approach:

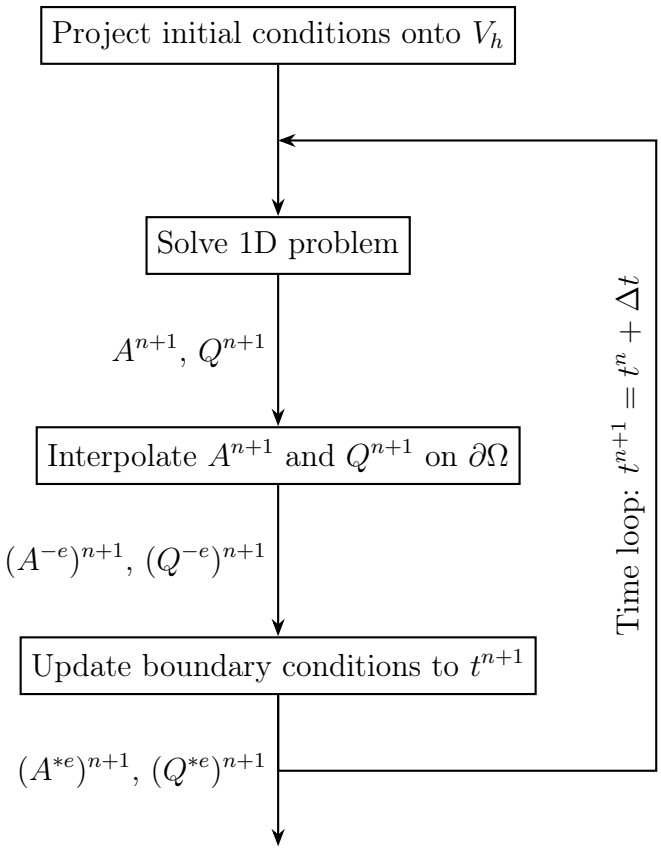
$$\begin{aligned} g((A^{*e})^{n+1}, (Q^{*e})^{n+1}) = \frac{1}{Q_{\text{ref}}} \Bigg\{ & \frac{(P^{*e})^{n+1} - (P^{*e})^{n}}{R_1} - \Big[(Q^{*e})^{n+1} - (Q^{*e})^{n}\Big] + \\ & + \frac{\Delta t}{R_1 R_2 C} \Big[(P^{*e})^{n+1} - P_{\text{out}} - (R_1 + R_2)\,(Q^{*e})^{n+1}\Big] \Bigg\} = 0\,, \end{aligned} \tag{3.54}$$

where Q_{ref} is the usual arbitrary flow rate needed to make the expression dimensionless.

3.6. Sketch of the algorithm

The sketch of the algorithm to simulate one isolated vessel is illustrated in Algorithm 3.1. More specifically, at each time step, the solution of the 1D haemodynamics model is evolved to the current time step t^{n+1} by assembling the right-hand side and computing $\mathbf{U}^{n+1}$, according to the AB update rule (3.18). Then, the internal state $(\mathbf{U}^{-e})^{n+1}$ can be obtained by interpolating the new solution $\mathbf{U}^{n+1}$ on the boundary $\partial\Omega$. Finally, we employ the internal state $(\mathbf{U}^{-e})^{n+1}$ to solve the inverse Riemann problem (3.44) numerically by means of Newton's method, thus obtaining the new intermediate state $(\mathbf{U}^{*e})^{n+1}$. In the next time iteration, the intermediate state $(\mathbf{U}^{*e})^{n+1}$ will be used to assemble the new right-hand side, which in turn will be fed into the AB update rule (3.18).

Algorithm 3.1 1D haemodynamics model for one isolated blood vessel.



4 | Extension to blood vessel networks

A complex vascular network may include vessels with different dimensions or different material properties, or it may also exhibit some bifurcation points, i.e. points where one single vessel divides into two separate vessels. For this purpose, we need to extend the model for one isolated vessel (discussed in Chapter 3) to handle discontinuous material points and junctions. This extension is achieved by formulating suitable inverse Riemann problems at the interface between different vessels, which interact through physically motivated matching conditions (i.e. the continuity of flow rate and total pressure, which stem from the mass and momentum conservation, respectively).

4.1. Discontinuous material properties

In this section, we treat blood vessels with different dimensions and material properties, which is particularly relevant when studying the physical wave reflections produced by stents. We remark that the expression for the direct Riemann problem (3.40) holds only when the material properties are continuous along the two adjacent cells. Hence, we should devise a technique capable of handling discontinuous material properties consistently.

For this purpose, we proceed as proposed in [2], i.e. we consider each compartment with constant material properties separately and link that compartment with its adjacent compartments by means of suitable matching conditions. As done for the boundary conditions in Section 3.5, also the matching conditions should be enforced on the intermediate state $\mathbf{U}^{*e}$ of each compartment. A graphical representation is offered in Figure 4.1.

Physical considerations (linked to the mass and momentum conservation) lead to choose the continuity of the flow rate and the total pressure as matching conditions; by supposing to write the matching conditions between the compartments “1” and “2”, the resulting

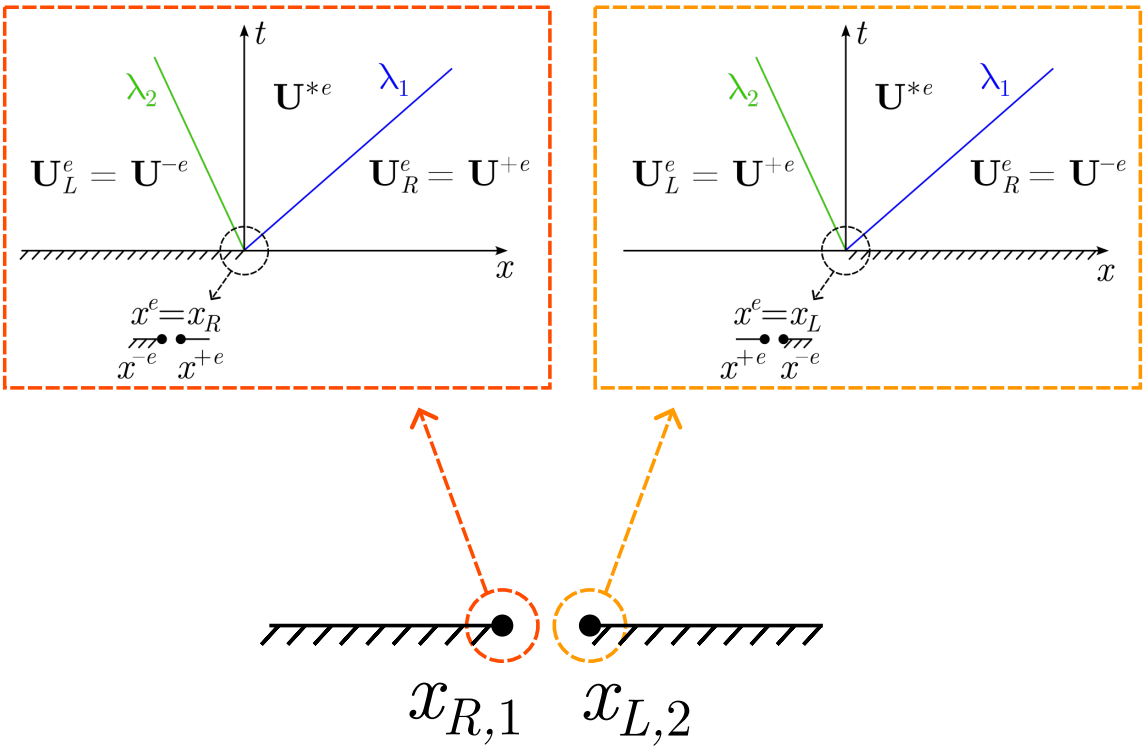


Figure 4.1: Graphical representation of the inverse Riemann problem at discontinuous material points. The domain is identified by the dashed oblique line. Each compartment with constant material properties is treated separately and the matching conditions are enforced on the intermediate state $\mathbf{U}^{*e}$ of the each compartment.

matching conditions read:

$$Q_1^{*e} = Q_2^{*e} \,, \qquad P_{\text{tot},1}^{*e} = P_{\text{tot},2}^{*e} \,, \tag{4.1}$$

where the expression for $P_{\text{tot}}^{*e} = P_{\text{tot}}(A^{*e}, Q^{*e})$ is already provided in (2.29). We finally highlight that this approach is equivalent to enforcing the continuity of the numerical fluxes at the interface between two compartments for a flat velocity profile (i.e. when $\alpha = 1$), since the flux vector is equal to $\mathbf{F}(A, u) = (Q, P_{\text{tot}})^\top$ in the $(A, u)^\top$ formulation (2.28). This also serves as a mathematical justification for the choice of (4.1) as matching conditions.

By employing a similar framework to the one introduced in (3.44) to handle the boundary conditions, we define the following inverse Riemann problem for discontinuous material

points:

$$\begin{cases} \mathbf{g}\left(\mathbf{U}_1^{*e}, \mathbf{U}_2^{*e}\right) = \mathbf{0} \\ \\ \mathbf{U}_1^{*e} = \mathbf{R}(\mathbf{U}_{L,1}^{e}, \mathbf{U}_{R,1}^{e}) \\ \mathbf{U}_2^{*e} = \mathbf{R}(\mathbf{U}_{L,2}^{e}, \mathbf{U}_{R,2}^{e}) \\ \\ A_{L,1}^{e} = A_{R,1}^{e} \\ A_{L,2}^{e} = A_{R,2}^{e} \end{cases} \tag{4.2}$$

where the unknowns are composed of the intermediate state and the external state for each compartment, formally $(\mathbf{U}_1^{*e}, \mathbf{U}_1^{+e}, \mathbf{U}_2^{*e}, \mathbf{U}_2^{+e})$, for a total of 8 unknowns per discontinuous material point. Like (3.44), also the system (4.2) is non-linear, so Newton's method is employed to solve it numerically. The term $\mathbf{R}(\mathbf{U}_L^e, \mathbf{U}_R^e)$ still represents the direct Riemann problem as provided in (3.40). We finally define the residual function starting from the matching conditions (4.1):

$$\mathbf{g}\left(\mathbf{U}_1^{*e}, \mathbf{U}_2^{*e}\right) = \begin{pmatrix} \dfrac{Q_1^{*e} - Q_2^{*e}}{\overline{Q}_{\text{ref}}} \\ \dfrac{P_{\text{tot},1}^{*e} - P_{\text{tot},2}^{*e}}{\overline{P}_{\text{tot,ref}}} \end{pmatrix} = \mathbf{0} \ , \tag{4.3}$$

where $\overline{Q}_{\text{ref}}$ and $\overline{P}_{\text{tot,ref}}$ are arbitrary values to make the residual dimensionless and thus to simplify the tolerance setup for Newton's solver. These quantities are chosen as the mean of the reference values in each compartment, namely:

$$\overline{Q}_{\text{ref}} = \frac{Q_{\text{ref},1} + Q_{\text{ref},2}}{2} \ , \qquad \overline{P}_{\text{tot,ref}} = \frac{P_{\text{tot,ref},1} + P_{\text{tot,ref},2}}{2} \ . \tag{4.4}$$

It is worth mentioning that other approaches exist to treat the discontinuous material properties, e.g. an augmented formulation is devised and studied numerically in [26, 27]. This procedure would allow to handle varying and discontinuous material properties, without any addition of matching conditions, but the resulting mathematical formulation is more complex and it is well established with finite volume schemes only. These are the reasons why we opted for the matching conditions (4.1), which are also more common in literature (e.g. we refer to [2, 9, 10]).

4.2. Junctions: bifurcations and confluences

The numerical technique to handle bifurcation points follows an approach analogous to the one employed for discontinuous material points in Section 4.1, as described in [3, 9, 19].

Moreover, the following procedure can be readily extended to confluence points, i.e. points where two blood vessels converge into one larger vessel, as it is common for the venous circulation.

In order to deal with bifurcation points, we should treat each blood vessel separately and then enforce the matching conditions on the intermediate state $\mathbf{U}^{*e}$, as already explained in Section 4.1 and illustrated graphically in Figure 4.2. If we suppose to analyze a bifurcation where the blood vessel "1" splits into the vessels "2" and "3", the matching conditions will prescribe the continuity of flow rate and total pressure as follows:

$$Q_1^{*e} = Q_2^{*e} + Q_3^{*e} \,, \qquad P_{\text{tot},1}^{*e} = P_{\text{tot},2}^{*e} = P_{\text{tot},3}^{*e} \,, \tag{4.5}$$

where the total pressure $P_{\text{tot}}^{*e} = P_{\text{tot}}(A^{*e}, Q^{*e})$ is expressed by (2.29).

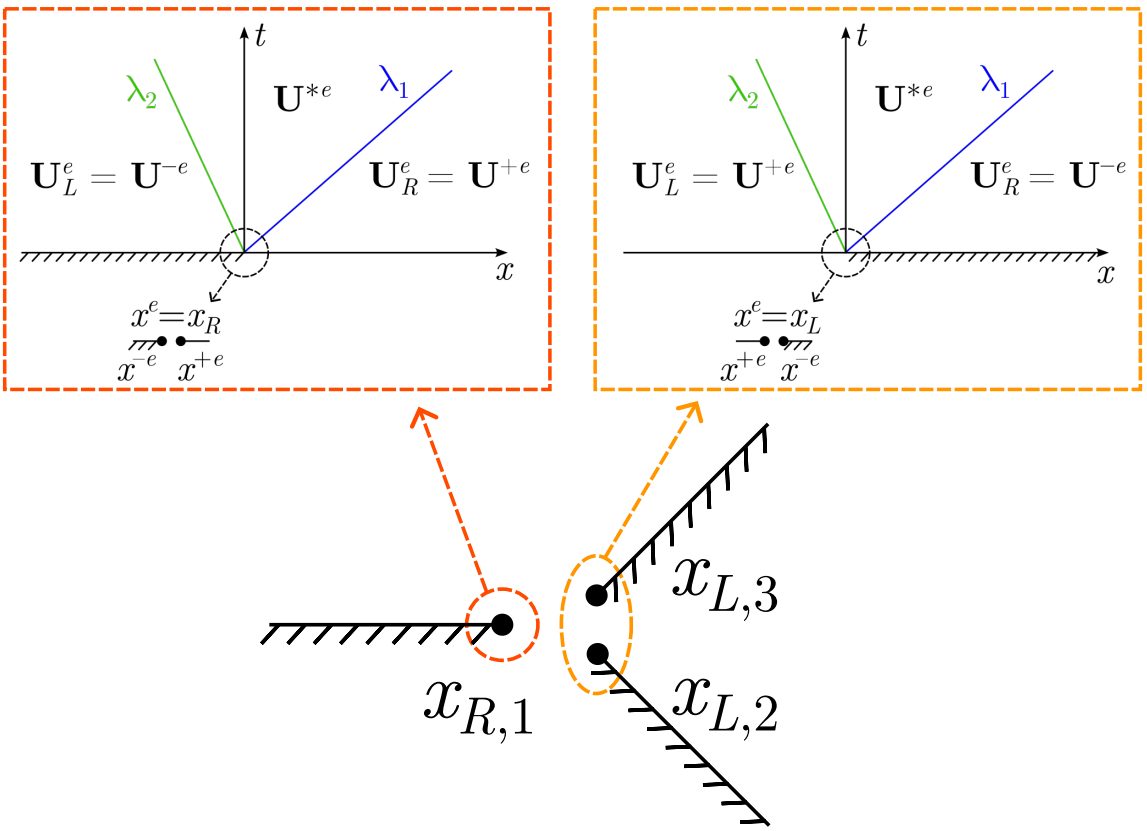


Figure 4.2: Graphical representation of the inverse Riemann problem at bifurcation points. The domain is identified by the dashed oblique line. Each blood vessel is treated separately and the matching conditions are enforced on the intermediate state $\mathbf{U}^{*e}$ of the each vessel.

Since these matching conditions should be enforced on the intermediate state $\mathbf{U}^{*e}$ of each blood vessel, we can extend the inverse Riemann problem (4.2) to a bifurcation point,

yielding:

$$\begin{cases} \mathbf{g}\left(\mathbf{U}_1^{*e}, \mathbf{U}_2^{*e}, \mathbf{U}_3^{*e}\right) = \mathbf{0} \\ \\ \mathbf{U}_1^{*e} = \mathbf{R}(\mathbf{U}_{L,1}^e, \mathbf{U}_{R,1}^e) \\ \mathbf{U}_2^{*e} = \mathbf{R}(\mathbf{U}_{L,2}^e, \mathbf{U}_{R,2}^e) \\ \mathbf{U}_3^{*e} = \mathbf{R}(\mathbf{U}_{L,3}^e, \mathbf{U}_{R,3}^e) \\ \\ A_{L,1}^e = A_{R,1}^e \\ A_{L,2}^e = A_{R,2}^e \\ A_{L,3}^e = A_{R,3}^e \end{cases} \tag{4.6}$$

where, in analogy to (4.3), the residual function $\mathbf{g}$ is used to enforce the matching conditions (4.5), formally:

$$\mathbf{g}\left(\mathbf{U}_1^{*e}, \mathbf{U}_2^{*e}, \mathbf{U}_3^{*e}\right) = \begin{pmatrix} \dfrac{Q_1^{*e} - Q_2^{*e} - Q_3^{*e}}{\overline{Q}_{\text{ref}}} \\ \\ \dfrac{P_{\text{tot},1}^{*e} - P_{\text{tot},2}^{*e}}{\overline{P}_{\text{tot,ref}}} \\ \\ \dfrac{P_{\text{tot},1}^{*e} - P_{\text{tot},3}^{*e}}{\overline{P}_{\text{tot,ref}}} \end{pmatrix} = \mathbf{0} \, . \tag{4.7}$$

Again, $\overline{Q}_{\text{ref}}$ and $\overline{P}_{\text{tot,ref}}$ are arbitrary reference values needed to make the residual dimensionless and are chosen as the mean of the reference values in each vessel, namely:

$$\begin{aligned} \overline{Q}_{\text{ref}} &= \frac{Q_{\text{ref},1} + Q_{\text{ref},2} + Q_{\text{ref},3}}{3} \, , \\ \overline{P}_{\text{tot,ref}} &= \frac{P_{\text{tot,ref},1} + P_{\text{tot,ref},2} + P_{\text{tot,ref},3}}{3} \, . \end{aligned} \tag{4.8}$$

We finally remark that this procedure can be extended to confluences by choosing properly the signs of the flow rate appearing in the matching conditions; that is to say, the first equation of (4.5) should be replaced with:

$$Q_1^{*e} + Q_2^{*e} = Q_3^{*e} \, . \tag{4.9}$$

The definition of the residual function $\mathbf{g}$ follows as a consequence and is omitted for the sake of brevity. The inverse Riemann problem arising at confluence points is illustrated in Figure 4.3.

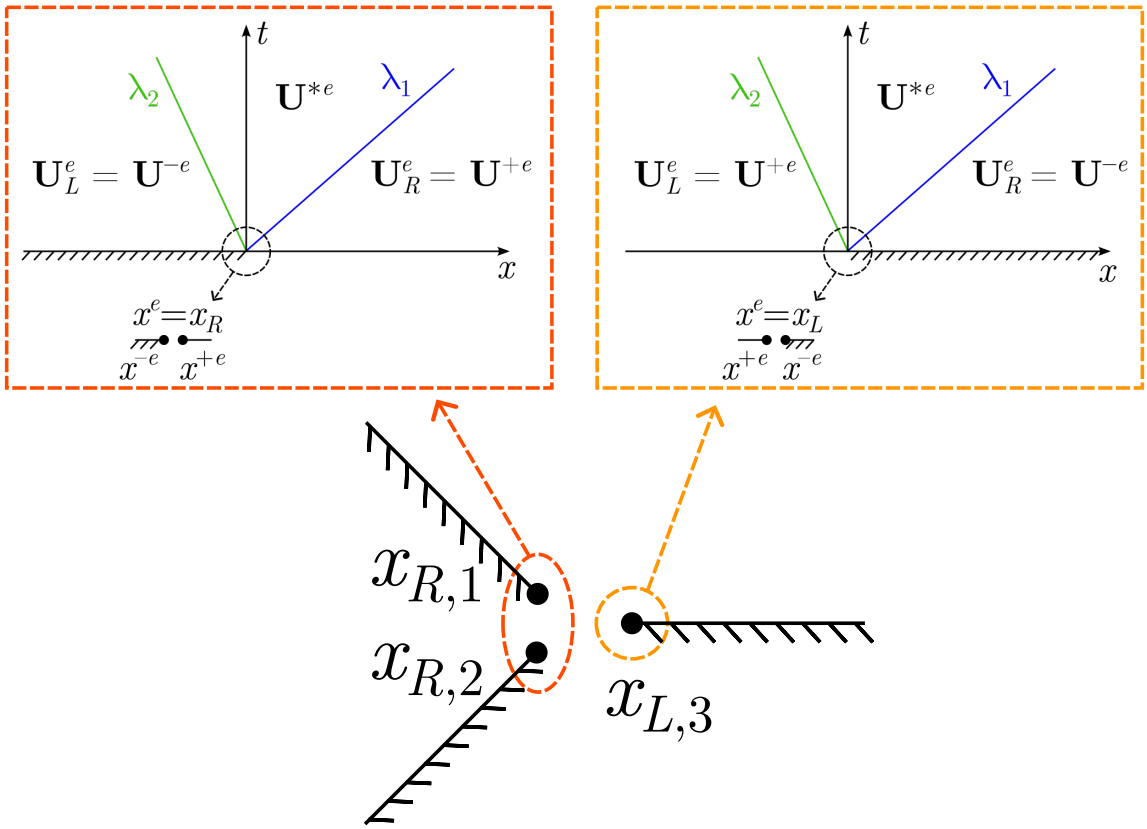


Figure 4.3: Graphical representation of the inverse Riemann problem at confluence points. The domain is identified by the dashed oblique line. Each blood vessel is treated separately and the matching conditions are enforced on the intermediate state $\mathbf{U}^{*e}$ of the each vessel.

5 | 3D-1D coupling

The 1D vascular flow model presented in this work is suitable to simulate the blood flow in vessels, however it would be inaccurate in describing the portions of the cardiovascular system where the physical phenomena are inherently three-dimensional, in particular the heart and its adjacent regions. On the other hand, 3D FSI models are computationally more expensive and the introduction of 1D models is meant to reduce the computational burden. These are the reasons why it is interesting to devise a numerical technique to couple the 3D FSI problem and the 1D haemodynamics model, as discussed in [19, 28, 29, 30]. More specifically, in this chapter, we consider a partitioned algorithm for the 3D-1D coupling and solve the coupling conditions explicitly, without any additional sub-iteration within the same time step.

5.1. Some notes about the 3D FSI formulation

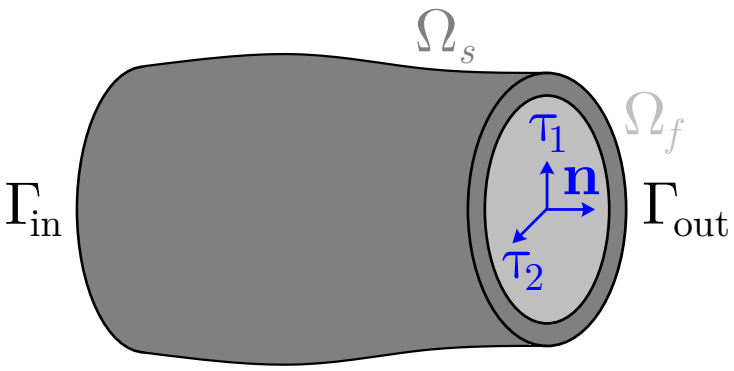


Figure 5.1: 3D compliant tube. Ω_f and Ω_s represent the fluid and structure 3D domains, respectively. Γ_{in} and Γ_{out} represent the inlet and outlet sections, respectively. For each cross section, $\boldsymbol{\tau}_1$ and $\boldsymbol{\tau}_2$ are the two tangential unit vector, while $\mathbf{n}$ is the normal unit vector.

The 3D FSI formulation adopted here is the same as the one described in [19], which is composed of a fluid subproblem, a structure subproblem and the corresponding interface

conditions, namely:

$$\begin{cases} \left.\begin{aligned} &\rho_f \frac{\partial \mathbf{u}_{3D}}{\partial t} + \rho_f \left((\mathbf{u}_{3D} - \mathbf{u}_{ALE}) \cdot \nabla \right) \mathbf{u}_{3D} - \nabla \cdot \mathrm{T}_f = \mathbf{0} && \forall \mathbf{x} \in \Omega_f,\ \forall t \in (0, t_f] \\ &\nabla \cdot \mathbf{u}_{3D} = 0 && \forall \mathbf{x} \in \Omega_f,\ \forall t \in (0, t_f] \end{aligned} \right\} \text{fluid eqs.} \\ \left.\begin{aligned} &\mathbf{u}_{3D} = \frac{\partial \boldsymbol{\eta}}{\partial t} && \forall \mathbf{x} \in \Gamma_{fs},\ \forall t \in (0, t_f] \\ &\mathrm{T}_f \cdot \mathbf{n} = \mathrm{T}_s \cdot \mathbf{n} && \forall \mathbf{x} \in \Gamma_{fs},\ \forall t \in (0, t_f] \end{aligned} \right\} \text{interface conditions} \\ \left.\begin{aligned} &\rho_s \frac{\partial^2 \widetilde{\boldsymbol{\eta}}}{\partial t^2} - \nabla \cdot \widetilde{\mathrm{T}}_s = \mathbf{0} && \forall \mathbf{x} \in \widetilde{\Omega}_s,\ \forall t \in (0, t_f] \end{aligned} \right\} \text{structure eqs.} \end{cases} \tag{5.1}$$

where:

- Ω_f and Ω_s are the fluid and structure domain, respectively, written in the deformed configuration (as shown in Figure 5.1), while $\widetilde{\Omega}_f$ and $\widetilde{\Omega}_s$ are their counterparts in the reference configuration (we consider the initial configuration as the reference configuration); this entails that both the domains Ω_f and Ω_s vary in time; the interface between the two subproblems is identified by $\Gamma_{fs} := \Omega_f \cap \Omega_s$; hereafter, we will also denote the 3D domain as $\Omega_{3D} := \Omega_f \cup \Omega_s$;
- ρ_f is the fluid density;
- $\mathbf{u}_{3D}$ represents the fluid velocity;
- $\mathrm{T}_f = -P_{3D}\mathbb{I} + \mu \left(\nabla \mathbf{u}_{3D} + (\nabla \mathbf{u}_{3D})^\top \right)$ is the fluid stress tensor, with P_{3D} being the pressure and μ being the fluid viscosity;
- ρ_s is the structure density;
- $\boldsymbol{\eta}$, $\widetilde{\boldsymbol{\eta}}$ represent the structure displacement in the Eulerian and Lagrangian reference frames, respectively;
- T_s is the structure Cauchy stress tensor, while $\widetilde{\mathrm{T}}_s$ is the first Piola-Kirchhoff stress tensor, linked by the relation: $\widetilde{\mathrm{T}}_s = J \mathrm{T}_s \mathrm{F}^{-\top}$, with $J = \det \mathrm{F}$ and $\mathrm{F} = \nabla \mathbf{x}$ being the

deformation gradient.

We remark that the structure subproblem in (5.1) is written in Lagrangian coordinates, as it is customary in solid mechanics, while the fluid subproblem is expressed in an arbitrary Lagrangian-Eulerian (ALE) framework, in order to deal with moving boundaries (see [31]).

Moreover, under the hypothesis of an hyperelastic material, the first Piola-Kirchhoff stress tensor can be obtained in terms of a suitable strain energy density function (SEDF) Θ:

$$\widetilde{\mathrm{T}}_s = \frac{\partial \Theta}{\partial \mathrm{F}} \, . \tag{5.2}$$

Common expressions for the SEDF include:

- *Hookean model:* under the hypothesis of sufficiently small strain, Θ can be expressed by:
$$\Theta = \frac{E}{2(1+\xi)} \, \varepsilon(\mathrm{F}) : \varepsilon(\mathrm{F}) + \frac{\xi E}{2(1+\xi)(1-2\xi)} \left[\mathrm{tr}(\varepsilon(\mathrm{F}))\right]^2 , \tag{5.3}$$
with E being the Young modulus, ξ the Poisson ratio, $\varepsilon(\mathrm{F})$ the small strain tensor, which is equal to:
$$\varepsilon(\mathrm{F}) = \frac{1}{2}\left(\mathrm{F} + \mathrm{F}^\top\right) - \mathbb{I} \, ; \tag{5.4}$$

- *Saint Venant-Kirchhoff model:* this constitutive model holds in the case of finite deformation; hence, Θ is given by:
$$\Theta = \frac{E}{2(1+\xi)} \, \mathrm{E}(\mathrm{F}) : \mathrm{E}(\mathrm{F}) + \frac{\xi E}{2(1+\xi)(1-2\xi)} \left[\mathrm{tr}(\mathrm{E}(\mathrm{F}))\right]^2 , \tag{5.5}$$
where $\mathrm{E}(\mathrm{F})$ represents the Green-Lagrange strain tensor, formally expressed as:
$$\mathrm{E}(\mathrm{F}) = \frac{1}{2}\left(\mathrm{F}^\top \mathrm{F} - \mathbb{I}\right) \, . \tag{5.6}$$

Throughout this work, we will consider the following boundary conditions for the standalone 3D FSI problem.

- *Fluid subproblem*:
  - *Inlet section* ($\Gamma_{f,\mathrm{in}} := \Omega_f \cap \Gamma_{\mathrm{in}}$): the pressure is prescribed at the inlet by means of the following Neumann boundary condition:
$$\mathrm{T}_f \, \mathbf{n} = -P_{\mathrm{in}}(t) \, \mathbf{n} \, , \tag{5.7}$$

where $P_{\text{in}}(t)$ is an arbitrary pressure profile.

– *Outlet section* ($\Gamma_{f,\text{out}} := \Omega_f \cap \Gamma_{\text{out}}$): the interaction between the fluid and structure problems gives rise to an hyperbolic system; this implies that we should choose the outlet boundary condition carefully, in order to avoid spurious wave reflections. To address this issue, we define a non-reflecting boundary condition for the fluid subproblem, which consists of the following resistive condition:

$$P_{\text{3D}} = R_t \, Q_{\text{3D}} \qquad \forall \mathbf{x} \in \Gamma_{f,\text{out}} \, , \tag{5.8}$$

where R_t represents the terminal resistance and is equal to:

$$R_t = \frac{\rho_f \, c_0}{A_0} \, . \tag{5.9}$$

The intermediate steps in the derivation of this boundary condition are reported in Appendix D. From a numerical standpoint, (5.8) is prescribed on the fluid stress tensor:

$$\mathrm{T}_f \, \mathbf{n} = -R_t \, Q_{\text{3D}} \, \mathbf{n} \qquad \forall \mathbf{x} \in \Gamma_{f,\text{out}} \, . \tag{5.10}$$

Finally, from an implementation perspective, Q_{3D} is taken as a proxy of the actual flow rate at the outlet and is approximated by the value computed at the previous time step:

$$\mathrm{T}_f \, \mathbf{n}\Big|_{t^{n+1}} = -R_t \, Q_{\text{3D}} \, \mathbf{n}\Big|_{t^n} \qquad \forall \mathbf{x} \in \Gamma_{f,\text{out}} \, . \tag{5.11}$$

The error induced by this approximation will produce minor spurious wave reflections at the outlet, as discussed in Section 7.4.

- *Structure subproblem*: at both ends of the 3D compliant tube (denoted by $\Gamma_{s,\text{in}} := \Omega_s \cap \Gamma_{\text{in}}$ and $\Gamma_{s,\text{out}} := \Omega_s \cap \Gamma_{\text{out}}$), we allow the structure to move radially by means of homogeneous Neumann boundary conditions, while the normal displacement is set to zero, namely:

$$\begin{aligned} (\mathrm{T}_s \, \mathbf{n}) \cdot \boldsymbol{\tau}_1 &= 0 \qquad \forall \mathbf{x} \in \Gamma_{s,\text{in}} \cup \Gamma_{s,\text{out}} \, , \\ (\mathrm{T}_s \, \mathbf{n}) \cdot \boldsymbol{\tau}_2 &= 0 \qquad \forall \mathbf{x} \in \Gamma_{s,\text{in}} \cup \Gamma_{s,\text{out}} \, , \\ \boldsymbol{\eta} \cdot \mathbf{n} &= 0 \qquad \forall \mathbf{x} \in \Gamma_{s,\text{in}} \cup \Gamma_{s,\text{out}} \, . \end{aligned} \tag{5.12}$$

Concerning the numerical approximation, we employ a continuous Galerkin finite element method for both the fluid and the structure subproblems, with Lagrange polynomials as

basis functions, while the time discretization is obtained by means of implicit backward differentiation formula (BDF) schemes. We refer to [21] for the weak formulation of both fluid and structure subproblems, and to [20] for a more detailed analysis of BDF schemes. It is worth noting that, throughout this work, we employ $\mathbb{P}_1$-$\mathbb{P}_1$ elements for the fluid subproblem and $\mathbb{P}_1$ for the structure subproblem, while the time discretization is achieved with a 2^{nd}-order BDF scheme. SUPG-PSPG stabilization is used to avoid artificial checkerboard patterns in the pressure, as proposed in [32]. Further numerical tests are reported in Section 7.4.

5.2. Coupling conditions

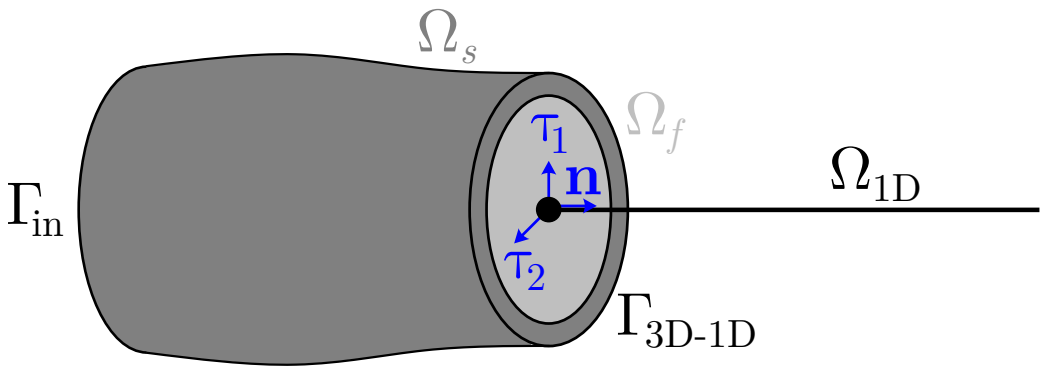


Figure 5.2: 3D compliant tube, coupled with 1D blood flow model at the outlet. Ω_f and Ω_s represent the fluid and structure 3D domains, respectively. Ω_{1D} represents the 1D domain. Γ_{in} is the inlet section. $\Gamma_{3D\text{-}1D}$ is the coupling interface between the 3D FSI problem and the 1D blood flow model. For each cross section, $\boldsymbol{\tau}_1$ and $\boldsymbol{\tau}_2$ are the two tangential unit vector, while $\mathbf{n}$ is the normal unit vector.

In this section, we analyze the possible matching conditions to couple the 3D FSI model (5.1) and the 1D blood flow model (2.2). As proposed in [29], we should enforce the following coupling conditions on the continuous formulation, representing the continuity of flow rate, cross-sectional area and normal stress between the two dimensionally heterogenous problems:

$$\int_{\Gamma_{f,3D\text{-}1D}} \mathbf{u}_{3D} \cdot \mathbf{n}\, d\Gamma = Q_{1D}\Big|_{\Gamma_{3D\text{-}1D}}, \tag{5.13}$$

$$\int_{\Gamma_{3D\text{-}1D}} d\Gamma = A_{1D}\Big|_{\Gamma_{3D\text{-}1D}}, \tag{5.14}$$

$$\frac{1}{|\Gamma_{f,3D\text{-}1D}|}\int_{\Gamma_{f,3D\text{-}1D}} (\mathrm{T}_f\, \mathbf{n}) \cdot \mathbf{n}\, d\Gamma = -P_{1D}\Big|_{\Gamma_{3D\text{-}1D}}, \tag{5.15}$$

where:

- $\Gamma_{\text{3D-1D}} := \Omega_{\text{3D}} \cap \Omega_{\text{1D}}$ represents the coupling interface between the 3D and 1D domains, while $\Gamma_{f,\text{3D-1D}} := \Omega_f \cap \Gamma_{\text{3D-1D}}$, as shown in Figure 5.2;
- $\mathbf{n}$ is the unit vector normal to the coupling interface $\Gamma_{\text{3D-1D}}$;
- $\mathbf{u}_{\text{3D}}$ is the fluid velocity in the 3D FSI problem;
- Q_{1D} and A_{1D} are the the flow rate and cross-sectional area of the 1D model, while P_{1D} is the pressure, which is linked to A_{1D} by the constitutive law (2.3);
- T_f represents the stress tensor for the 3D fluid subproblem.

It is worth noting that the conditions (5.14)–(5.15) are not compatible, since the cross-sectional area and the average normal stress of the 3D FSI formulation are not required to satisfy the tube law (2.3), which in turn links the cross-sectional area A_{1D} to the pressure P_{1D} in the 1D model. That is to say, we need to discard one matching condition among (5.14)–(5.15). Consequently, in this work, we will consider only the coupling condition (5.15), as further analyzed in Section 5.2.1.

5.2.1. Coupling conditions – 3D FSI model

We highlight that all the coupling conditions listed in (5.13)–(5.14)–(5.15) are integral relations for the 3D FSI problem, while Navier-Stokes equations require pointwise data as boundary conditions; this issue is known in literature as *defective boundary conditions*. As proposed in [29], we proceed by assuming that the normal stress of the 3D fluid problem evaluated at $\Gamma_{\text{3D-1D}}$ is constant, while the tangential stresses are considered to be null (which is a physically meaningful hypothesis for compliant tubes). Thus, (5.15) is converted into a non-homogeneous Neumann boundary condition, formally:

$$
\begin{aligned}
&(\mathrm{T}_f\,\mathbf{n})\cdot\boldsymbol{\tau}_1 = 0 \qquad \forall \mathbf{x} \in \Gamma_{f,\text{3D-1D}}\ ,\\
&(\mathrm{T}_f\,\mathbf{n})\cdot\boldsymbol{\tau}_2 = 0 \qquad \forall \mathbf{x} \in \Gamma_{f,\text{3D-1D}}\ ,\\
&\frac{1}{|\Gamma_{f,\text{3D-1D}}|}\int_{\Gamma_{f,\text{3D-1D}}} (\mathrm{T}_f\,\mathbf{n})\cdot\mathbf{n}\,\mathrm{d}\Gamma = -P_{\text{1D}}\Big|_{\Gamma_{\text{3D-1D}}} \xrightarrow{\mathrm{T}_f\,\mathbf{n}\approx\text{const}} (\mathrm{T}_f\,\mathbf{n})\cdot\mathbf{n} = -P_{\text{1D}}\Big|_{\Gamma_{\text{3D-1D}}}\ .
\end{aligned}
\tag{5.16}
$$

Concerning the 3D structure problem, we still enforce a homogeneous Dirichlet boundary condition along the normal direction and homogeneous Neumann boundary conditions

along the tangential directions, as formally written in (5.12).

This is not the unique method to treat the 3D-1D coupling and an exhaustive list of numerical techniques can be found in [19, 30]. For instance, we could prescribe the continuity of the flow rate on the fluid subproblem (5.13), giving rise to a flow rate defective boundary condition; various approaches have been devised to deal with defective boundary conditions in compliant tubes, e.g. an augmented formulation involving Lagrange multipliers is used to enforce the flow rate in [33]. Regarding the 3D structure subproblem, we could enforce the continuity of the cross-sectional area (5.14); however, this approach would introduce numerical instabilities and spurious oscillations at the coupling interface, as shown in [29]. Indeed, [29] proves some relevant stability results only when homogeneous boundary conditions are prescribed on the structure subproblem; this justifies our choice of (5.16)–(5.12) as coupling conditions.

5.2.2. Coupling conditions – 1D haemodynamics model

As proposed in [28], it is reasonable to enforce the continuity of the flow rate and cross-sectional area at the coupling interface $\Gamma_{\text{3D-1D}}$ by means of the incoming characteristic variable W_{in}, formally:

$$W_{\text{in}}(A_{\text{1D}}, Q_{\text{1D}})\Big|_{\Gamma_{\text{3D-1D}}} = W_{\text{in}}(A_{\text{3D}}, Q_{\text{3D}})\Big|_{\Gamma_{f,\text{3D-1D}}} , \tag{5.17}$$

where the definition of W_{in} is given by (3.49), while the 3D quantities can be obtained as:

$$\begin{aligned} Q_{\text{3D}}\Big|_{\Gamma_{f,\text{3D-1D}}} &:= \int_{\Gamma_{f,\text{3D-1D}}} \mathbf{u}_{\text{3D}} \cdot \mathbf{n}\, \mathrm{d}\Gamma , \\ \overline{P}_{\text{3D}}\Big|_{\Gamma_{f,\text{3D-1D}}} &:= \frac{1}{|\Gamma_{f,\text{3D-1D}}|} \int_{\Gamma_{f,\text{3D-1D}}} P_{\text{3D}}\, \mathrm{d}\Gamma , \\ A_{\text{3D}}\Big|_{\Gamma_{f,\text{3D-1D}}} &:= A_{\text{3D}}\left(\overline{P}_{\text{3D}}\right)\Big|_{\Gamma_{f,\text{3D-1D}}} , \end{aligned} \tag{5.18}$$

where the latter relation is provided by inverting the tube law (2.3). However, we should recall that an analytical expression for W_{in} exists only for a flat velocity profile (i.e. when $\alpha = 1$, as already discussed in Section 2.4 and in Appendix A). In order to address this issue, we apply Roe's linearization introduced in Section 3.4.2, so that W_{in} is provided by:

$$W_{\text{in}}(A, Q) = \begin{cases} \dfrac{Q}{A} \pm 4c & \text{if } \alpha = 1 \text{ (i.e. for flat velocity profiles)} \\ \widetilde{\mathbf{l}}_{1,2} \cdot \mathbf{U} & \text{if } \alpha \neq 1 \text{ (i.e. for non-flat velocity profiles)} \end{cases} , \tag{5.19}$$

as already reported in (2.24)–(3.37).

As presented in Section 3.5, since we employ a DG weak formulation for the 1D haemodynamics model, we should enforce the boundary conditions on the intermediate state $\mathbf{U}^{*e}$, arising from the wave propagation on the boundary (Figure 3.3). This condition is formally described by the inverse Riemann problem (3.44). This implies that $A_{1D}|_{\Gamma_{3D\text{-}1D}}$ and $Q_{1D}|_{\Gamma_{3D\text{-}1D}}$ considered so far are indeed equivalent to the physical variables evaluated at the intermediate state, i.e. $A_{1D}|_{\Gamma_{3D\text{-}1D}} \equiv A^{*e}$ and $Q_{1D}|_{\Gamma_{3D\text{-}1D}} \equiv Q^{*e}$. As already shown for other boundary conditions in Section 3.5, we can define a residual function $g(A^{*e}, Q^{*e})$ needed to enforce (5.17) on the coupling interface:

$$g(A_{1D}, Q_{1D})\Big|_{\Gamma_{3D\text{-}1D}} \equiv g(A^{*e}, Q^{*e}) = \frac{W_{in}(A^{*e}, Q^{*e}) - W_{in}(A_{3D}, Q_{3D})}{W_{in,ref}} = 0 \,, \tag{5.20}$$

where $W_{in,ref}$ is an arbitrary reference value, which makes the expression (5.20) dimensionless; one possible choice for $W_{in,ref}$ is given by (3.52). Finally, (5.20) can be substituted into the inverse Riemann problem (3.44), hence enforcing the coupling condition (5.17) on the 1D haemodynamics model.

5.3. Sketch of the algorithm

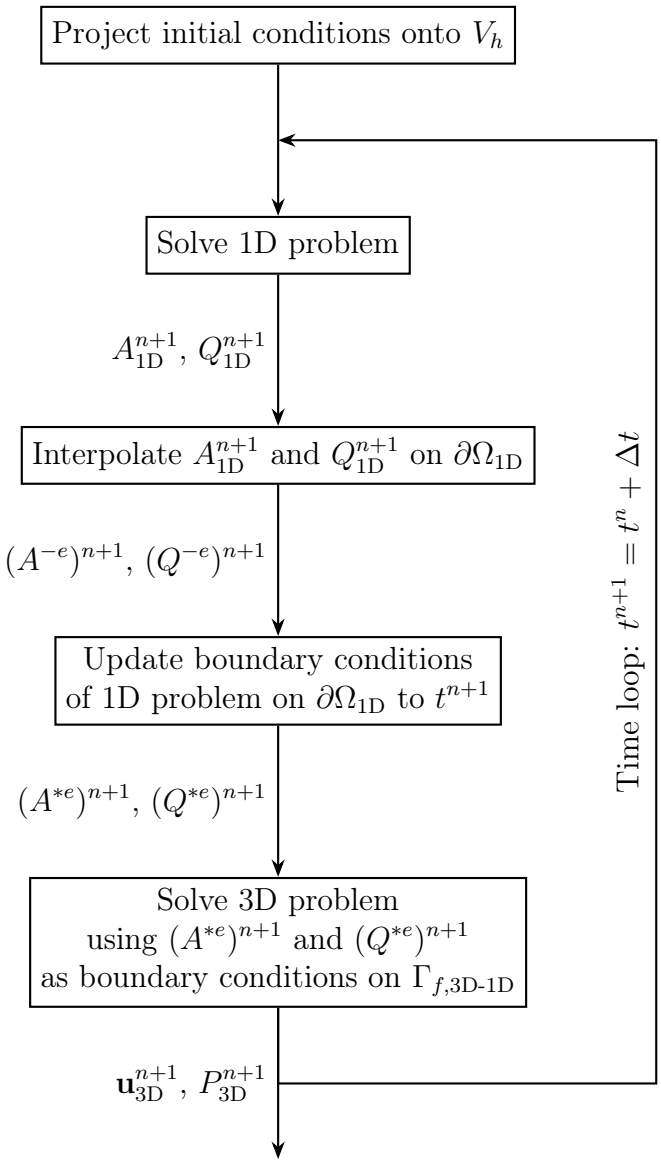
Concerning the implementation of the 3D-1D coupling described above, we adopt a partitioned algorithm (that is, the 1D and 3D subproblems are solved sequentially rather than monolithically). In principle, a partitioned approach requires sub-iterations at each time step to resolve the coupling interface conditions consistently, as the quantities exchanged between the two subdomains (namely pressure and flow rate) are evaluated using information from the previous iteration. However, in this work we proceed as proposed in [29], and employ an explicit coupling strategy, meaning that no additional sub-iteration is performed to correct the quantities evaluated at the coupling interface $\Gamma_{3D\text{-}1D}$. The stability and consistency of this approach is studied more rigorously in [29].

More specifically, given the time step t^{n+1}, firstly we evolve the 1D problem solution to the current time step, then we update the boundary conditions of the 1D model by solving the inverse Riemann problem (3.44). It is noteworthy that the quantities of the 3D model appearing in (5.17) are evaluated at the previous time step t^n, so the residual function (5.20) becomes:

$$g((A^{*e})^{n+1}, (Q^{*e})^{n+1}) = \frac{W_{in}((A^{*e})^{n+1}, (Q^{*e})^{n+1}) - W_{in}((A_{3D})^{n}, (Q_{3D})^{n})}{W_{in,ref}} = 0 \,. \tag{5.21}$$

For sufficiently small time steps, the error introduced by evaluating the quantities in (5.21) at different time steps is negligible and bounded, as proven in [29]. Finally, the 3D model can be updated by using the boundary conditions (5.7)–(5.12) on the inlet Γ_{in} and (5.16)–(5.12) on the coupling interface $\Gamma_{\text{3D-1D}}$. The algorithm is illustrated in Algorithm 5.1.

Algorithm 5.1 3D-1D coupling.



6 | Software implementation

The entire implementation of this 1D blood flow model has been developed within the `life`^{x} project, written in C++17 standard (see [13, 14] or visit the `life`^{x} website[6.1]). `life`^{x} is a high-performance library for multiphysics and multiscale simulations, focused on cardiovascular application; its core utilities include a CFD solver with moving boundaries, an FSI solver, an electrophysiology solver for the cardiac function and other more general-purpose tools, such as mesh readers/generators, solvers for linear and non-linear systems, output handlers, etc. Concerning the finite element implementation, `life`^{x} relies on `deal.II` open-source library, so most of the conventions adopted below refers to `deal.II` code conventions (see [15] or visit `deal.II` website[6.2]). Moreover, `deal.II` offers some parallelization capabilities, whose performance is further analyzed in Section 7.6. In this chapter, we provide the code design for both the standalone 1D vascular flow solver and the 3D-1D coupling implementation.

6.1. Standalone 1D haemodynamics model

For the time being, the 1D haemodynamics model is structured as a `life`^{x} example (i.e. located at `examples/blood_flow_1d` directory), but a simple generalization of the existing code may be carried out, in order to include it into `life`^{x} shared library.

All the classes have been designed to represent the underlying mathematical and numerical formulations, adhering to the single-responsibility principle (SRP) as much as possible. A list of the most important implemented classes is offered below.

1. `lifex::examples::BloodVessel` (Figure 6.1) is responsible for storing and manipulating the blood vessel properties, such as the Young modulus E, the wall thickness h, the reference values P_{ref} and A_{ref} for the tube law (2.3), the blood density ρ, etc. These properties are grouped together into an aggregate, named `ParamProperties`. This class inherits from the abstract class `lifex::CoreModel`, as it is customary

[6.1] https://lifex.gitlab.io/lifex-public/
[6.2] https://dealii.org/

within `life`[x], in order to provide a common interface for declaring and parsing parameters from `.prm` files. In addition, this class offers a `solve_riemann` method implementing the direct Riemann problem expressed by (3.40) and some `compute_*` methods, which intuitively reflect the underlying mathematical formulation, e.g. `compute_pressure_from_area` implements the tube law (2.3), `compute_eigenvalue` returns one of the eigenvalues of $\mathrm{H}(\mathbf{U})$ (2.17), `compute_left_eigenvector` and `compute_right_eigenvector` return the left and right eigenvectors (2.19), etc.

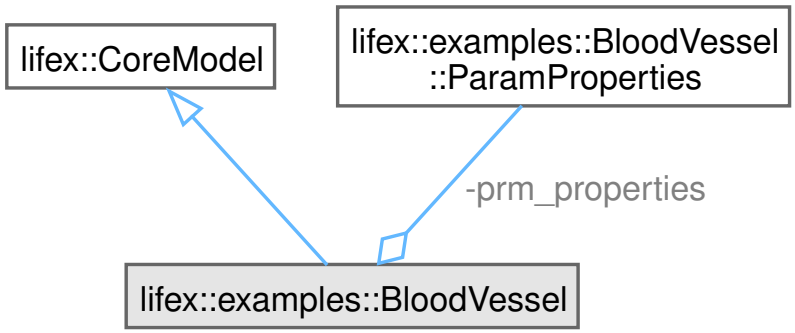


Figure 6.1: UML diagram for `lifex::examples::BloodVessel`.

2. `lifex::examples::BloodVesselHelpers::RiemannData` stores internally the Riemann problem data, in order to provide an intuitive interface for accessing the left state $\mathbf{U}_L^e$, the right state $\mathbf{U}_R^e$, the internal state $\mathbf{U}^{-e}$ and the external state $\mathbf{U}^{+e}$, as illustrated in Figure 3.3.

3. `lifex::examples::BloodVesselHelpers::InverseRiemannSolverHandler` implements the inverse Riemann problem for an arbitrary number of vessels, i.e. it is capable of handling the boundary conditions (3.44), the discontinuous material points (4.2) and the bifurcations (4.6). The numerical solution of the inverse Riemann problem is achieved by employing Newton's method for non-linear systems.

4. `lifex::examples::BloodVesselBCs::BC` (Figure 6.2) handles the boundary conditions, which are implemented polymorphically. In particular, the inverse Riemann problem (3.44) shares the same mathematical structure for all the boundary conditions, except for the boundary residual function $g(A^{*e}, Q^{*e})$, which is provided by the purely virtual method `boundary_residual` and must be overridden by all the derived classes. This design offers high flexibility in implementing new boundary conditions, since one needs only to define a custom boundary residual function.

5. `lifex::examples::BloodVesselBCs::BCParams` (Figure 6.2) inherits from `BC` ab-

stract class and from `lifex::CoreModel`, in order to let the user specify custom parameters in `.prm` file before launching the model. This class is used as the base class to implement all the boundary conditions listed in Section 3.5, namely:

- `lifex::examples::BloodVesselBCs::InletBC` provides the inlet boundary condition (Section 3.5.1); it is an abstract class, declaring a purely virtual method `compute_value`, which is used to specify a custom profile for the inlet quantities, e.g. `lifex::examples::BloodVesselBCs::GaussianPulse` inherits from this class and prescribes a Gaussian-shaped single pulse (as defined in (7.1)), `lifex::examples::BloodVesselBCs::InletCSV` reads inlet data from `.csv` files, etc.;
- `lifex::examples::BloodVesselBCs::NonReflectingBC` implements the non-reflecting boundary condition (Section 3.5.2);
- `lifex::examples::BloodVesselBCs::WindkesselBC` is used for the coupling with the 3-element Windkessel model (Section 3.5.3).

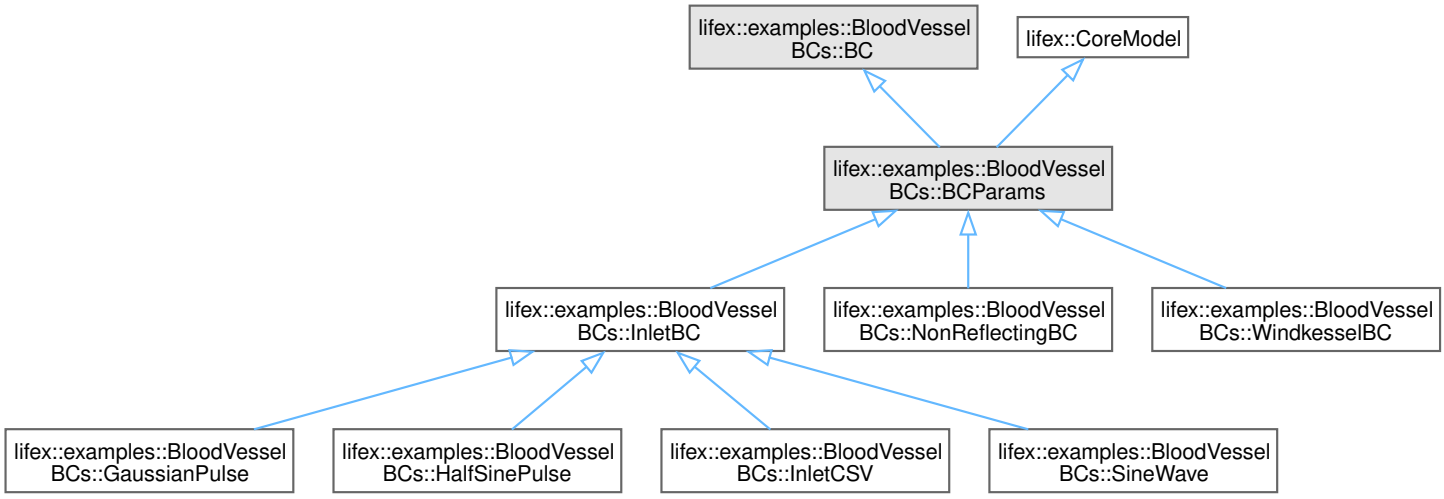


Figure 6.2: UML diagram for `lifex::examples::BloodVesselBCs::BC` and `lifex::examples::BloodVesselBCs::BCParams`.

6. `lifex::utils::ABHandler` implements the explicit AB time-stepping scheme (3.18) (up to 4^{th} order). Unlike the other classes commented in this section, it is located at `source/core/numerics/time_handler.hpp` and is designed to be completely general-purpose, i.e. it may prove useful in many other algorithms where AB schemes are required for the time discretization.
7. `lifex::examples::BloodFlow1D` (Figure 6.3) collaborates with other classes to

implement the whole 1D blood flow model, as illustrated in Algorithm 3.1. Each class provides a specific functionality of the numerical method, following the SRP principle, more specifically:

- `lifex::utils::MeshHandler` creates the mesh or loads it from a `.msh` file;
- `lifex::utils::NonLinearSolverHandler` implements Newton's method for non-linear systems and it is used for the numerical solution of the inverse Riemann problem (3.44)–(4.2)–(4.6);
- `lifex::utils::OutputHandlerBase` and `lifex::utils::RestartHandler` are used for output and serialization purposes;
- `lifex::utils::ABHandler` handles the time discretization by employing an AB scheme, as discussed above.

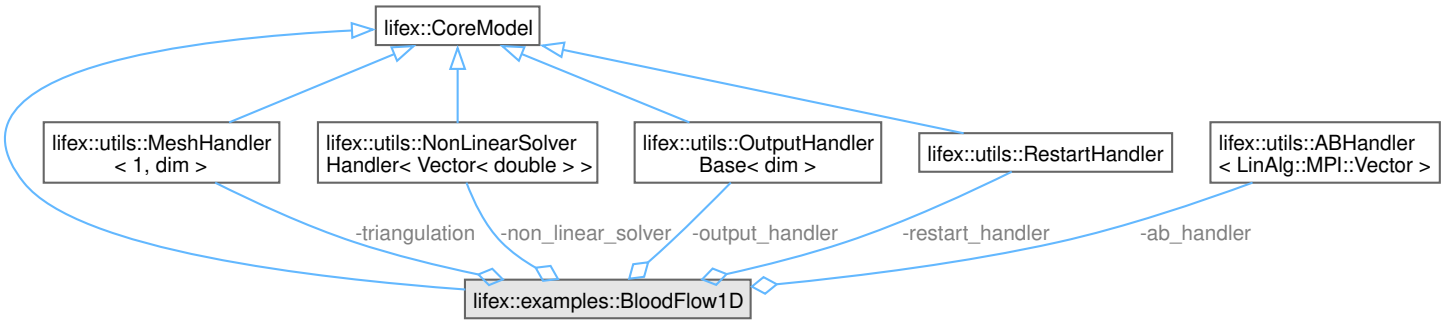


Figure 6.3: UML diagram for `lifex::examples::BloodFlow1D`.

8. `lifex::examples::ExampleBloodFlow1D` (Figure 6.4) serves as an example class for using `lifex::examples::BloodFlow1D` and setting up boundary conditions.

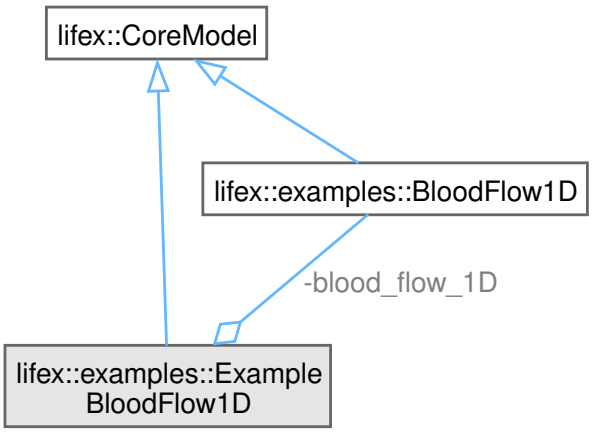


Figure 6.4: UML diagram for `lifex::examples::ExampleBloodFlow1D`.

6.2. 3D-1D coupling implementation

The 3D-1D coupling is achieved by interfacing the current 1D haemodynamics model (i.e. `lifex::examples::BloodFlow1D`) with `lifex` built-in 3D FSI model (i.e. `lifex::FSI`, which in turn relies on `lifex::FluidDynamics` for the fluid subproblem and on `lifex::Mechanics` for the structure subproblem). Again, this code is located among the other `lifex` examples at `examples/coupling_3d_1d`, but it is designed to possibly be included within `lifex` core utilities.

Given the explicit coupling described in Algorithm 5.1, it is sufficient to evolve the 1D blood flow solver at each time step, exchange the interface quantities between the two models by means of suitable coupling conditions and then evolve the 3D FSI solver with the updated quantities. For this purpose, we have implemented the following classes within the `lifex::examples::CouplingBCs` namespace.

1. `lifex::examples::CouplingBCs::Coupling1D` (Figure 6.5) implements the explicit coupling condition (5.21) on the 1D haemodynamics model. This is achieved by overriding the `boundary_residual` method from `lifex::examples::BloodVesselBCs::BC`, thus providing a suitable residual function $g(A^{*e}, Q^{*e})$. This class stores a constant reference to the `lifex::FluidDynamics` instance, in order to interpolate the flow rate $Q_{3D}|_{\Gamma_{f,3D\text{-}1D}}$ and the pressure $P_{3D}|_{\Gamma_{f,3D\text{-}1D}}$ on the coupling interface $\Gamma_{f,3D\text{-}1D}$.

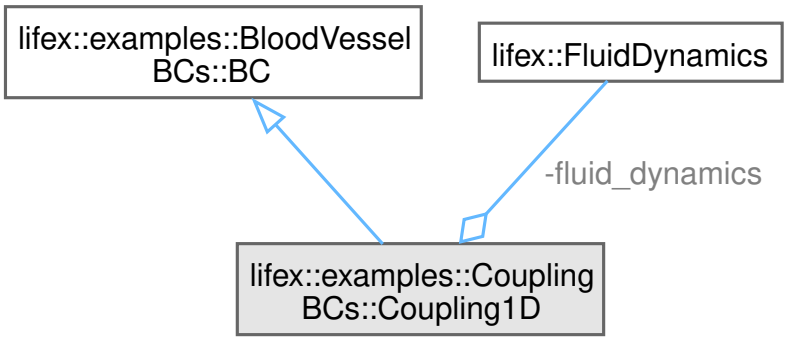


Figure 6.5: UML diagram for `lifex::examples::CouplingBCs::Coupling1D`.

2. `lifex::examples::CouplingBCs::CouplingFluidNeumann` (Figure 6.6) inherits from `lifex::utils::FunctionNeumann`, which is the base class used to define Neumann boundary conditions in `lifex`. It provides the coupling condition on the 3D fluid subproblem (5.16).

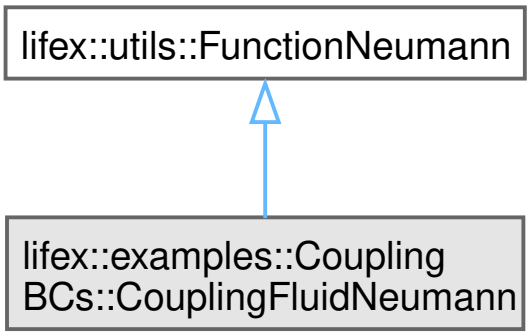


Figure 6.6: UML diagram for `lifex::examples::CouplingBCs::CouplingFluidNeumann`.

Finally, these two classes interact together within the class `lifex::examples::ExampleCoupling3D1D` (Figure 6.7), which is responsible for handling the 3D-1D coupling described in Algorithm 5.1.

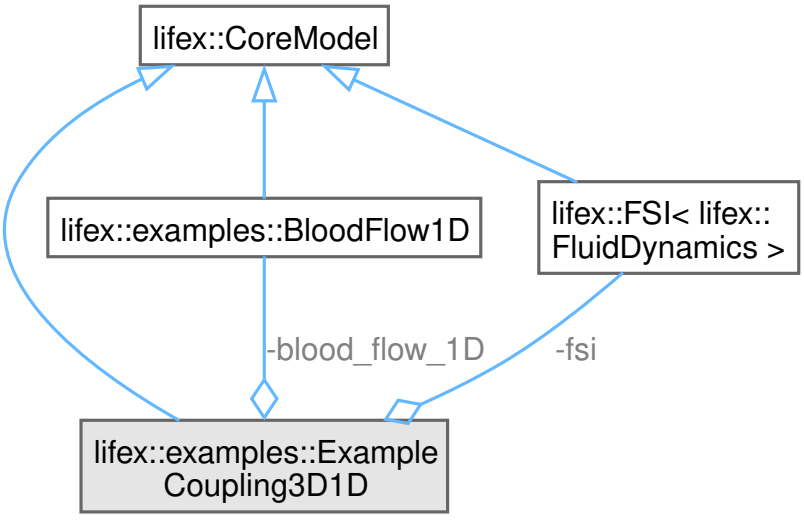


Figure 6.7: UML diagram for `lifex::examples::ExampleCoupling3D1D`.

7 | Numerical results

In this chapter, we present relevant numerical results, to show some possible applications of our software. Firstly, we verify the current implementation against benchmark solutions from literature [5]. Then, we compare Roe's approximate Riemann solver (Section 3.4.2) with the exact Riemann solver (Section 3.4.1), showing that we cannot obtain a real advantage by using Roe's linearization, even though it allows to choose an arbitrary velocity profile and is more physically consistent. As proof of concept, a stented vessel is also analyzed and a preliminary study on the 3D-1D coupling is carried out, underlining the spurious wave reflections that may arise at the coupling interface. Finally, we analyze the convergence rate in terms of mesh spacing (i.e. the h-convergence), polynomial degree (i.e. the p-convergence) and time step (i.e. the time convergence).

7.1. Verification against benchmark solutions

Here we perform a verification of the implemented numerical method. For this purpose, we will refer to five main test cases, whose parameters are reported in Tables E.1–E.2. The simulation setup, together with the reference solutions, is taken from the benchmark paper [5]. It is worth mentioning that [5] employs the inconsistent approach described in Section 2.5, i.e. it assumes $\alpha = 1$ (2.4) while still introducing the viscous effects by means of the viscous coefficient $K_R \neq 0$ (2.5); this procedure would allow to use the characteristic variables $W_{1,2}$ and to solve the Riemann problem exactly, hence using the analytical expressions (3.32)–(3.33). However, in this work we have adopted an approximate Riemann solver in order to introduce consistently a momentum flux correction factor $\alpha \neq 1$ with non-flat velocity profiles. A list of all the test cases is offered below.

1. *inviscid Gaussian pulse*: a Gaussian-shaped single pulse is studied in a 10-m long duct, while the velocity profile is assumed to be flat (and therefore the flow is inviscid); the inflow is thus prescribed as:

$$Q_{\text{in}}(t) = Q_{\text{max}} \exp\left(-a\,(t - t_0)^2\right) \tag{7.1}$$

with $Q_{\max} = 10^{-6}\,\mathrm{m}^3/\mathrm{s}$, $t_0 = 0.05\,\mathrm{s}$ and an attenuation factor equal to $a = 10^4\,\mathrm{s}^{-2}$. At the outlet, a non-reflecting boundary condition is enforced, as described in Section 3.5.2. The parameters are chosen such that the reference wave speed is equal to:

$$c_{\mathrm{ref}} = \sqrt{\frac{\beta}{2\rho A_{\mathrm{ref}}}} A_{\mathrm{ref}}^{\frac{1}{4}} = 6.172\,\mathrm{m/s}\ , \tag{7.2}$$

while the characteristic time T can be defined as the time interval needed for the Gaussian-shaped inflow to reach a value of $Q_{\mathrm{in}} = 1\%\ Q_{\max}$, i.e.:

$$T = 2\sqrt{\frac{2\ln(10)}{a}} = 0.042\,92\,\mathrm{s}\ . \tag{7.3}$$

Consequently, the wavelength is approximately $c_{\mathrm{ref}} T = 0.2649\,\mathrm{m}$, significantly smaller than the tube length. This allows us to study the entire Gaussian-shaped waveform along the duct.

2. *viscous Gaussian pulse*: the setup is the same as the *inviscid Gaussian pulse* test case, but the velocity profile is blunt (i.e. $\gamma = 9$ (2.6) and the fluid is actually considered viscous).

3. *common carotid artery*: it consists of a simulation of the blood flow within the *common carotid artery*, which arises from the aortic arch and is responsible for supplying oxygenated blood to the head and neck (Figure 7.1). The inflow $Q_{\mathrm{in}}(t)$ is taken from an *in-vivo* signal of one cardiac cycle (as reported in [5]). At the outlet, we couple the 1D haemodynamics model with a 3-element Windkessel model, as already discussed in Section 3.5.3. The 1D simulation performs 9 cardiac cycles, in order to reach a periodic state and to guarantee independence from the given initial conditions. The initial cross-sectional area A_0 is chosen by prescribing a null initial pressure along the vessel, i.e. by inverting the tube law (2.3) with $P_0 = 0\,\mathrm{Pa}$:

$$A_0 = \left(\sqrt{A_{\mathrm{ref}}} - \frac{A_{\mathrm{ref}}}{\beta} P_{\mathrm{ref}}\right)^2\ . \tag{7.4}$$

4. *upper thoracic aorta*: it represents the proximal segment of the descending aorta, extending from the aortic arch to the diaphragm, and constitutes the primary conduit for oxygenated blood towards the systemic circulation (Figure 7.1). The inflow $Q_{\mathrm{in}}(t)$ is prescribed and taken from an *in-vivo* signal [5], while a 3-element Windkessel boundary condition is applied at the outlet (Section 3.5.3). In this case, 21 cardiac cycles are simulated. The initial cross-sectional area is given by (7.4).

5. *aortic bifurcation*: it is the terminal compartment of the *abdominal aorta*, where it divides into the two *common iliac arteries*, supplying oxygenated blood to the lower portion of the human body (Figure 7.1). This junction is significant when studying the physical wave reflections arising at the bifurcations, thus allowing to verify the implementation described in Section 4.2. The prescribed inflow is taken from [5], a 3-element Windkessel boundary condition is enforced at the outlet (Section 3.5.3). 27 cardiac cycles are simulated to reach a periodic state. The initial cross-sectional area is given by (7.4).

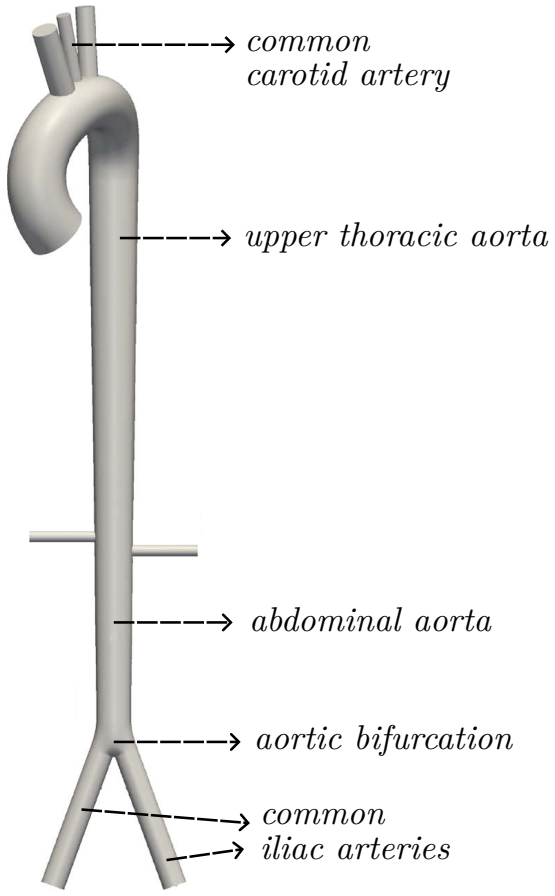


Figure 7.1: 3D model of a full aorta, considering the main branches up to the *aortic bifurcation*, taken from [34]. The coronary and intercostal vessels are neglected in this idealized geometry.

As already discussed in Section 2.1, we need to assume a suitable velocity profile (2.6) (either parabolic or blunt, as depicted in Figure 2.2) apriori, before starting the simulation. This choice is driven by the value of the Womersley number (2.9); hence, we report the values of Wo in Table 7.1, in order to justify the choice of a parabolic velocity profile (for $\mathrm{Wo} \approx 1$) rather than a blunt velocity profile (when $\mathrm{Wo} \gtrsim 10$).

Another important metric for assessing the stability of the numerical scheme is represented

by the CFL criterion (3.19), as already shown in Section 3.2. In order to simplify the evaluation of the CFL condition for each test case, we introduce the following definition of the CFL number:

$$\mathrm{CFL} := \frac{\lambda_{\max}\,\Delta t}{h} \tag{7.5}$$

with $\lambda_{\max} = \max_{i\in\{1,2\}} |\lambda_i|$ being the maximum eigenvalue of the Jacobian matrix $\mathrm{H}(\mathbf{U})$ (2.17), Δt the time step, h the mesh spacing. We recall that, under physiological conditions, the system is inherently subcritical, with $c \gg \alpha\dfrac{Q}{A}$ (as already mentioned in Section 2.4). This implies that $\lambda_{\max} \approx c_{\mathrm{ref}}$, leading to the following approximation for the maximum CFL number:

$$\mathrm{CFL}_{\max} \approx \frac{c_{\mathrm{ref}}\,\Delta t}{h}\ . \tag{7.6}$$

Numerical evidence suggests $\mathrm{CFL}_{\max} = \mathcal{O}(10^{-2})$ to ensure stability for the problem at hand, with a 2^{nd}-order AB time-stepping scheme and Legendre polynomials of degree $p \in [2, 4]$, as reported in Table 7.1.

Parameter	**Units**	*inviscid Gaussian pulse*	*viscous Gaussian pulse*	*common carotid artery*	*upper thoracic aorta*
Pulse period T	s	0.042 92	0.042 92	1.100	0.9550
Womersley number Wo	–	61.99	61.99	3.691	15.85
Velocity profile	–	Flat ($\alpha = 1$)	Blunt $\left(\gamma = 9, \alpha = \frac{11}{10}\right)$	Parabolic $\left(\gamma = 2, \alpha = \frac{4}{3}\right)$	Blunt $\left(\gamma = 9, \alpha = \frac{11}{10}\right)$
Reference wave speed c_{ref}	m/s	6.172	6.172	6.635	5.016
Maximum CFL number $\mathrm{CFL}_{\max}$	–	0.030 86	0.030 86	0.052 66	0.020 78

Parameter	**Units**	*aortic bifurcation* (*abdominal aorta*)	*aortic bifurcation* (*common iliac arteries*)
Pulse period T	s	1.100	1.100
Womersley number Wo	–	10.58	7.382
Velocity profile	–	Blunt $\left(\gamma = 9, \alpha = \frac{11}{10}\right)$	Blunt $\left(\gamma = 9, \alpha = \frac{11}{10}\right)$
Reference wave speed c_{ref}	m/s	6.143	7.268
Maximum CFL number $\mathrm{CFL}_{\max}$	–	0.071 43	0.085 51

Table 7.1: Womersley number (Wo), velocity profile and maximum CFL number ($\mathrm{CFL}_{\max}$) for each test case. Wo is computed using (2.9) and is evaluated at the reference area A_{ref} appearing in the tube law (2.3), while $\mathrm{CFL}_{\max}$ is given by (7.6).

Some plots showing a comparison with the 1D and 3D reference simulations from [5] are illustrated in Figures 7.2–7.3–7.4. Figure 7.2a reports the *inviscid Gaussian pulse* results, where the waveform propagates unchanged along the duct (as expected in an inviscid flow), with a loss on the peak pressure equal to 0.4801 % compared with the inlet peak pressure (essentially due to numerical, but unphysical, diffusion). On the other hand, the

viscous Gaussian pulse waveforms (Figure 7.2b) exhibit an exponential decay on the peak pressure along the duct, due to viscous effects.

It is worth noting that the 1D simulation for the *upper thoracic aorta* case is capable of capturing the dicrotic notch correctly. The dicrotic notch is a physically relevant feature of the pressure waveforms within arteries, and is defined as a local deflection of the pressure due to the wave reflections induced by the closure of the aortic valve; that is to say, the transition between the systole and the diastole is characterized by a local backflow, especially along the aorta, as depicted in Figure 7.3b. Finally, we examine the flow partitioning at the *aortic bifurcation*, where the maximum flow rate decreases from approximately $7.5 \times 10^{-5}\,\mathrm{m}^3/\mathrm{s}$ (at the middle point of *abdominal aorta*) to $2.7 \times 10^{-5}\,\mathrm{m}^3/\mathrm{s}$ (at the middle point of *common iliac arteries*), as illustrated in Figure 7.4.

In order to assess the difference between the results presented in Figures 7.2–7.3–7.4 and the ones obtained in [5], we introduce the following errors:

- *inviscid Gaussian pulse*, *viscous Gaussian pulse*: each time step reported in [5] exhibits a large region where the pressure is null (see Figure 7.2); thus, in order to avoid the division by zero, which would invalidate the error evaluation, we consider only the following absolute error on pressure for these test cases, evaluated at the final time step $t_f = 1.5\,\mathrm{s}$:

$$e_{\mathrm{abs,1D}} = \max_{x \in \mathcal{X}, t = t_f} \left| P_h(x,t) - P_{\mathrm{1D}}(x,t) \right| , \tag{7.7}$$

  where $\mathcal{X}$ denotes the space discretization used in [5], $P_h(x,t)$ is the pressure obtained in this work and $P_{\mathrm{1D}}(x,t)$ is the pressure obtained in [5] with a 1D simulation;

- *common carotid artery*, *upper thoracic aorta*, *aortic bifurcation*: only the results at the middle point of each vessel are available in [5], so we will evaluate the maximum absolute and relative errors at $x = 0.5L$ (with L being the vessel length), with respect to the 1D reference solutions presented in [5]:

$$\begin{aligned} e_{\mathrm{abs,1D}} &= \max_{x = 0.5L, t \in \mathcal{T}} \left| P_h(x,t) - P_{\mathrm{1D}}(x,t) \right| , \\ e_{\mathrm{rel,1D}} &= \max_{x = 0.5L, t \in \mathcal{T}} \left| \frac{P_h(x,t) - P_{\mathrm{1D}}(x,t)}{P_{\mathrm{1D}}(x,t)} \right| , \end{aligned} \tag{7.8}$$

  and with respect to the 3D reference solutions presented in [5], whose results are

averaged over the cross-section to be compared with their 1D counterparts:

$$
\begin{aligned}
e_{\text{abs,3D}} &= \max_{x=0.5L, t\in\mathcal{T}} \left| P_h(x,t) - P_{\text{3D}}(x,t) \right| , \\
e_{\text{rel,3D}} &= \max_{x=0.5L, t\in\mathcal{T}} \left| \frac{P_h(x,t) - P_{\text{3D}}(x,t)}{P_{\text{3D}}(x,t)} \right| ,
\end{aligned}
\tag{7.9}
$$

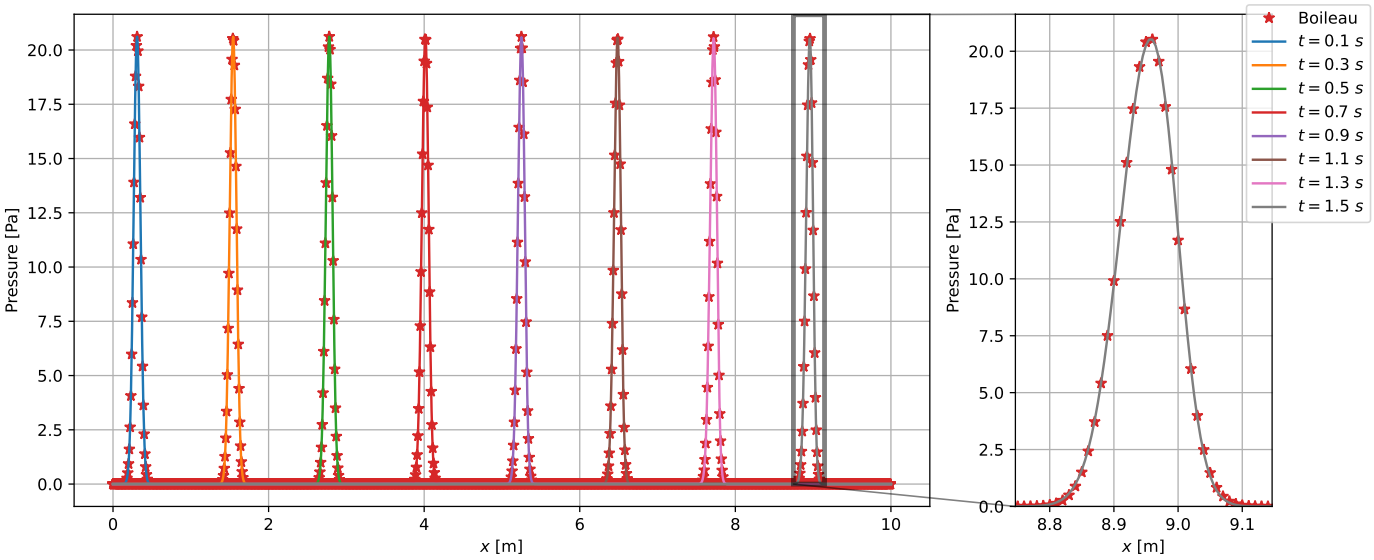
where $\mathcal{T}$ represents the time discretization employed in [5].

We report the values of these errors in Table 7.2, where we highlight that the maximum relative error with respect to the 3D simulations is equal to 3.116 %, thus showing good agreement between 1D and 3D models.

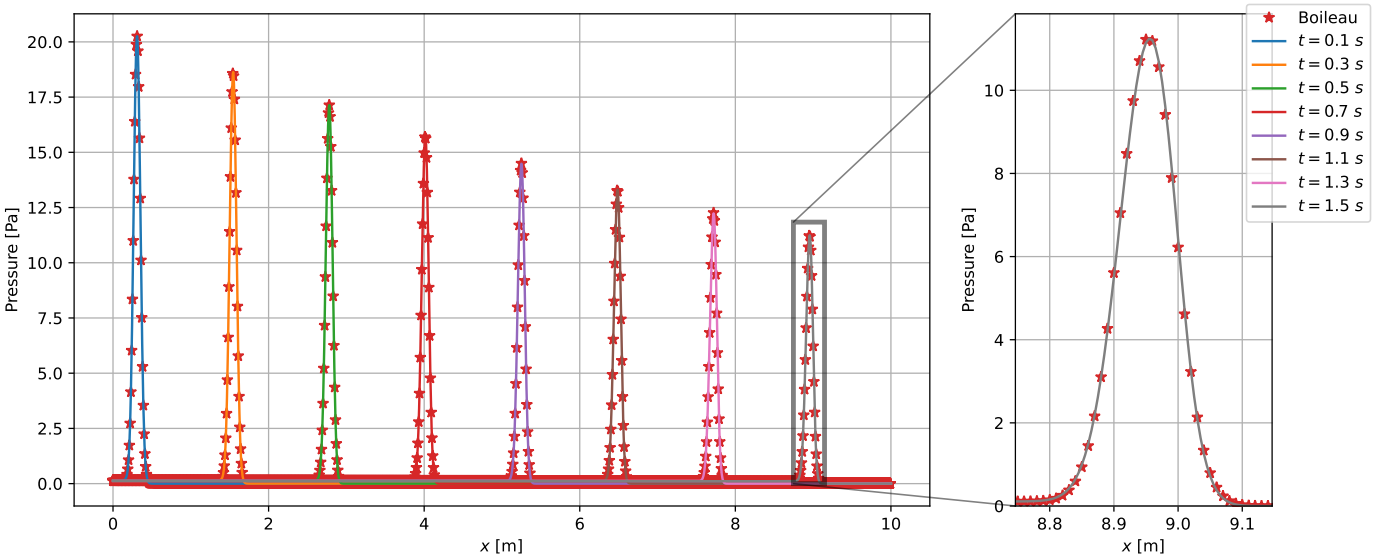
Error	**Units**	*inviscid Gaussian pulse*	*viscous Gaussian pulse*	*common carotid artery*	*upper thoracic aorta*
$e_{\text{abs,1D}}$	Pa	0.3759	0.2400	11.78	71.59
$e_{\text{rel,1D}}$	–	–	–	0.082 76 %	0.5272 %
$e_{\text{abs,3D}}$	Pa	–	–	102.2	430.0
$e_{\text{rel,3D}}$	–	–	–	0.7590 %	3.116 %

Error	**Units**	*aortic bifurcation* (*abdominal aorta*)	*aortic bifurcation* (*common iliac arteries*)
$e_{\text{abs,1D}}$	Pa	20.43	22.70
$e_{\text{rel,1D}}$	–	0.1457 %	0.1620 %
$e_{\text{abs,3D}}$	Pa	105.2	151.2
$e_{\text{rel,3D}}$	–	0.7319 %	0.8965 %

Table 7.2: Errors on pressure with respect to the results obtained in [5]. The subscript "rel" refers to relative errors, while the subscript "abs" refers to absolute errors. The subscript "1D" indicates that the error is computed with respect to the 1D simulations performed in [5], while the subscript "3D" indicates that the error is computed with respect to the 3D simulations performed in [5]. The definitions of $e_{\text{abs,1D}}$ and $e_{\text{rel,1D}}$ are reported in (7.7) for *inviscid Gaussian pulse*, *viscous Gaussian pulse* test cases and in (7.8) for *common carotid artery*, *upper thoracic aorta*, *aortic bifurcation* test cases, while $e_{\text{abs,3D}}$ and $e_{\text{rel,3D}}$ are defined in (7.9). For *aortic bifurcation*, the errors at the middle point of *abdominal aorta* and *common iliac arteries* are reported separately.

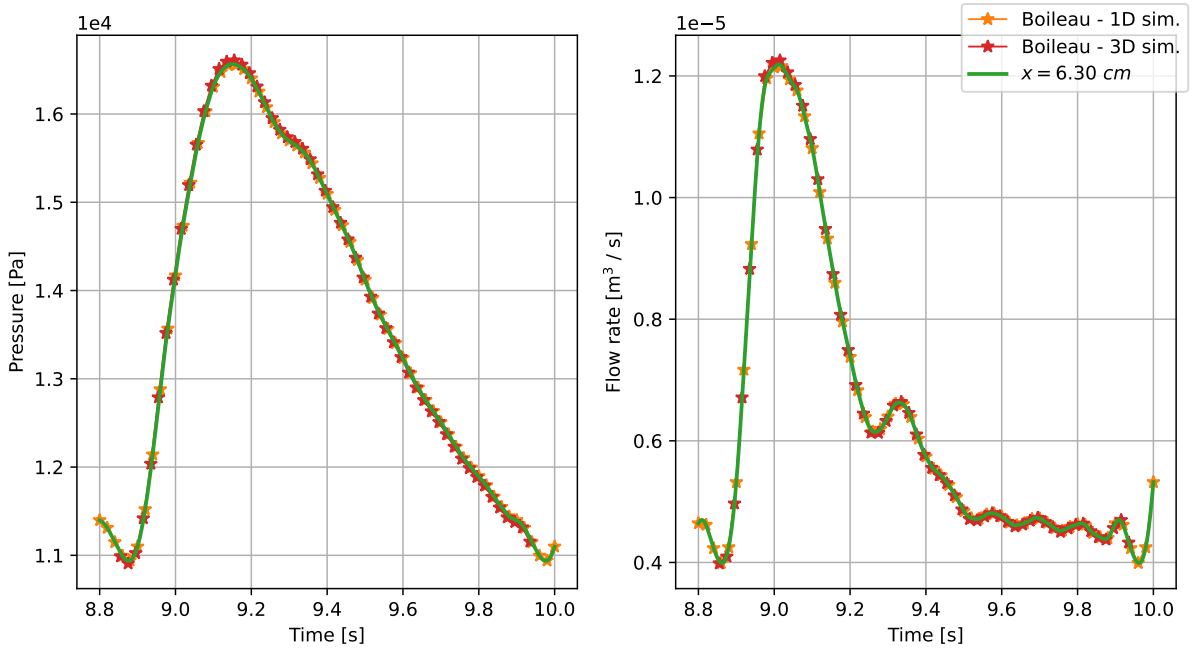


(a) *inviscid Gaussian pulse*.

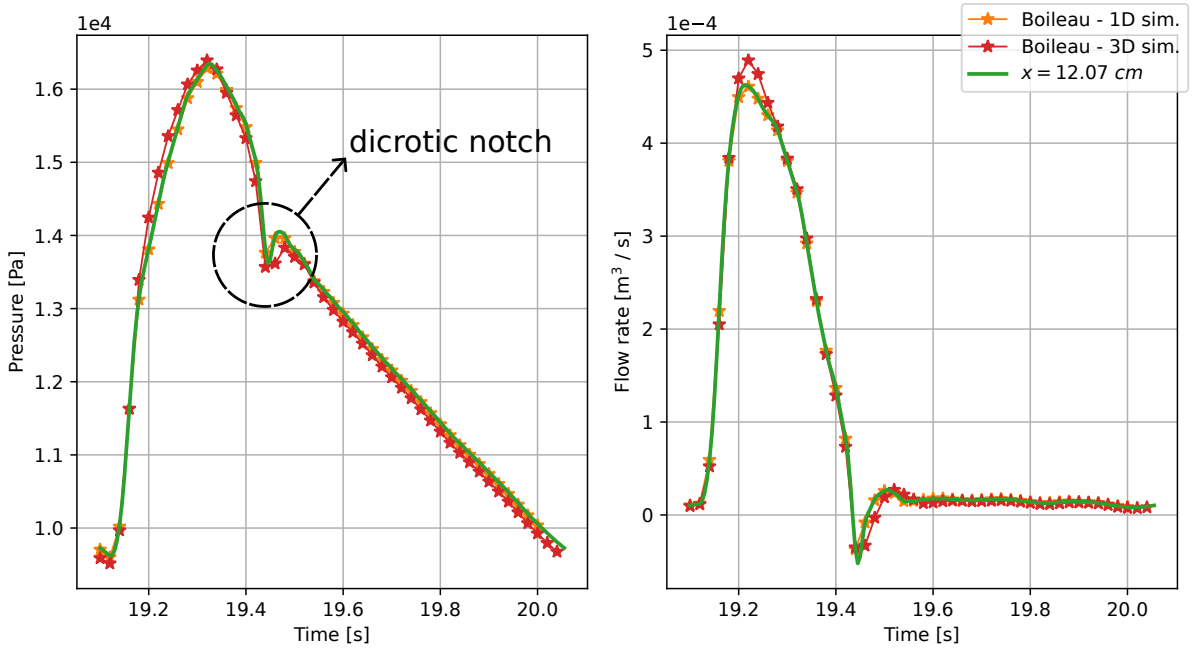


(b) *viscous Gaussian pulse*.

Figure 7.2: *inviscid Gaussian pulse* and *viscous Gaussian pulse* test cases, showing the Gaussian-shaped pressure waveform along the duct at different time steps. The data taken from [5] are marked with the label *Boileau*.

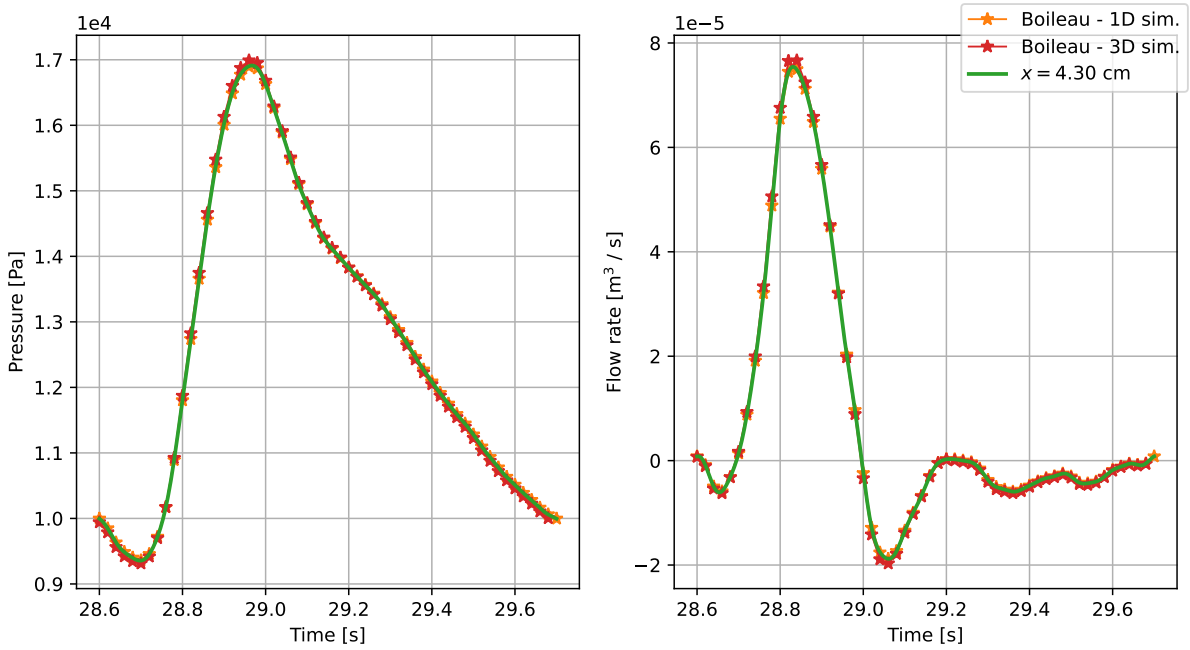


(a) *common carotid artery*.

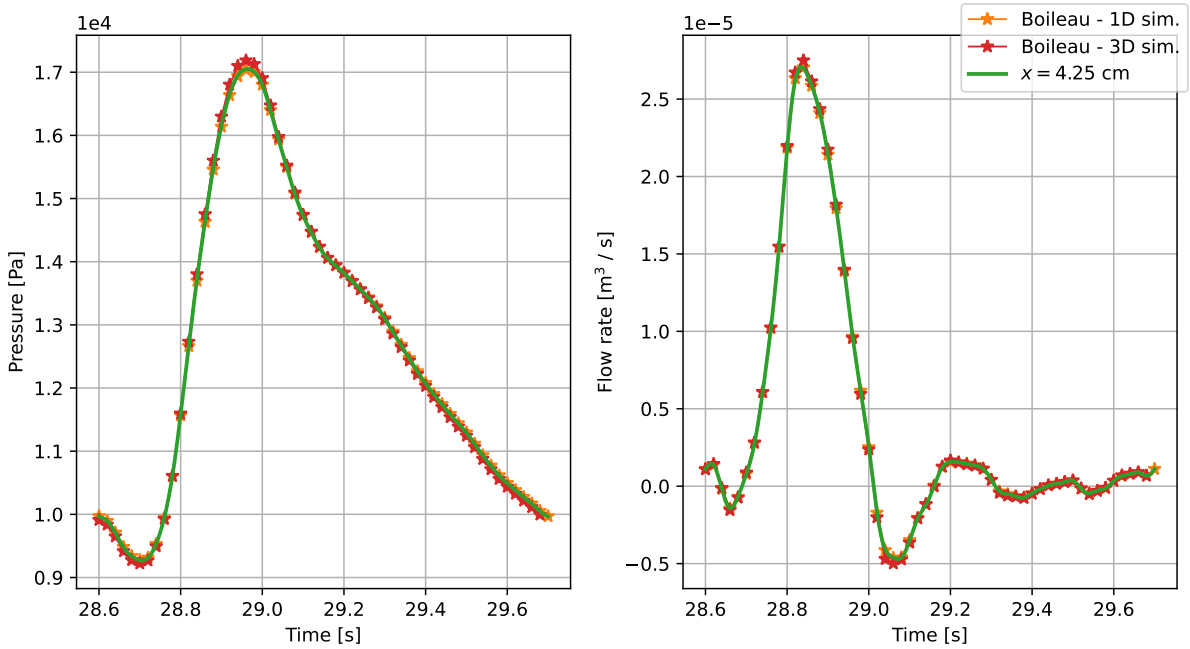


(b) *upper thoracic aorta*.

Figure 7.3: *common carotid artery* and *upper thoracic aorta* test cases, plotting the pressure and the flow rate at the middle point of the blood vessel during the last cardiac cycle. The 1D and 3D simulations taken from [5] are marked with the labels *Boileau - 1D sim.* and *Boileau - 3D sim.*



(a) *abdominal aorta.*



(b) *common iliac arteries.*

Figure 7.4: *aortic bifurcation* test case, plotting the pressure and the flow rate at the middle point of the *abdominal aorta* and *common iliac arteries* during the last cardiac cycle. The 1D and 3D simulations taken from [5] are marked with the labels *Boileau - 1D sim.* and *Boileau - 3D sim.*

7.2. Comparison between exact Riemann solver and Roe's approximate Riemann solver

In this section, we analyze the difference between the exact Riemann solver and Roe's linearization in terms of accuracy with respect to 3D simulations. We recall that the former can be employed only if we consider $\alpha = 1$, as extensively described in Sections 2.5–3.4.1; this assumption means that the velocity profile is considered flat when writing the non-linear momentum transport term from (2.2), which is physically inconsistent. On the other hand, Roe's approximate Riemann solver can be used for any value of the momentum flux correction factor (possibly $\alpha \neq 1$), by introducing a proper linearization and thus solving the Riemann problem (3.28) numerically, as explained in Section 3.4.2.

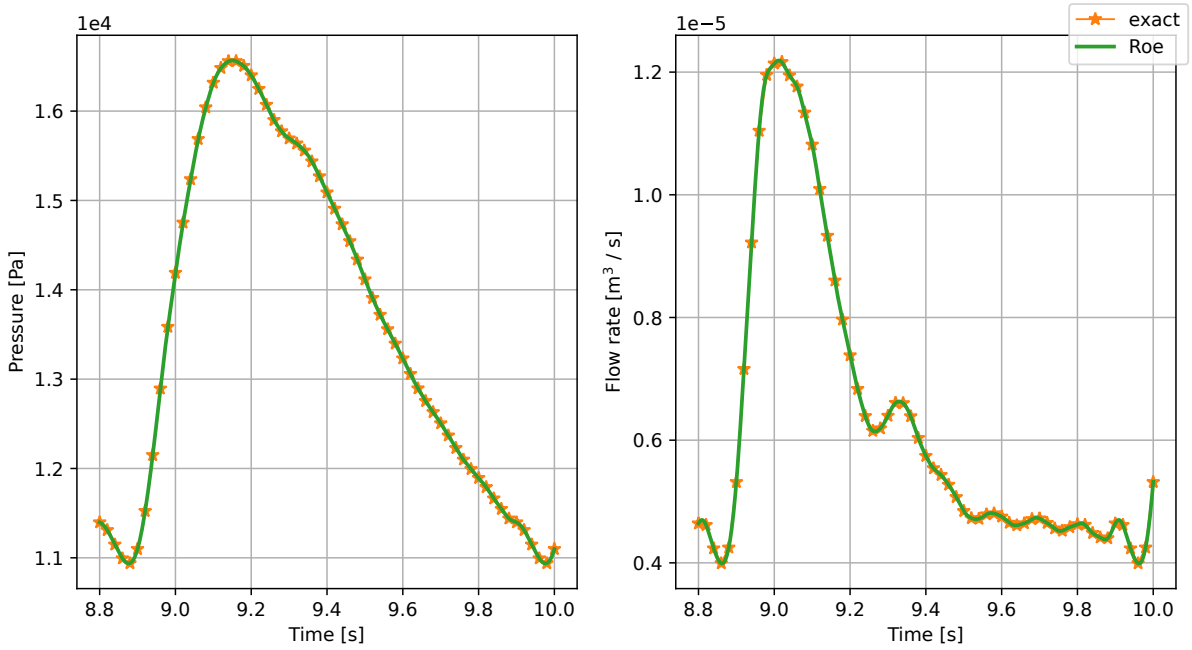
In order to carry out this comparison, we still employ *common carotid artery* and *upper thoracic aorta* test cases, whose parameters are listed in Table E.1, which have been already introduced in Section 7.1. These two test cases are solved by adopting either the exact Riemann solver (Section 3.4.1) or Roe's linearization (Section 3.4.2); the pressure and the flow rate at the middle of each vessel are plotted in Figure 7.5.

For a more quantitative comparison, we refer to Table 7.3, where we evaluate the absolute error $e_{\text{abs,3D}}$ and the relative error $e_{\text{rel,3D}}$ on the pressure with respect to the 3D simulations performed in [5] (whose definition are already reported in (7.9)). In both the *common carotid artery* test case (adopting a parabolic velocity profile) and the *upper thoracic aorta* test case (adopting a blunt velocity profile), $e_{\text{abs,3D}}$ and $e_{\text{rel,3D}}$ are greater when Roe's approximate Riemann solver is employed, meaning that the error induced by Roe's linearization is more relevant than the inconsistency introduced by considering $\alpha = 1$, which justifies the approach commonly used in literature (shown in Section 2.5).

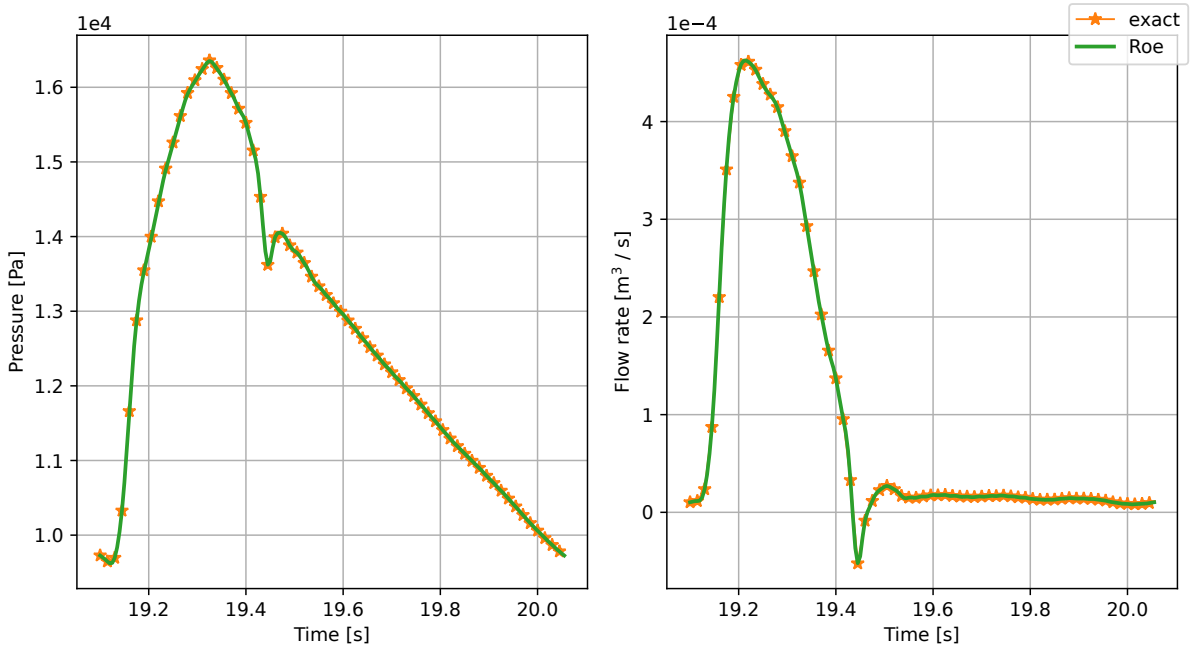
Error	Units	*common carotid artery (exact)*	*common carotid artery (Roe)*
$e_{\text{abs,3D}}$	Pa	92.98	102.2
$e_{\text{rel,3D}}$	–	0.6922 %	0.7590 %

Error	Units	*upper thoracic aorta (exact)*	*upper thoracic aorta (Roe)*
$e_{\text{abs,3D}}$	Pa	417.5	430.0
$e_{\text{rel,3D}}$	–	3.109 %	3.116 %

Table 7.3: Errors on pressure with respect to the results obtained in [5]. The subscript "rel" refers to relative errors, while the subscript "abs" refers to absolute errors. $e_{\text{abs,3D}}$ and $e_{\text{rel,3D}}$ are defined in (7.9). Errors are computed for both the simulations employing the exact Riemann solver (denoted by the label *exact*) and the ones using Roe's linearization (denoted by the label *Roe*).



(a) *common carotid artery*.



(b) *upper thoracic aorta*.

Figure 7.5: *common carotid artery* and *upper thoracic aorta* test cases, plotting the pressure and the flow rate at the middle point of the blood vessel during the last cardiac cycle. Results are obtained by either employing the exact Riemann solver (Section 3.4.1, denoted by the label *exact*) or Roe's approximate Riemann solver (Section 3.4.2, denoted by the label *Roe*).

7.3. Stented vessel

Here we present a modified version of the *common carotid artery* test case (which has been already introduced in Section 7.1, see Table E.1 for the parameters). In particular, we insert a 5-cm long stent in the *common carotid artery*, which is a device used to reinforce weakened or stenosed arterial segments. The stent increases the local vessel stiffness, which is represented in our model by an increased Young modulus in that portion of the blood vessel. This is also helpful to test the implementation of the discontinuous material properties, as described in Section 4.1.

More specifically, we employ the same parameters as *common carotid artery* test case (Table E.1) and we list only the parameters differing from that base case in Table 7.4. The stent is modeled as a 5-cm long region where the Young modulus E is increased 100 times, leading to the following piecewise constant expression:

$$E(x) = \begin{cases} 7 \times 10^5 \,\text{Pa} & \text{if } 0 \text{ cm} < x < 3.8 \text{ cm} \\ 7 \times 10^7 \,\text{Pa} & \text{if } 3.8 \text{ cm} < x < 8.8 \text{ cm} \\ 7 \times 10^5 \,\text{Pa} & \text{if } 8.8 \text{ cm} < x < 12.6 \text{ cm} \end{cases} \tag{7.10}$$

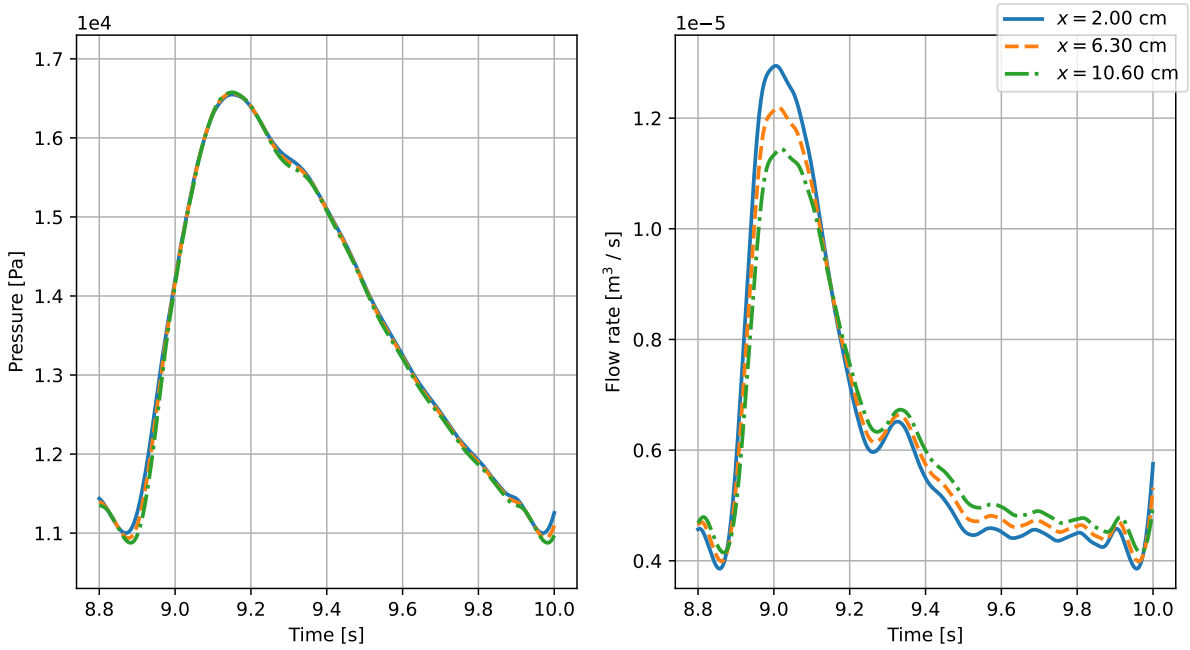
This implies that two discontinuous material points are introduced and so we need to solve the inverse Riemann problem (4.2) in two separate points (i.e. at $x = 3.8$ cm and at $x = 8.8$ cm). Indeed, the mesh is generated such that the nodes correspond to those discontinuous material points, i.e. the computational grid is composed of 4 nodes, located at $x = \{0, 3.8, 8.8, 12.6\}$ cm, for an average mesh spacing equal to 4.2 cm.

Compared to the *common carotid artery* base case, the reduced mesh spacing and the discontinuous material points produce greater numerical instability in terms of CFL criterion (3.19). That is to say, we need a lower time step (which is decreased to 5×10^{-5} s) and lower-degree polynomials ($p = 2$ when the stent is applied) to satisfy the CFL condition, while ensuring the convergence of Newton's solver applied to the inverse Riemann problem (4.2).

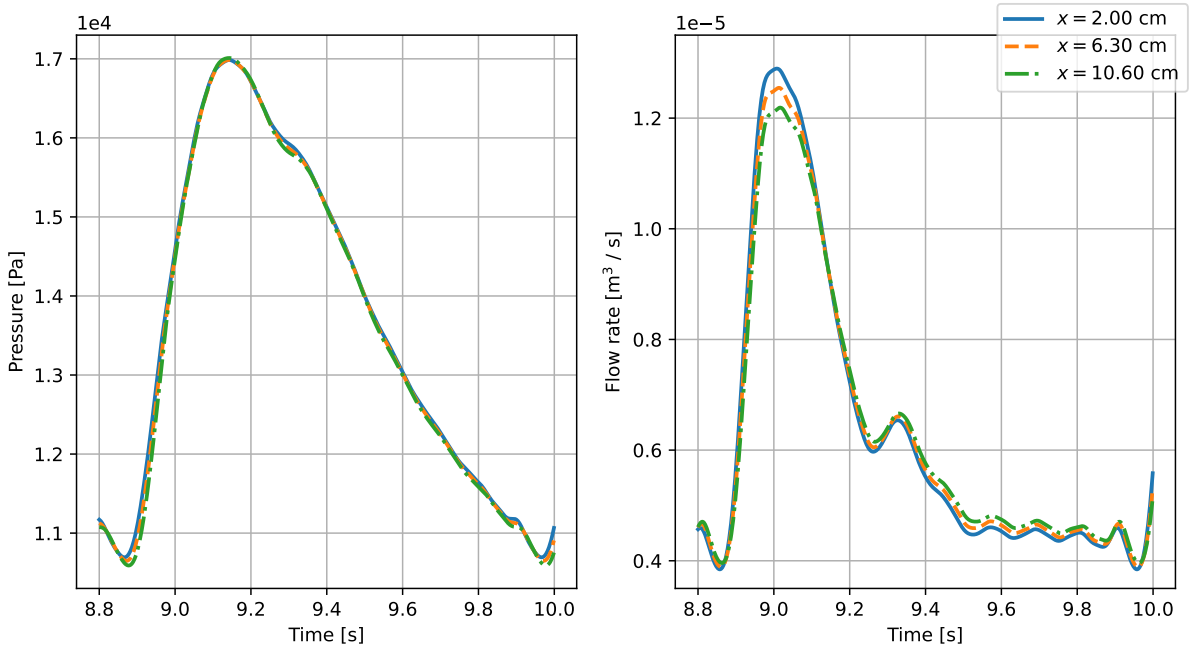
In Figure 7.6, the pressure and the flow rate at proximal ($x = 2$ cm), middle ($x = 6.3$ cm) and distal ($x = 10.6$ cm) points are plotted. We highlight the effect of the stent on the flow rate: the stent stiffens the vessel walls, resulting in a lower compliance and so in a reduced capability of storing blood mass. As a consequence, the flow rate is increased along the stented vessel, and this is more visible at the distal point, where the peak flow rate grows by 6.556 % when the stent is introduced (as shown in Figure 7.6b compared with Figure 7.6a).

Parameter	Units	Value
Number of cells M	–	3
Polynomial degree p	–	2
Time step Δt	s	5×10^{-5}
Young modulus E	Pa	$7 \times 10^5 \div 7 \times 10^7$

Table 7.4: Simulation setup for stented *common carotid artery* test case.



(a) *common carotid artery*, without stent.



(b) *common carotid artery*, with stent.

Figure 7.6: *common carotid artery* test case, plotting the pressure and the flow rate at $x = \{2, 6.3, 10.6\}$ cm during the last cardiac cycle.

7.4. 3D-1D coupling results

In order to test the 3D-1D coupling implementation (described in Chapter 5) and study wave reflections phenomena, we introduce a new test case, named *pressure Gaussian pulse* hereafter, whose parameters are listed in Table F.1. In this section, we will provide the results concerning the standalone simulations (adopting either the 3D model or the 1D model) and compare them with the 3D-1D coupled simulation.

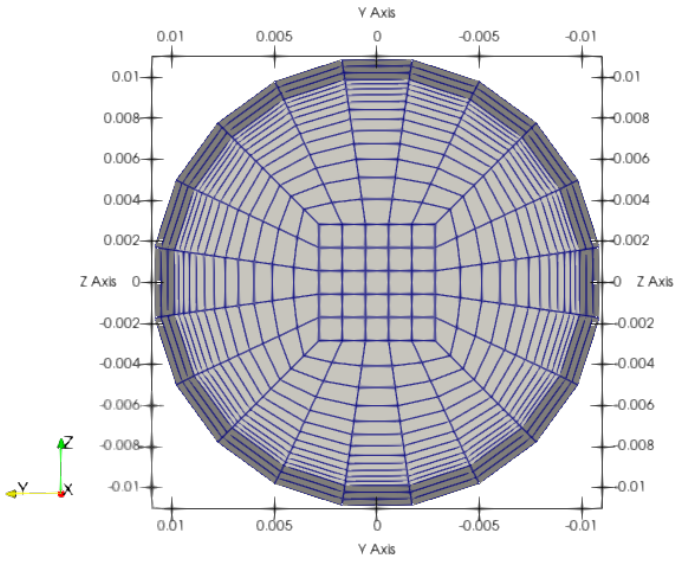
For each simulation contained in this section, we enforce a known pressure profile at the inlet, given by a Gaussian-shaped pulse (similar to (7.1)):

$$P_{\text{in}}(t) = P_{\max} \exp\left(-a\,(t - t_0)^2\right) \tag{7.11}$$

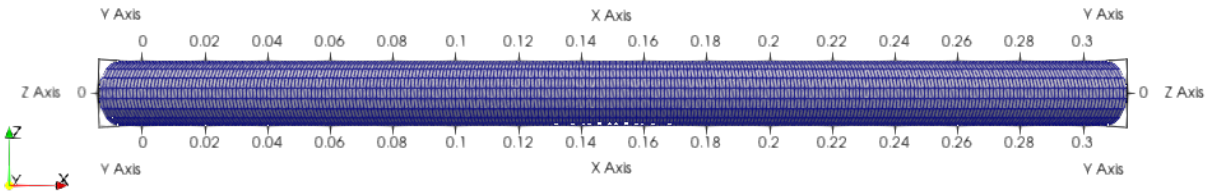
with a maximum pressure $P_{\max} = 2 \times 10^3\,\text{Pa}$, an attenuation factor $a = 10^4\,\text{s}^{-2}$ and a reference time $t = 0.05\,\text{s}$. Using the formula (7.3), we obtain a period equal to $T = 0.042\,92\,\text{s}$ and a wavelength approximately equal to $c_{\text{ref}}T = 0.2598\,\text{m}$, with $c_{\text{ref}} = c(A_{\text{ref}})$ being the reference wave speed (2.12). Even though this Gaussian-shaped profile is not representative of clinical data, it is useful to study the unphysical wave reflections induced by the 3D-1D coupling.

The inlet pressure (7.11) is prescribed using the inlet boundary condition described in Section 3.5.1 for the 1D haemodynamics model, while we employ the boundary condition (5.7)–(5.12) for the 3D FSI problem. The outlet is set to be non-reflecting, as provided in Section 3.5.2 for the 1D model and in (5.11)–(5.12) for the 3D problem. The 3D-1D coupling is achieved by means of the residual function (5.21) for the 1D problem and the coupling conditions (5.16)–(5.12) for the 3D problem. Furthermore, we adopt the Saint Venant-Kirchhoff constitutive model (5.5) within the structure subproblem.

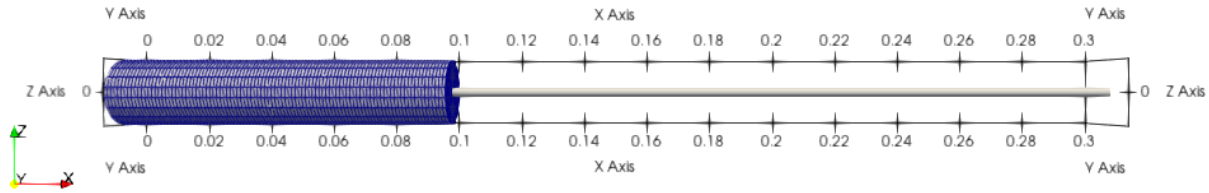
Since 3D FSI simulations are computationally expensive, we consider only a 30-cm long duct (Figure 7.7), in order to study the entire waveform while keeping the cost of the simulation under a reasonable threshold. In the standalone 3D simulation, we employ a structured mesh, with 79500 elements for the fluid mesh (which consists of a full cylinder) and 18000 elements for the structure mesh (which consists of a cylindrical shell), as illustrated in Figure 7.7. The discretization along the x-axis is achieved with elements of longitudinal width equal to 1 mm, in order to sample the wavelength correctly with 1^{st}-order polynomials (see Figure 7.7b). Moreover, the fluid mesh is refined towards the wall, in order to capture the steep velocity gradient at the walls (see Figure 7.7a). In the coupled simulation, the 3D cylindrical mesh is just 10-cm long, while the remaining 20-cm long portion of the domain is simulated using the 1D model, as depicted in Figure 7.7c.



(a) y-z plane. Both fluid mesh (light grey) and structure mesh (dark grey) are illustrated.



(b) x-z plane, standalone 3D simulation.



(c) x-z plane, coupled simulation.

Figure 7.7: Visualization of the mesh adopted in the *pressure Gaussian pulse* test case. The meshes employed for both the standalone and coupled simulations are reported.

Figures 7.10–7.11 show the pressure wave propagation on the x-z slice, evaluated at the time steps $t = \{0.06, 0.08, 0.10\}$ s. The pressure and flow rate waveforms for both the standalone and coupled simulations are analyzed more quantitatively in Figure 7.12, at the time steps $t = \{0.06, 0.08, 0.10\}$ s.

From Figure 7.12, it should be noticed that the 3D and 1D simulations exhibit a different dispersion and diffusion in the wave propagation. This is expected, since the Womersley number is equal to $\mathrm{Wo} = 62.00$ for this test case, so we adopt a blunt velocity profile within the 1D model; however, the blunt velocity profile presented in Section 2.1 is suitable for clinical case studies, where the maximum Womersley number is $\mathrm{Wo}_{\max} \approx 15$ (e.g. within the aorta or the vena cava). This produces a wrong prediction in the wall velocity gradient (and so in the wall shear stress), which in turn produces lower pressure losses in the standalone 1D simulation, as shown in Figure 7.8. On the other hand, the objective of this section is not to verify the 1D model against the 3D FSI model (as already done in Section 7.1), but rather to study the coupling between them, so we deem this discrepancy to be acceptable.

It is noteworthy that the approximation (5.11) implies that the flow rate is evaluated at the previous time step when implementing the non-reflecting boundary condition on the 3D FSI model. As a consequence, some spurious wave reflections traveling backward in the duct arise, and unphysical backflow is observed when these artificial waves reach the inlet, as it is visible at $t = 0.08\,\mathrm{s}$ (Figure 7.12b). Similar considerations follow for the coupled simulation, where the coupling condition (5.16) generates some numerical artifacts in the 3D portion of the problem, visible especially at $t = 0.08\,\mathrm{s}$ (Figure 7.12b); yet, these spurious wave reflections are bounded and decay over time, as we can highlight by comparing the results at $t = 0.08\,\mathrm{s}$ (Figure 7.12b) to the ones at $t = 0.10\,\mathrm{s}$ (Figure 7.12c).

We can proceed to assess the impact of the explicit coupling (i.e. the coupling conditions are solved without any correction sub-iteration by using the quantities from the 3D model at the previous time step (5.21)). For this purpose, Figure 7.9 shows the error between pressure and flow rate predicted by the 3D model and the ones predicted by the 1D model at the coupling interface $\Gamma_{\text{3D-1D}}$, formally:

$$e_{P,\text{3D-1D}} := (P_{\text{3D}} - P_{\text{1D}})\Big|_{\Gamma_{\text{3D-1D}}}, \qquad e_{Q,\text{3D-1D}} := (Q_{\text{3D}} - Q_{\text{1D}})\Big|_{\Gamma_{\text{3D-1D}}}. \tag{7.12}$$

Indeed, $e_{P,\text{3D-1D}}$ and $e_{Q,\text{3D-1D}}$ follow a similar pattern over time and their oscillations are bounded, as predicted by the stability results in [29]. Yet, it is noteworthy that the amplitudes of these errors differ. Indeed, $e_{P,\text{3D-1D}}$ exhibits a maximum amplitude equal to $4.452 \times 10^{-1}\,\mathrm{Pa}$, which is just $0.022\,26\,\%$ of the maximum pressure wave amplitude; this

result is achieved by means of the coupling condition (5.16), which guarantees the continuity of the pressure between the two models. On the other hand, $e_{Q,\text{3D-1D}}$ has a maximum amplitude equal to $5.757 \times 10^{-6}\,\text{m}^3/\text{s}$, corresponding to 5.573 % of the maximum flow rate amplitude; this higher error is due to the fact that the coupling conditions (5.16)–(5.21) do not enforce the continuity of the flow rate directly on the coupling interface $\Gamma_{\text{3D-1D}}$.

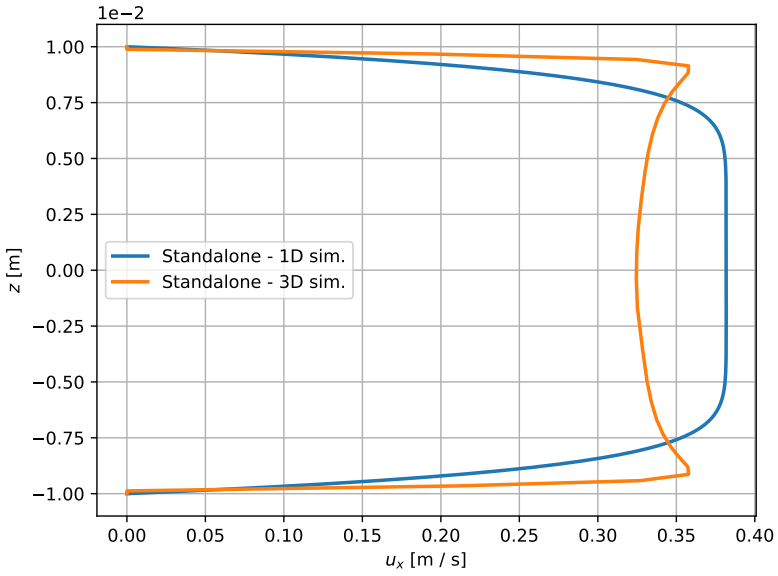


Figure 7.8: Inlet velocity profile predicted by the 1D model and the 3D model at $t = 0.05\,\text{s}$.

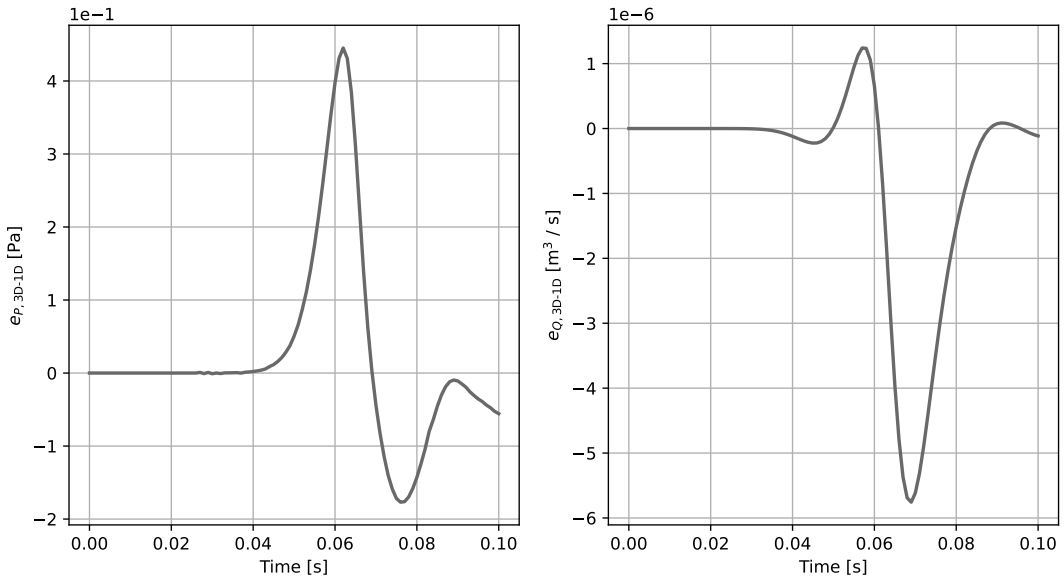
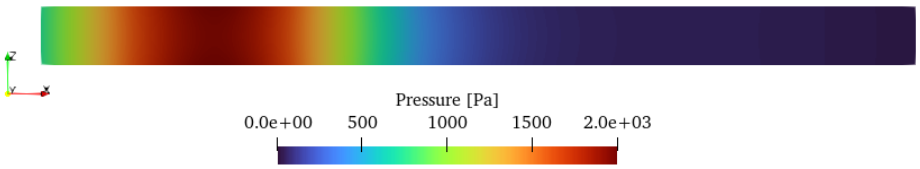
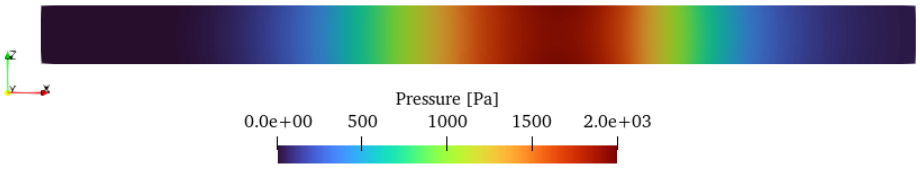


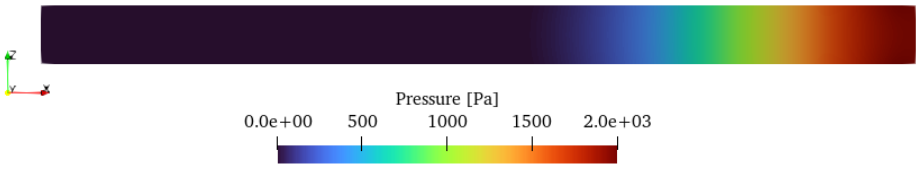
Figure 7.9: Difference between pressure and flow rate predicted by the 3D model and the ones predicted by the 1D model at the coupling interface $\Gamma_{\text{3D-1D}}$. $e_{P,\text{3D-1D}}$ and $e_{Q,\text{3D-1D}}$ are defined in (7.12).



(a) $t = 0.06\,\mathrm{s}$.

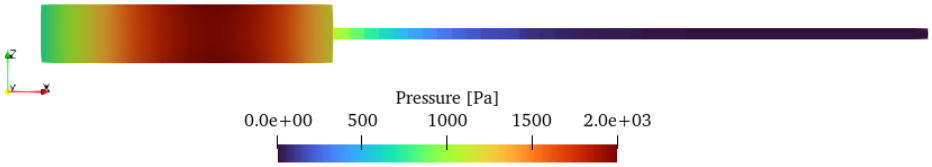


(b) $t = 0.08\,\mathrm{s}$.

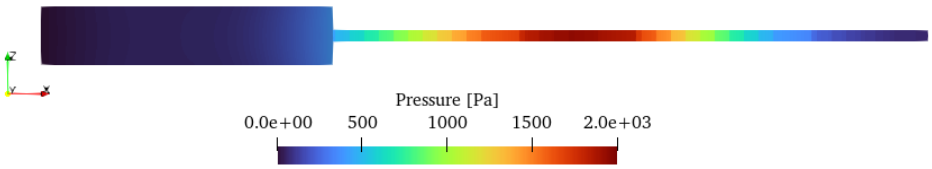


(c) $t = 0.10\,\mathrm{s}$.

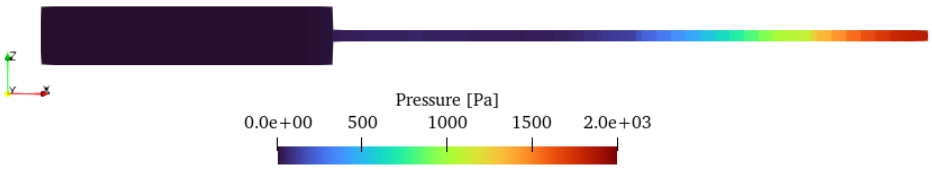
Figure 7.10: *pressure Gaussian pulse* test case, showing the pressure wave propagation in the standalone 3D simulation. x-z slices are plotted at different time steps.



(a) $t = 0.06\,\mathrm{s}$.

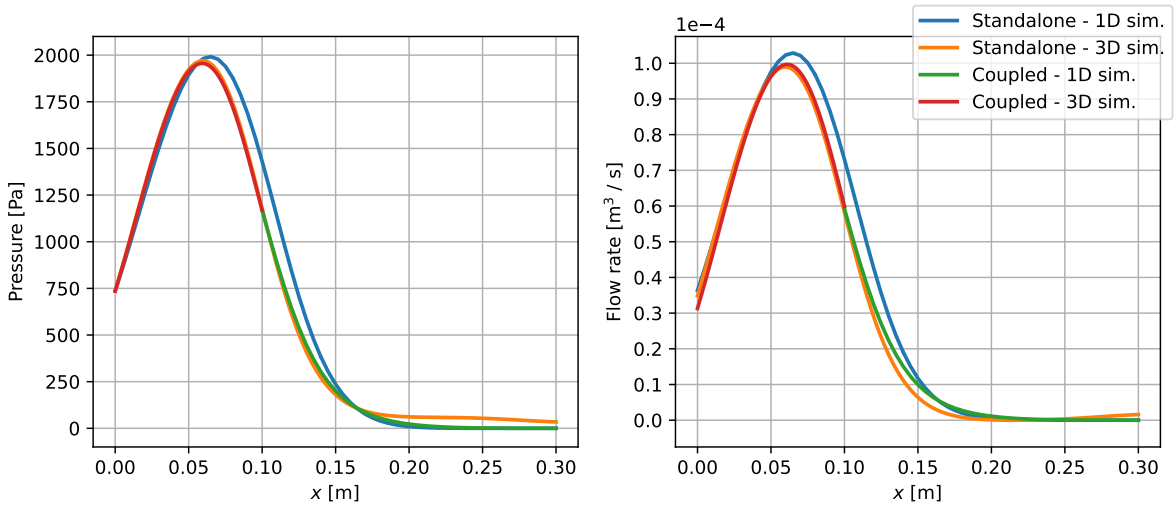


(b) $t = 0.08\,\mathrm{s}$.

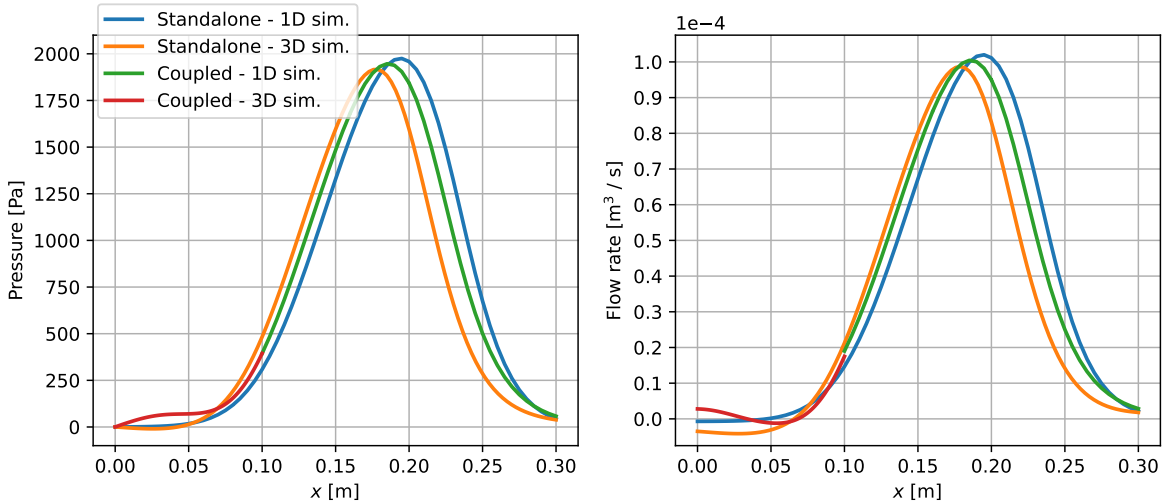


(c) $t = 0.10\,\mathrm{s}$.

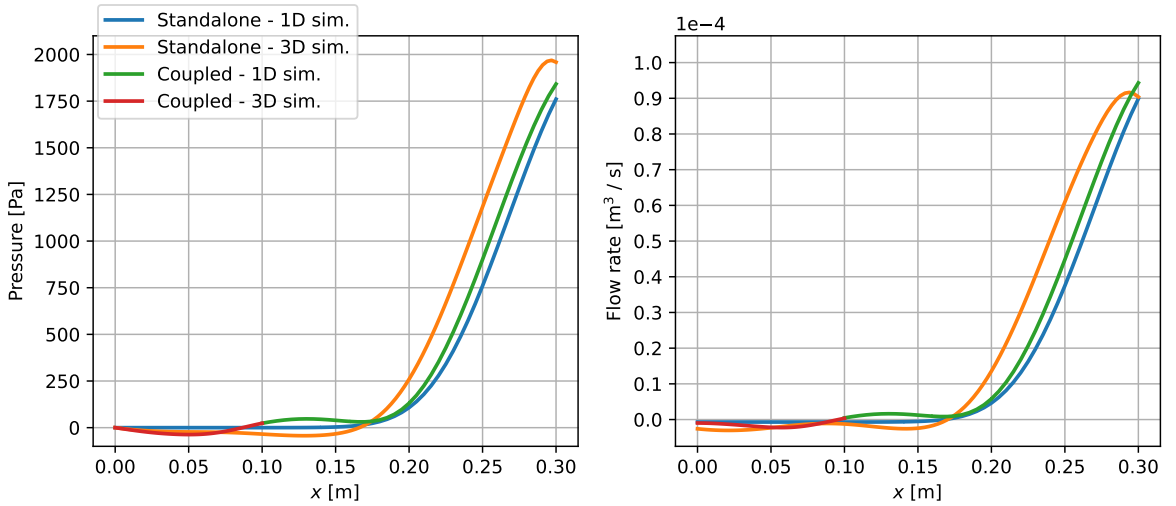
Figure 7.11: *pressure Gaussian pulse* test case, showing the pressure wave propagation in the 3D-1D coupled simulation. x-z slices are plotted at different time steps.



(a) $t = 0.06\,\mathrm{s}$.



(b) $t = 0.08\,\mathrm{s}$.



(c) $t = 0.10\,\mathrm{s}$.

Figure 7.12: *pressure Gaussian pulse* test case, showing the Gaussian-shaped waveform along the duct at different time steps. Standalone and coupled simulations are plotted.

7.5. Convergence analysis

The purpose of this section is to analyze the h-, p- and time convergence (i.e. the convergence with respect to the mesh spacing h, the polynomial degree p and the time step Δt), while proving a sufficient consistence between the inviscid implementation (employing the exact Riemann solver introduced in Section 3.4.1) and the viscous implementation (employing Roe's approximate Riemann solver, defined in Section 3.4.2).

In order to assess the rate of convergence and compute the L^2-error norms (later defined in (7.21)), we need an exact solution $\mathbf{U}_{\text{ex}}(x,t) = (A_{\text{ex}}(x,t), Q_{\text{ex}}(x,t))^\top$ to be compared with the numerical approximation $\mathbf{U}_h(x,t) = (A_h(x,t), Q_h(x,t))^\top$. To find a suitable expression for $\mathbf{U}_{\text{ex}}(x,t)$, under the hypothesis of small perturbations, we can linearize the governing equations (2.2)–(2.3) around the tube law reference state $(A_{\text{ref}}, P_{\text{ref}})$ and, by substituting $(A, Q, P) \approx (A_{\text{ref}} + a', q', P_{\text{ref}} + p')$, it is possible to derive the following linearized system, defined on the 1D spatial domain Ω:

$$\begin{cases} \dfrac{\partial a'}{\partial t} + \dfrac{\partial q'}{\partial x} = 0 & \forall x \in \Omega = (x_L, x_R),\ \forall t \in (0, t_f] \\ L_{1\text{D}} \dfrac{\partial q'}{\partial t} + \dfrac{\partial p'}{\partial x} + R_{1\text{D}} q' = 0 & \forall x \in \Omega = (x_L, x_R),\ \forall t \in (0, t_f] \\ p' = \dfrac{a'}{C_{1\text{D}}} \end{cases} \tag{7.13}$$

where the equivalent inductance, compliance and resistance have the following constant expressions:

$$L_{1\text{D}} = \frac{\rho}{A_{\text{ref}}}\,, \qquad C_{1\text{D}} = \frac{A_{\text{ref}}}{\rho c_{\text{ref}}^2}\,, \qquad R_{1\text{D}} = \frac{\rho K_R}{A_{\text{ref}}^2}\,. \tag{7.14}$$

After that, we can assume that the perturbations a' and q' can be decomposed into a superimposition of elementary periodic waves. For simplicity, given the linearity of (7.13), we can consider only one of these elementary periodic waves, thus representing a' and q' in the complex plane:

$$a'(x,t) = \widehat{a} e^{i(\omega t - kx)}\,, \qquad q'(x,t) = \widehat{q} e^{i(\omega t - kx)}\,, \tag{7.15}$$

with $\widehat{a}$ and $\widehat{q}$ being the wave amplitudes, $i = \sqrt{-1}$, ω the angular frequency and k the wave number. By substituting (7.15) into (7.13), we can find the following relation $k = k(\omega)$:

$$L_{1\text{D}} C_{1\text{D}} \omega^2 - k^2 - i R_{1\text{D}} C_{1\text{D}} \omega = 0\,. \tag{7.16}$$

Then, we can solve (7.16) for the wave number k and, for simplicity, we select the sign of

k representing a right-running wave:

$$k = \sqrt{C_{1D}\omega\left(L_{1D}\omega - iR_{1D}\right)}\,. \tag{7.17}$$

Similarly, the relation $\widehat{a} = \widehat{a}(\widehat{q})$ can be obtained by substituting (7.15) into the first equation of (7.13), yielding:

$$\widehat{a} = \frac{k}{\omega}\widehat{q}\,. \tag{7.18}$$

Finally, the physical solution of (7.13), denoted by $\widetilde{\mathbf{U}}_{\text{ex}}(x,t)$, can be expressed as the real part of (7.15), i.e.:

$$\widetilde{\mathbf{U}}_{\text{ex}}(x,t) = \begin{bmatrix} \widetilde{A}_{\text{ex}}(x,t) \\ \widetilde{Q}_{\text{ex}}(x,t) \end{bmatrix} = \begin{bmatrix} A_{\text{ref}} + \Re\left\{\widehat{a}e^{i(\omega t - kx)}\right\} \\ \Re\left\{\widehat{q}e^{i(\omega t - kx)}\right\} \end{bmatrix}\,. \tag{7.19}$$

This procedure has been devised and applied successfully in [18].

In order to perform the convergence analysis, we introduce two new test cases, named *inviscid sine wave* and *viscous sine wave* hereafter. These test cases have the same parameters as *inviscid Gaussian pulse* and *viscous Gaussian pulse* (already reported in Table E.1), except for the inflow $Q_{\text{in}}(t)$, which consists of one single sine wave:

$$Q_{\text{in}}(t) = \widehat{Q}\sin\left(\frac{2\pi}{T}t\right) \tag{7.20}$$

with $\widehat{Q} = 10^{-6}\,\text{m}^3/\text{s}$ being the sine wave amplitude and $T = 0.1\,\text{s}$ its period. This inflow leads to the trivial choice of the angular frequency $\omega = \dfrac{2\pi}{T}$ and the complex amplitude $\widehat{q} = -i\widehat{Q}$. The wave number k and the cross-sectional area amplitude $\widehat{a}$ follow from (7.17) and (7.18), respectively. In addition, the final time is increased up to $t_f = 2.0\,\text{s}$, in order to let the wave reach the outlet and so study the fully-developed flow (without any residual dependence from the initial conditions). This exact solution is plotted in Figure 7.13 at the final time $t_f = 2.0\,\text{s}$, where we highlight the exponential decay of the pressure wave when viscous effects are taken into consideration.

We remark that $\widetilde{\mathbf{U}}_{\text{ex}}(x,t)$ (7.19) is formally the exact solution of the linearized problem (7.13). However, under the hypothesis of small perturbations, $\widetilde{\mathbf{U}}_{\text{ex}}(x,t)$ may be considered a good approximation of the exact solution to the original non-linear problem (2.2)–(2.3), denoted by $\mathbf{U}_{\text{ex}}(x,t)$. In particular, the sine wave boundary condition for the inflow (7.20), together with the selected values for the inflow amplitude $\widehat{Q}$ and for the period T, leads to

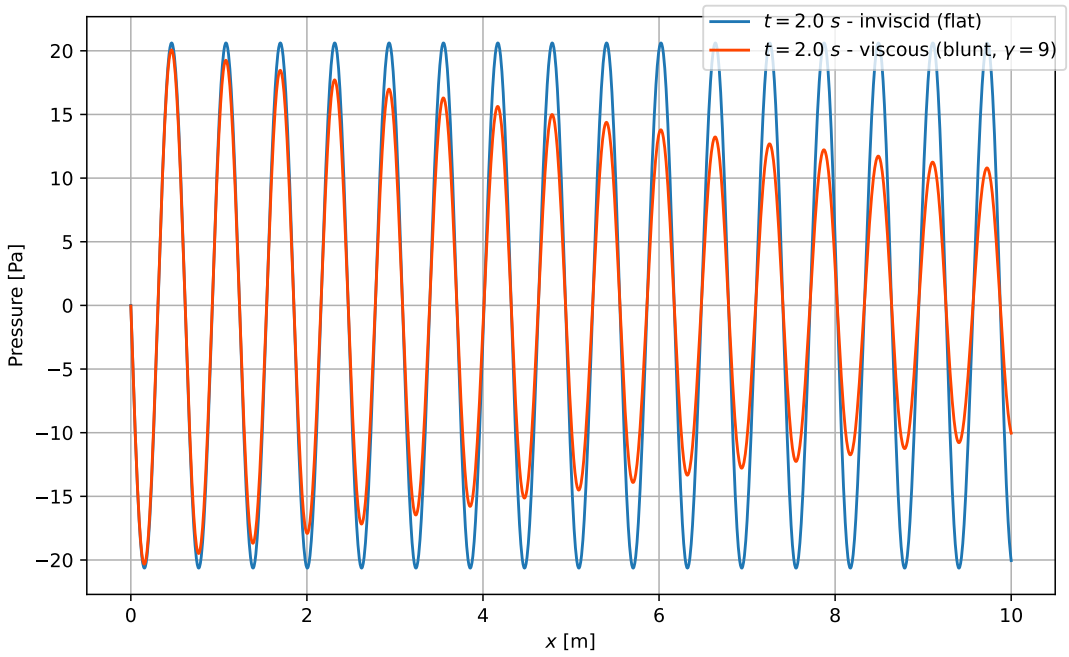


Figure 7.13: Pressure along the duct at $t = 2.0\,\mathrm{s}$ for *inviscid sine wave* and *viscous sine wave* test cases.

perturbations $a'(x,t)$ and $q'(x,t)$ whose maximum amplitudes are $|\widehat{a}| = 1.620 \times 10^{-7}\,\mathrm{m}^2$ and $|\widehat{q}| = 10^{-6}\,\mathrm{m}^3/\mathrm{s}$. It should be noticed that $|\widehat{a}|$ represents only the $0.051\,57\,\%$ of the reference cross-sectional area A_{ref}; consequently, the hypothesis of small perturbations is verified and we can assume that $\mathbf{U}_{\mathrm{ex}}(x,t) \approx \widetilde{\mathbf{U}}_{\mathrm{ex}}(x,t)$.

Throughout this section, the convergence rate is measured using the following L^2-error norms, computed with respect to $\mathbf{U}_{\mathrm{ex}}(x,t) = (A_{\mathrm{ex}}(x,t), Q_{\mathrm{ex}}(x,t))^\top \approx \widetilde{\mathbf{U}}_{\mathrm{ex}}(x,t)$ and evaluated at the final time $t_f = 2.0\,\mathrm{s}$:

$$
\begin{aligned}
e_A &:= \left\| A_{\mathrm{ex}}|_{t=t_f} - A_h|_{t=t_f} \right\|_{L^2} = \sqrt{\int_\Omega \left(A_{\mathrm{ex}}|_{t=t_f} - A_h|_{t=t_f} \right)^2 \mathrm{d}x}\,, \\
e_Q &:= \left\| Q_{\mathrm{ex}}|_{t=t_f} - Q_h|_{t=t_f} \right\|_{L^2} = \sqrt{\int_\Omega \left(Q_{\mathrm{ex}}|_{t=t_f} - Q_h|_{t=t_f} \right)^2 \mathrm{d}x}\,.
\end{aligned}
\tag{7.21}
$$

As a final remark, we highlight that the L^2-error norms e_A and e_Q are influenced by three sources of numerical errors:

1. the space discretization, represented by both the mesh spacing h and the polynomial degree p;
2. the time discretization, given by the selected time step Δt and the order of the AB

scheme s;

3. the iterative solution of the inverse Riemann problems (3.44)–(4.2)–(4.6) by means of Newton's method.

When quantifying one error source, the others must remain below an acceptable threshold to avoid degrading the convergence rate. Indeed, the source of error 3 becomes negligible with respect to space and time discretization errors under an appropriate setup of Newton's solver tolerance. On the other hand, when assessing the space convergence, we should ensure that the time-discretization induced error has a lower order of magnitude (e.g. by reducing the time step Δt properly); conversely, when measuring the time discretization, the space-discretization induced error should be negligible (e.g. by choosing a sufficiently small mesh spacing h).

7.5.1. h-convergence

We proceed to study the h-convergence for *inviscid sine wave* and *viscous sine wave* test cases. As stated before, we use the same parameters as *inviscid Gaussian pulse* and *viscous Gaussian pulse* test cases. Table 7.5 reports only the parameters differing from Table E.1.

For DG methods, the theoretical maximum order of h-convergence is $p+1$ in terms of L^2-error norm in the most general case. However, when the exact solution is such that $u(x,t) \in W^{p+3,\infty}(\Omega),\ \forall x \in \Omega = (x_L, x_R),\ \forall t \in (0, t_f]$, with $W^{p+3,\infty}(\Omega)$ being a Sobolev space, and the mesh is sufficiently uniform, it can be proven that the theoretical convergence rate increases up to $p+2$ for DG methods applied to 1D non-linear hyperbolic equations. For a more theoretical analysis, we refer to Theorem 3 in [16]. It should be noticed that this theorem is derived for 1D non-linear scalar hyperbolic equations, whereas we are analyzing the system of non-linear hyperbolic equations (2.2), defined on a 1D domain. However, numerical evidence shows a convergence rate equal to 4 when $p = 2$ (Figure 7.14). This phenomenon is often referred to as *super-convergence* in the literature [16].

In particular, Figure 7.14 exhibits the following regimes, sorted below by the number of cells M:

1. $M \in [10, 20]$ (i.e. $h \in [0.5, 1.0]$ m): it is the rightmost portion of Figure 7.14, showing a plateau. It is noteworthy that the selected period of the inflow sine wave (given by (7.20)) corresponds to a wavelength $c_{\text{ref}}T = 0.6172\,\text{m}$ (with c_{ref} computed in (7.2)). Hence, the L^2-error norms do not decrease even with further mesh refinements when

$h \gtrsim c_{\text{ref}}T$. Indeed, to capture the waveforms correctly, we should employ either a finer mesh (i.e. $h \ll c_{\text{ref}}T$) or higher-degree polynomials (i.e. $p > 2$).

2. $M \in [50, 70]$ (i.e. $h \in [0.143, 0.2]$ m): a super-convergent regime is reached, with a maximum convergence rate equal to 4.

3. $M \in [100, 500]$ (i.e. $h \in [0.02, 0.1]$ m): it is the leftmost portion of Figure 7.14, where the L^2-error norms remain constant even when the mesh is refined. This behavior occurs since the time-discretization errors are not negligible with respect to the space-discretization errors; that is to say, we should employ a lower time step to further reduce the time-discretization errors, thus reaching an optimal convergence rate even for $M > 100$. We will refer to this phenomenon as *error saturation* hereafter.

Parameter	**Units**	**Value**
Vessel length L	m	10
Number of cells M	–	$\{10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 250, 500\}$
Polynomial degree p	–	2
Final time t_f	s	2.0
Time step Δt	s	10^{-4}

Table 7.5: Simulation setup for h-convergence analysis.

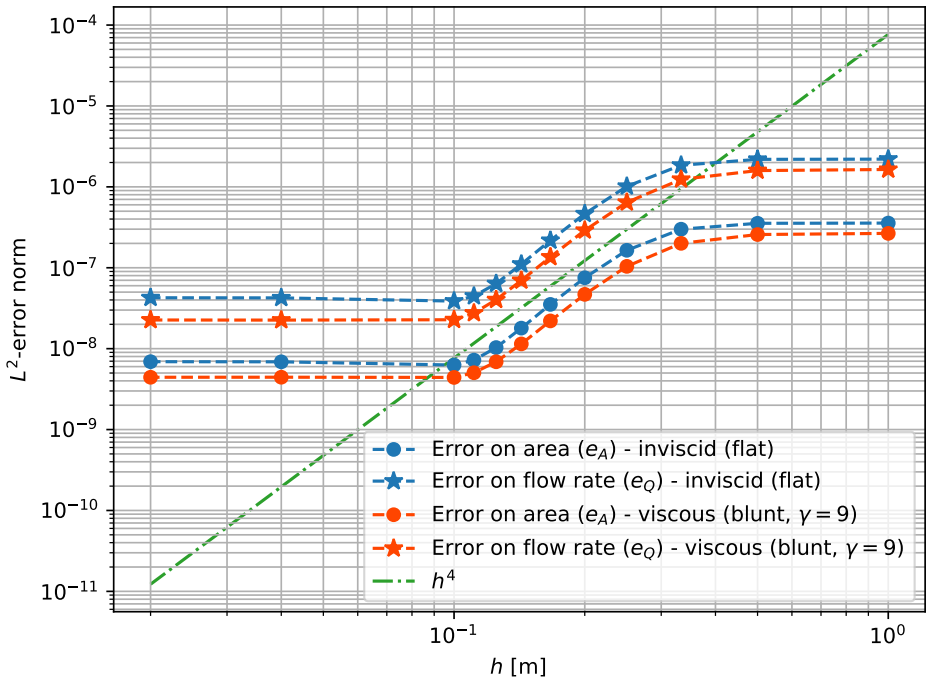


Figure 7.14: h-convergence of L^2-error norms in log-log scale. The L^2-error norms e_A and e_Q are defined in (7.21). Results are plotted at time $t = 2.0\,\text{s}$, with a polynomial degree equal to $p = 2$. Both *inviscid sine wave* and *viscous sine wave* test cases are reported.

7.5.2. p-convergence

The p-convergence analysis is presented in this section. As done for the h-convergence (see Section 7.5.1), we report in Table 7.6 only the differences with respect to the *inviscid Gaussian pulse* and *viscous Gaussian pulse* parameters (already listed in Table E.1).

Similarly to h-convergence, also p-convergence exhibits three regimes, as we can observe in Figure 7.15:

1. $p \in [1, 4]$ (leftmost part of Figure 7.15): the L^2-error norms remain constant even with an increasing polynomial degree. This behavior is strictly related to mesh size ($h = 1.0\,\text{m}$ for the p-convergence analysis). Indeed, lower-order polynomials do not capture the waveform correctly if the mesh size is greater than the wavelength (i.e. when $h \gtrsim c_{\text{ref}} T = 0.6172\,\text{m}$).

2. $p \in [6, 8]$: the L^2-error norms converge exponentially with respect to the polynomial degree p, as it is predicted by the spectral/hp element methods theory.

3. $p \in [9, 10]$ (rightmost part of Figure 7.15): here, we can notice the L^2-error norms saturation due to the selected time step. Thus, in order to obtain a further reduction

of the L^2-error norms in this region, we should employ a lower time step (as already explained in Section 7.5.1).

Parameter	Units	Value
Vessel length L	m	10
Number of cells M	–	10
Polynomial degree p	–	$\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$
Final time t_f	s	2.0
Time step Δt	s	10^{-4}

Table 7.6: Simulation setup for p-convergence analysis.

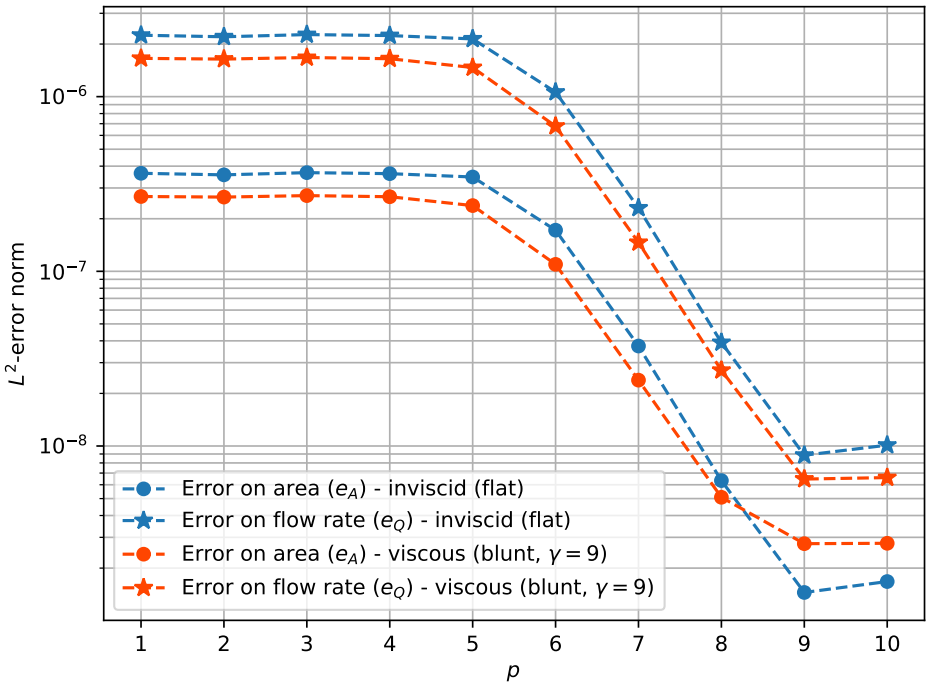


Figure 7.15: p-convergence of L^2-error norms in semilog-y scale. The L^2-error norms e_A and e_Q are defined in (7.21). Results are plotted at time $t = 2.0\,\text{s}$, with a a mesh spacing equal to $h = 1.0\,\text{m}$. Both *inviscid sine wave* and *viscous sine wave* test cases are reported.

7.5.3. Time convergence

Concerning the time convergence, we report in Table 7.7 the parameters differing from the *inviscid Gaussian pulse* and *viscous Gaussian pulse* base cases. We should address the

following limitations while assessing the time convergence rate (whose theoretical value is equal to s, with s being the order of the AB scheme):

1. only time steps satisfying the CFL condition (3.19) can be employed, otherwise the AB time-advancing scheme would be numerically unstable, leading to divergence of the inverse Riemann problem (3.44) and/or to unphysical results;
2. we should use sufficiently large time steps, in order to avoid error saturation phenomena and thus measure the maximum convergence rate correctly.

These two conditions are conflicting, since condition 1 would require sufficiently small time steps (i.e. $\Delta t < \Delta t_{\max}$, as defined in (3.19)), while condition 2 would require larger time steps (i.e. $\Delta t > \Delta t_{\text{opt}}$, where Δt_{opt} is the minimum time step guaranteeing the optimal convergence rate). However, when applying explicit schemes to stiff problems, it may occur that $\Delta t_{\text{opt}} > \Delta t_{\max}$, hence we are not able to correctly evaluate the order of convergence while ensuring numerical stability. We refer to [20] for a theoretical perspective about stiff problems coupled with explicit time-stepping schemes.

The latter behavior can be observed in Figure 7.16, where the maximum order of convergence is equal to 1, even though the selected order of the AB scheme is 2. Indeed, we should employ a higher time step to avoid the error saturation and reach the theoretical convergence rate, but this is not possible due to the stability requirement $\Delta t < \Delta t_{\max}$ (3.19).

Parameter	**Units**	**Value**
Vessel length L	m	10
Number of cells M	–	500
Polynomial degree p	–	1
Final time t_f	s	2.0
Time step Δt	s	$\{10^{-4}, 2 \times 10^{-4}, 3.333 \times 10^{-4}, 5 \times 10^{-4}\}$

Table 7.7: Simulation setup for time convergence analysis.

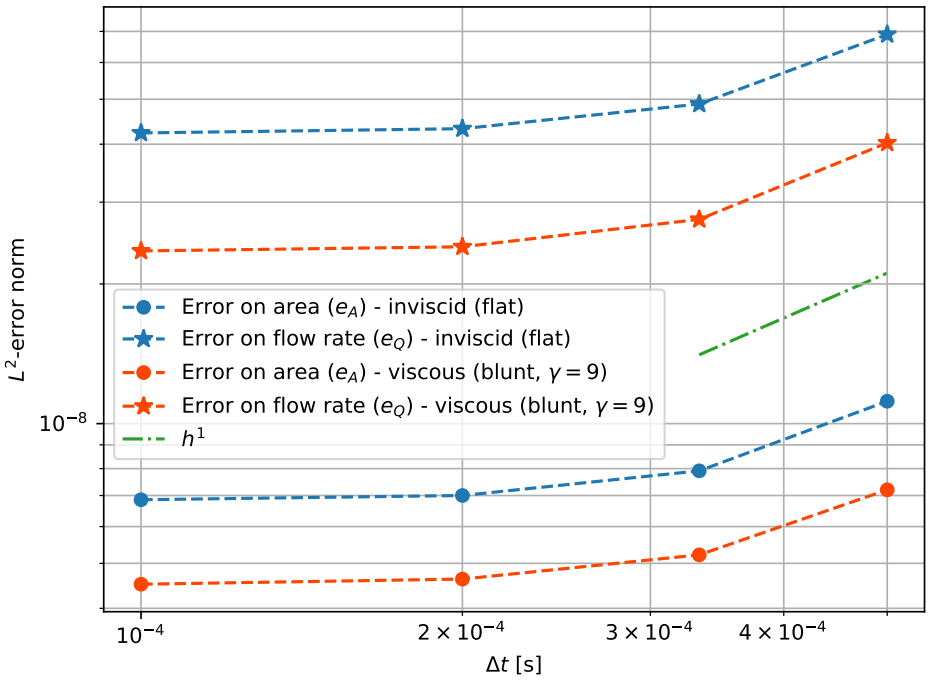


Figure 7.16: Time convergence of L^2-error norms in log-log scale. The L^2-error norms e_A and e_Q are defined in (7.21). Results are plotted at time $t = 2.0\,\text{s}$, with a a mesh spacing equal to $h = 1.0\,\text{m}$. Both *inviscid sine wave* and *viscous sine wave* test cases are reported.

7.6. Parallel scalability

In this section, we show the results concerning the parallel scalability. For this purpose, we still employ the *inviscid sine wave* and *viscous sine wave* test cases (introduced in Section 7.5). As already done in Sections 7.5.1–7.5.2–7.5.3, we list in Table 7.8 only the difference with respect to the *inviscid Gaussian pulse* and *viscous Gaussian pulse* base cases. All the results reported below are obtained on a CPU AMD Ryzen 5 4600H (3GHz, 6 cores, 12 threads).

In order to test the parallelization, the mesh size h is set to $2 \times 10^{-2}\,\text{m}$ and the polynomial degree is chosen as $p = 4$, hence leading to 5000 degrees of freedom (2500 DoFs for the cross-sectional area A_h and 2500 DoFs for the flow rate Q_h). From Figure 7.17, it can be noticed that the maximum speed-up (reached with 4 cores) is equal to ~ 2.8x, associated with an efficiency of $\sim 70\,\%$. The following limitations prevent the algorithm from scaling linearly on multiple cores.

- Even if the right-hand side assembly (3.13) is perfectly parallelizable through `deal.II`'s

distributed triangulations[7.1], and the same holds true for the AB update rule (3.18), these operations are not particularly expensive (the benchmark presented here completes in $\sim 20\,\mathrm{s}$ in sequential mode). Therefore, the overhead introduced by the ghost entries update is non-negligible with respect to the cost of the entire algorithm (the update of the ghost entries represents $\sim 7.5\,\%$ of the total wall-clock time with 4 cores).

- Newton's method needed to solve the inverse Riemann problem (3.44) is employed twice per time step (i.e. we need to solve (3.44) for each endpoint of the blood vessel). By exploiting `deal.II`'s distributed triangulations, we can assign each endpoint to one `MPI` rank. Therefore, in the case of 2 parallel processes, this is particularly advantageous, since each process handles one and only one Newton's solver. However, when the number of `MPI` ranks is greater or equal to 3, we can use only 2 parallel processes to solve the non-linear systems given by (3.44), while we are forced to keep the remaining processes in an idle state. It is worth noting that a complex blood vessel network notably increases the number of inverse Riemann problems – given by the boundary conditions (3.44), the discontinuous material properties (4.2), and the bifurcations (4.6); consequently, the parallelization across multiple `MPI` ranks becomes more convenient in more realistic/clinical scenarios.
- `deal.II` authors have proven a perfect scalability for a number of degrees of freedom per core $\gtrsim 10^5$ (as shown in [35]); however, this number of degrees of freedom would be too large for the problem at hand and would not reflect a real scenario.
- The benchmark presented here has been obtained on a laptop computer, with the corresponding hardware limitations, such as reduced CPU clock speed, RAM bandwidth and cache size, cooling capabilities, etc.

We finally remark that Figure 7.17 exhibits a good agreement in timings between the inviscid simulations (employing an exact Riemann solver, as described in Section 3.4.1) and viscous simulations (employing Roe's approximate Riemann solver, as explained in Section 3.4.2).

[7.1] https://dealii.org/current/doxygen/deal.II/group__distributed.html

Parameter	Units	Value
Vessel length L	m	10
Number of cells M	–	500
Polynomial degree p	–	4
Final time t_f	s	2.0
Time step Δt	s	10^{-4}

Table 7.8: Simulation setup for parallel scalability analysis.

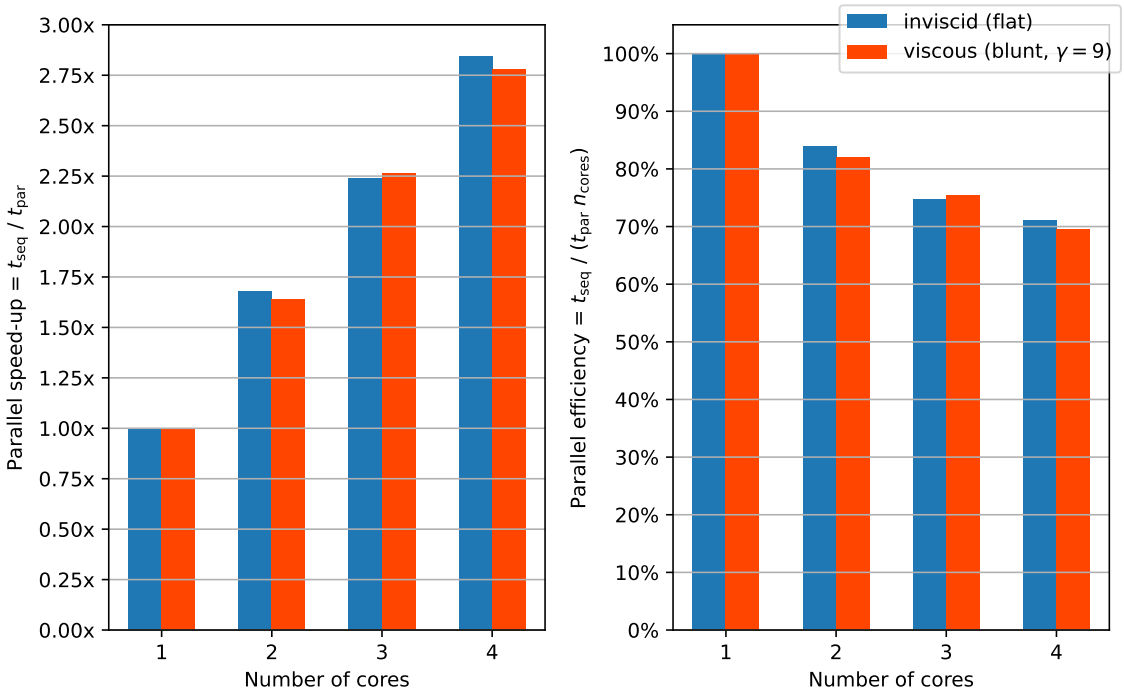


Figure 7.17: Parallel scalability, measured in terms of speed-up (i.e. the ratio between the wall time of the sequential run and the wall time of the parallel run, $t_{\text{seq}}/t_{\text{par}}$) and in terms of efficiency (i.e. the speed-up divided by the number of cores, $t_{\text{seq}}/(t_{\text{par}}\ n_{\text{cores}})$). Both *inviscid sine wave* and *viscous sine wave* test cases are reported.

8 | Conclusions and future developments

In this work, a DG-SEM method, coupled with an explicit AB time-advancing scheme, has been applied to the 1D haemodynamics equations. Indeed, DG-SEM methods are particularly attractive for wave propagation problems, due to their exponential p-convergence, thus providing the same accuracy with fewer degrees of freedom compared with continuous Galerkin formulations. A key contribution of the present work is the development of a mathematical and numerical formulation that relaxes the commonly adopted assumption of a flat velocity profile, which is physically inconsistent in a real setup (i.e. the velocity profile is either parabolic or blunt in blood vessels, based on the Womersley number). This is achieved by extending Roe's linearization to the 1D vascular flow equations, derived here for the case of a general velocity profile.

This formulation has been extended to complex network of vessels, accounting for regions with discontinuous material properties (which may represent medical devices such as stents) and bifurcation points by means of suitable matching conditions, stemming from the continuity of the flow rate and total pressure. This extension enables a comprehensive description of the circulatory system, in order to apply the present 1D model to real/clinical scenarios.

The 1D haemodynamics model has been successfully coupled with a 3D FSI model, demonstrating a possible multiscale framework for cardiovascular simulations. This approach is relevant in applications where a region of clinical interest (such as the heart or specific arterial segments) is resolved with a 3D model, while the rest of the circulatory system is simulated by means of computationally efficient 1D reduced-order solvers.

The software has been developed in C++ within the `life`^{x} framework, and its design allows high flexibility to employ this code in future studies, especially by introducing new boundary conditions through polymorphism.

Numerical experiments have been performed successfully in order to verify the implemented code, while proving a sufficient consistency with 3D FSI models. The h-, p- and

time convergence has been studied for the current numerical setup.

It would be interesting to introduce a numerical technique capable of describing the discontinuous material properties without additional matching conditions, e.g. by adapting the augmented formulation devised in [26, 27] to DG methods.

A more appropriate strategy for the 3D-1D coupling could be examined, in order to deal with the defective boundary conditions on the 3D FSI problem. Furthermore, we could perform additional sub-iterations within the same time step in order to require a stronger continuity of the flow rate on the coupling interface, thus improving considerably the results shown in this work.

A more detailed analysis could be carried out in order to study whether the super-convergent behavior still persists for higher-degree polynomials (i.e. with $p > 2$) and for different boundary conditions. Indeed, an analytical estimate should be derived to predict the order of h-convergence accurately for the problem at hand.

The time convergence presented in this work is suboptimal by one order. This is probably related to the choice of the AB scheme as time stepper; hence, it could be interesting the repeat the same numerical experiments with an explicit Runge-Kutta (RK) time-advancing scheme, since RK schemes exhibit a larger stability region and a less strict CFL condition on the maximum time step, thus displaying less numerical instability with respect to AB time steppers. Another reason for this reduced convergence rate could be the 1^{st}-order time discretization adopted to advance the boundary conditions in time; in this particular case, it would be sufficient to implement new boundary condition with an order of time discretization consistent with the order of the entire solver. Again, this can be readily achieved through polymorphism.

Finally, we remark that the nature of the algorithm prevents it from achieving perfect scalability, and only minor optimizations could be possibly applied. Further studies could be proposed to apply this code to a complex network of vessels (e.g. the 55-artery network employed in [2, 3, 6]), where a different inverse Riemann problem arises for each bifurcation point. In that case, the parallelization should become more advantageous, since multiple inverse Riemann problems must be solved iteratively within the same time step. However, since the current 1D reduced-order model is computationally inexpensive by nature, the parallel scalability represents only a secondary issue.

Bibliography

[1] A. Quarteroni and L. Formaggia. Mathematical Modelling and Numerical Simulation of the Cardiovascular System. In *Computational Models for the Human Body*, volume 12 of *Handbook of Numerical Analysis*, pages 3–127. Elsevier, 2004. doi: https://doi.org/10.1016/S1570-8659(03)12001-7.

[2] S.J. Sherwin, L. Formaggia, J. Peiró, and V. Franke. 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(6-7):673–700, 2003. doi: https://doi.org/10.1002/fld.543.

[3] S.J. Sherwin, V. Franke, J. Peiró, and K. Parker. One-dimensional modelling of a vascular network in space-time variables. *Journal of Engineering Mathematics*, 47 (3):217–250, 2003. doi: https://doi.org/10.1023/B:ENGI.0000007979.32871.e2.

[4] K.S. Matthys, J. Alastruey, J. Peiró, A.W. Khir, P. Segers, P.R. Verdonck, K.H. Parker, and S.J. Sherwin. Pulse wave propagation in a model human arterial network: Assessment of 1-D numerical simulations against in vitro measurements. *Journal of Biomechanics*, 40(15):3476–3486, 2007. doi: https://doi.org/10.1016/j.jbiomech.2007.05.027.

[5] E. Boileau, P. Nithiarasu, P.J. Blanco, L.O. Müller, F.E. Fossan, L.R. Hellevik, W.P. Donders, W. Huberts, M. Willemet, and J. Alastruey. A benchmark study of numerical schemes for one-dimensional arterial blood flow modelling. *International Journal for Numerical Methods in Biomedical Engineering*, 31(10):e02732, 2015. doi: https://doi.org/10.1002/cnm.2732.

[6] D. Kim and J. Tithof. One-dimensional modeling of blood flow: A comprehensive yet concise review. 2025. doi: https://doi.org/10.48550/arXiv.2508.18484.

[7] M.R. Pfaller, J. Pham, A. Verma, L. Pegolotti, N.M. Wilson, D.W. Parker, W. Yang, and A.L. Marsden. Automated generation of 0D and 1D reduced-order models of patient-specific blood flow. *International Journal for Numerical Methods in Biomedical Engineering*, 38(10):e3639, 2022. doi: https://doi.org/10.1002/cnm.3639.

[8] P.J. Blanco and L.O. Müller. One-Dimensional Blood Flow Modeling in the Cardiovascular System. From the Conventional Physiological Setting to Real-Life Hemodynamics. *International Journal for Numerical Methods in Biomedical Engineering*, 41 (3):e70020, 2025. doi: https://doi.org/10.1002/cnm.70020.

[9] L. Formaggia, D. Lamponi, and A. Quarteroni. One-dimensional models for blood flow in arteries. *Journal of Engineering Mathematics*, 47(3):251–276, 2003. doi: https://doi.org/10.1023/B:ENGI.0000007980.01347.29.

[10] J. Peiró and A. Veneziani. Reduced models of the cardiovascular system. In L. Formaggia, A. Quarteroni, and A. Veneziani, editors, *Cardiovascular Mathematics: Modeling and simulation of the circulatory system*, pages 347–394. Springer Milan, Milano, 2009. doi: https://doi.org/10.1007/978-88-470-1152-6_10.

[11] J. Alastruey, K.H. Parker, and S.J. Sherwin. Arterial pulse wave haemodynamics. In *11th International Conference on Pressure Surges*, pages 401–443. Virtual PiE Led t/a BHR Group, 2012.

[12] P.L. Roe. Approximate Riemann solvers, parameter vectors, and difference schemes. *Journal of Computational Physics*, 43(2):357–372, 1981. doi: https://doi.org/10.1016/0021-9991(81)90128-5.

[13] P.C. Africa. lifex: A flexible, high performance library for the numerical solution of complex finite element problems. *SoftwareX*, 20:101252, 2022. doi: https://doi.org/10.1016/j.softx.2022.101252.

[14] M. Bucelli. The lifex Library Version 2.0. *ACM Trans. Math. Softw.*, 51(4), 2025. doi: https://doi.org/10.1145/3748817.

[15] D. Arndt, W. Bangerth, M. Bergbauer, B. Blais, M. Fehling, R. Gassmöller, T. Heister, L. Heltai, M. Kronbichler, M. Maier, P. Munch, S. Scheuerman, B. Turcksin, S. Uzunbajakau, D. Wells, and M. Wichrowski. The deal.II library, version 9.7. *Journal of Numerical Mathematics*, 33(4):403–415, 2025. doi: https://doi.org/10.1515/jnma-2025-0115.

[16] W. Cao and Z. Zhang. Some Recent Developments in Superconvergence of Discontinuous Galerkin Methods for Time-Dependent Partial Differential Equations. *Journal of Scientific Computing*, 77(3):1402–1423, 2018. doi: https://doi.org/10.1007/s10915-018-0762-2.

[17] D. Doorly and S.J. Sherwin. Geometry and flow. In L. Formaggia, A. Quarteroni, and A. Veneziani, editors, *Cardiovascular Mathematics: Modeling and simulation of*

the circulatory system, pages 177–209. Springer Milan, 2009. doi: https://doi.org/10.1007/978-88-470-1152-6_5.

[18] J. Alastruey, T. Passerini, L. Formaggia, and J. Peiró. Physical determining factors of the arterial pulse waveform: theoretical analysis and calculation using the 1-D formulation. *Journal of Engineering Mathematics*, 77(1):19–37, 2012. doi: https://doi.org/10.1007/s10665-012-9555-z.

[19] A. Quarteroni, A. Veneziani, and C. Vergara. Geometric multiscale modeling of the cardiovascular system, between theory and practice. *Computer Methods in Applied Mechanics and Engineering*, 302:193–252, 2016. doi: https://doi.org/10.1016/j.cma.2016.01.007.

[20] R.J. LeVeque. *Finite Difference Methods for Ordinary and Partial Differential Equations*. Society for Industrial and Applied Mathematics, 2007. doi: https://doi.org/10.1137/1.9780898717839.

[21] A. Quarteroni. *Numerical Models for Differential Problems*. Springer Milan, 2014. doi: https://doi.org/10.1007/978-88-470-5522-3.

[22] G.E. Karniadakis and S.J. Sherwin. Advection and Advection–Diffusion. In *Spectral/hp Element Methods for Computational Fluid Dynamics*. Oxford University Press, 2005. doi: https://doi.org/10.1093/acprof:oso/9780198528692.003.0006.

[23] E.F. Toro. Notions on Hyperbolic Partial Differential Equations. In *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction*, pages 41–85. Springer Berlin Heidelberg, 1999. doi: https://doi.org/10.1007/978-3-662-03915-1_2.

[24] E.F. Toro. The Riemann Solver of Roe. In *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction*, pages 341–372. Springer Berlin Heidelberg, 1999. doi: https://doi.org/10.1007/978-3-662-03915-1_11.

[25] J.S. Hesthaven and T. Warburton. The key ideas. In *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications*, pages 19–41. Springer New York, 2008. doi: https://doi.org/10.1007/978-0-387-72067-8_2.

[26] E.F. Toro and A. Siviglia. Flow in Collapsible Tubes with Discontinuous Mechanical Properties: Mathematical Model and Exact Solutions. *Communications in Computational Physics*, 13(2):361–385, 2013. doi: https://doi.org/10.4208/cicp.210611.240212a.

[27] L.O. Müller, C. Parés, and E.F. Toro. Well-balanced high-order numerical schemes

for one-dimensional blood flow in vessels with varying mechanical properties. *Journal of Computational Physics*, 242:53–85, 2013. doi: https://doi.org/10.1016/j.jcp.2013.01.050.

[28] L. Formaggia, J.F. Gerbeau, F. Nobile, and A. Quarteroni. On the coupling of 3D and 1D Navier-Stokes equations for flow problems in compliant vessels. *Computer Methods in Applied Mechanics and Engineering*, 191(6):561–582, 2001. doi: https://doi.org/10.1016/S0045-7825(01)00302-4.

[29] L. Formaggia, A. Moura, and F. Nobile. On the stability of the coupling of 3D and 1D fluid-structure interaction models for blood flow simulations. *ESAIM: Modélisation mathématique et analyse numérique*, 41(4):743–769, 2007. doi: https://doi.org/10.1051/m2an:2007039.

[30] L. Formaggia, A. Quarteroni, and A. Veneziani. Multiscale models of the vascular system. In L. Formaggia, A. Quarteroni, and A. Veneziani, editors, *Cardiovascular Mathematics: Modeling and simulation of the circulatory system*, pages 395–446. Springer Milan, 2009. doi: https://doi.org/10.1007/978-88-470-1152-6_11.

[31] J. Donea, S. Giuliani, and J.P. Halleux. An arbitrary Lagrangian-Eulerian finite element method for transient dynamic fluid-structure interactions. *Computer Methods in Applied Mechanics and Engineering*, 33(1):689–723, 1982. doi: https://doi.org/10.1016/0045-7825(82)90128-1.

[32] T. Tezduyar and S. Sathe. Stabilization parameters in SUPG and PSPG formulations. *J Comput Appl Mech*, 4(1):71–88, 2003.

[33] L. Formaggia, A. Veneziani, and C. Vergara. Flow rate boundary problems for an incompressible fluid in deformable domains: Formulations and solution methods. *Computer Methods in Applied Mechanics and Engineering*, 199(9):677–688, 2010. doi: https://doi.org/10.1016/j.cma.2009.10.017.

[34] N. Xiao, J. Alastruey, and C.A. Figueroa. A systematic comparison between 1-D and 3-D hemodynamics in compliant arterial models. *International Journal for Numerical Methods in Biomedical Engineering*, 30(2):204–231, 2014. doi: https://doi.org/10.1002/cnm.2598.

[35] W. Bangerth, C. Burstedde, T. Heister, and M. Kronbichler. Algorithms and data structures for massively parallel generic adaptive finite element codes. *ACM Trans. Math. Softw.*, 38(2):14/1–28, 2012. doi: https://doi.org/10.1145/2049673.2049678.

A | Global form of the characteristic variables $W_{1,2}$

The characteristic variables $W_{1,2}$ must satisfy the following relations to exist in an analytical and global form:

$$\frac{\partial W_i}{\partial \mathbf{U}} = \mathbf{l}_i(\mathbf{U}) \qquad \forall i \in \{1,2\} \ , \tag{A.1}$$

$$\frac{\partial^2 W_i}{\partial Q \partial A} = \frac{\partial^2 W_i}{\partial A \partial Q} \qquad \forall i \in \{1,2\} \ , \tag{A.2}$$

as already reported in (2.21)–(2.23). The first relation (A.1) is needed to diagonalize the advection term from the quasi-linear form (2.10), while the relation (A.2) guarantees that $W_{1,2}$ can be obtained analytically by integrating (A.1). In other words, the left eigenvectors $\mathbf{l}_{1,2}$ of the Jacobian matrix $\mathrm{H}(\mathbf{U})$ are defined up to an arbitrary multiplicative factor $\zeta(A,Q)$ (2.19) and, by introducing (A.2), we are constraining $\zeta(A,Q)$ such that $W_{1,2}$ can be derived in an analytical and global form.

Firstly, we analyze the most general case, with a generic velocity profile and $\zeta(A,Q)$ depending on both the cross-sectional area A and the flow rate Q. We highlight that (A.1) can be rewritten into four scalar equations:

$$\frac{\partial W_i}{\partial A} = \left(\mathbf{l}_i(A,Q)\right)_1 \ , \qquad \frac{\partial W_i}{\partial Q} = \left(\mathbf{l}_i(A,Q)\right)_2 \qquad \forall i \in \{1,2\} \ , \tag{A.3}$$

with $\left(\mathbf{l}_i(A,Q)\right)_j$ being the j-th component of the i-th left eigenvector. By substituting (A.3) into (A.2), we obtain:

$$\frac{\partial \left(\mathbf{l}_i(A,Q)\right)_1}{\partial Q} = \frac{\partial \left(\mathbf{l}_i(A,Q)\right)_2}{\partial A} \qquad \forall i \in \{1,2\} \ . \tag{A.4}$$

Then, we introduce the analytical expression of the left eigenvectors (2.19), thus yielding:

$$\frac{\partial \zeta(A,Q)}{\partial Q}\left(\pm c_\alpha - \alpha\frac{Q}{A}\right) + \zeta(A,Q)\left(\pm\frac{\alpha(\alpha-1)}{c_\alpha}\frac{Q}{A^2} - \alpha\frac{1}{A}\right) = \frac{\partial \zeta(A,Q)}{\partial A}\,, \tag{A.5}$$

where c_α has been already defined in (2.18). However, we end up with another partial differential equation (PDE), with $\zeta(A,Q)$ as unknown, which does not admit an analytical solution.

As discussed in [1], in order to turn (A.5) into an ODE and obtain an analytical expression for $\zeta(A,Q)$, we can consider a flat velocity profile (i.e. $\alpha = 1$ (2.4) and $c_\alpha \equiv c$, with c being the wave speed (2.12)), and assume that the arbitrary function $\zeta(A,Q)$ depends on the cross-sectional area only (i.e. $\zeta(A,Q) = \zeta(A)$), thus simplifying (A.5) into:

$$-\zeta(A)\,\frac{1}{A} = \frac{\partial \zeta(A)}{\partial A}\,. \tag{A.6}$$

One possible choice of $\zeta(A)$ satisfying (A.6) is: $\zeta(A) = A^{-1}$. By substituting $\zeta(A) = A^{-1}$ into the left eigenvectors (2.19), the relations (A.3) become:

$$\begin{aligned} \frac{\partial W_{1,2}}{\partial A} &= \pm\frac{c}{A} - \frac{Q}{A^2}\,, \\ \frac{\partial W_{1,2}}{\partial Q} &= \frac{1}{A}\,. \end{aligned} \tag{A.7}$$

After that, it is possible to solve this system of ODEs analytically, thus yielding the following expression for the characteristic variables $W_{1,2}$:

$$W_{1,2} = \frac{Q}{A} \pm \int_0^A \frac{c}{A}\mathrm{d}A\,. \tag{A.8}$$

By introducing the tube law (2.3) (which is equivalent to assuming that the wave speed c is given by (2.12)), this expression can be further simplified into:

$$W_{1,2} = \frac{Q}{A} \pm 4c\,, \tag{A.9}$$

as already reported in (2.24). We highlight that this analytical expression for the characteristic variables (A.9) holds only for a flat velocity profile (i.e. with $\alpha = 1$) and with the tube law (2.3).

B | Derivation of Roe's averages

In this appendix, we present the complete derivation of Roe's averages $\widetilde{A}$, $\widetilde{Q}$, $\widetilde{c}$, whose expression is already reported in (3.36). We recall that the objective of Roe's method is to find a suitable linearization:

$$\frac{\partial \mathbf{U}}{\partial t} + \widetilde{\mathrm{H}} \frac{\partial \mathbf{U}}{\partial x} = \mathbf{0} \,, \tag{B.1}$$

where the linearized Jacobian matrix $\widetilde{\mathrm{H}}$ has been already defined in Section 3.4.2. We can start by enforcing the *conservation property* on $\widetilde{\mathrm{H}}$, i.e. $\widetilde{\mathrm{H}}$ should satisfy the following relation:

$$\widetilde{\mathrm{H}} \left(\mathbf{U}_R^e - \mathbf{U}_L^e\right) = \mathbf{F}(\mathbf{U}_R^e) - \mathbf{F}(\mathbf{U}_L^e) \,, \tag{B.2}$$

where the left state $\mathbf{U}_L^e$ and the right state $\mathbf{U}_L^e$ at the e-th boundary node are illustrated in Figure 3.2.

The approach proposed by Roe in [12] consists in choosing in a suitable variable change, denoted by $\mathbf{Z} = \mathbf{Z}(\mathbf{U})$, such that the conservation property is enforced along the following straight path:

$$\mathbf{Z}(\theta) = \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \mathbf{Z}_L^e + \theta \left(\mathbf{Z}_R^e - \mathbf{Z}_L^e\right) = \begin{pmatrix} z_{1,L}^e + \theta \left(z_{1,R}^e - z_{1,L}^e\right) \\ z_{2,L}^e + \theta \left(z_{2,R}^e - z_{2,L}^e\right) \end{pmatrix} \qquad \forall \theta \in [0,1] \,, \tag{B.3}$$

where $\mathbf{Z}_L^e := \mathbf{Z}(\mathbf{U}_L^e)$ and $\mathbf{Z}_R^e := \mathbf{Z}(\mathbf{U}_R^e)$. In this way, the jump in the solution at the interface between two cells is given by:

$$\begin{aligned} \mathbf{U}_R^e - \mathbf{U}_L^e &= \int_0^1 \frac{\mathrm{d}\mathbf{U}(\mathbf{Z}(\theta))}{\mathrm{d}\theta} \,\mathrm{d}\theta \\ &= \int_0^1 \frac{\mathrm{d}\mathbf{U}}{\mathrm{d}\mathbf{Z}} \frac{\mathrm{d}\mathbf{Z}}{\mathrm{d}\theta} \,\mathrm{d}\theta \\ &= \int_0^1 \frac{\mathrm{d}\mathbf{U}}{\mathrm{d}\mathbf{Z}} \,\mathrm{d}\theta \;\left(\mathbf{Z}_R^e - \mathbf{Z}_L^e\right) \\ &:= \widetilde{\mathrm{B}} \;\left(\mathbf{Z}_R^e - \mathbf{Z}_L^e\right) \end{aligned} \tag{B.4}$$

with $\widetilde{\mathrm{B}} := \int_0^1 \frac{\mathrm{d}\mathbf{U}}{\mathrm{d}\mathbf{Z}}\,\mathrm{d}\theta$. In particular, the variable change $\mathbf{Z} = \mathbf{Z}(\mathbf{U})$ must be chosen such that $\widetilde{\mathrm{B}}$ can be computed analytically. Similarly, the jump in the flux at the interface between two cells can be expressed as:

$$\begin{aligned}\mathbf{F}(\mathbf{U}_R^e) - \mathbf{F}(\mathbf{U}_L^e) &= \int_0^1 \frac{\mathrm{d}\mathbf{F}(\mathbf{U}(\mathbf{Z}(\theta)))}{\mathrm{d}\theta}\,\mathrm{d}\theta \\ &= \int_0^1 \frac{\mathrm{d}\mathbf{F}}{\mathrm{d}\mathbf{Z}}\frac{\mathrm{d}\mathbf{Z}}{\mathrm{d}\theta}\,\mathrm{d}\theta \\ &= \int_0^1 \frac{\mathrm{d}\mathbf{F}}{\mathrm{d}\mathbf{Z}}\,\mathrm{d}\theta\ (\mathbf{Z}_R^e - \mathbf{Z}_L^e) \\ &:= \widetilde{\mathrm{C}}\ (\mathbf{Z}_R^e - \mathbf{Z}_L^e)\ .\end{aligned} \tag{B.5}$$

Again, the variable change $\mathbf{Z} = \mathbf{Z}(\mathbf{U})$ must guarantee that the integration $\widetilde{\mathrm{C}} := \int_0^1 \frac{\mathrm{d}\mathbf{F}}{\mathrm{d}\mathbf{Z}}\,\mathrm{d}\theta$ can be performed analytically. Finally, substituting (B.4)–(B.5) into (B.2) yields:

$$\widetilde{\mathrm{H}}\,\widetilde{\mathrm{B}}\ (\mathbf{Z}_R^e - \mathbf{Z}_L^e) = \widetilde{\mathrm{C}}\ (\mathbf{Z}_R^e - \mathbf{Z}_L^e)\ , \tag{B.6}$$

which implies the following relation on the linearized Jacobian matrix:

$$\widetilde{\mathrm{H}} = \widetilde{\mathrm{C}}\,\widetilde{\mathrm{B}}^{-1}\ . \tag{B.7}$$

One possible, but not unique, choice for the variable change $\mathbf{Z} = \mathbf{Z}(\mathbf{U})$ which ensures the analytical computation of the matrices $\widetilde{\mathrm{B}}$ and $\widetilde{\mathrm{C}}$ is:

$$\mathbf{Z}(\mathbf{U}) = \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \begin{pmatrix} \sqrt{A} \\ \dfrac{Q}{\sqrt{A}} \end{pmatrix}\ , \tag{B.8}$$

which leads to the following parametrization of the solution $\mathbf{U}$ and the flux vector $\mathbf{F}(\mathbf{U})$:

$$\mathbf{U}(\mathbf{Z}) = \begin{pmatrix} A \\ Q \end{pmatrix} = \begin{pmatrix} z_1^2 \\ z_1 z_2 \end{pmatrix}\ , \qquad \mathbf{F}(\mathbf{U}(\mathbf{Z})) = \begin{pmatrix} Q \\ \alpha\dfrac{Q^2}{A} + \dfrac{\beta}{3\rho A_{\mathrm{ref}}}A^{\frac{3}{2}} \end{pmatrix} = \begin{pmatrix} z_1 z_2 \\ \alpha z_2^2 + \dfrac{\beta}{3\rho A_{\mathrm{ref}}}z_1^3 \end{pmatrix}\ . \tag{B.9}$$

It is noteworthy that the variable change (B.8) has been introduced such that $\mathbf{U}$ and $\mathbf{F}(\mathbf{U})$ are polynomial functions of $\mathbf{Z} = (z_1, z_2)^\top$. This choice simplifies considerably the

computation of $\widetilde{\mathrm{B}}$ and $\widetilde{\mathrm{C}}$, which reads:

$$\begin{aligned}\widetilde{\mathrm{B}} &= \int_0^1 \frac{\mathrm{d}\mathbf{U}}{\mathrm{d}\mathbf{Z}}\,\mathrm{d}\theta = \int_0^1 \begin{bmatrix} 2z_1 & 0 \\ z_2 & z_1 \end{bmatrix} \mathrm{d}\theta\,, \\ \widetilde{\mathrm{C}} &= \int_0^1 \frac{\mathrm{d}\mathbf{F}}{\mathrm{d}\mathbf{Z}}\,\mathrm{d}\theta = \int_0^1 \begin{bmatrix} z_2 & z_1 \\ \dfrac{\beta}{\rho A_{\mathrm{ref}}} z_1^2 & 2\alpha z_2 \end{bmatrix} \mathrm{d}\theta\,. \end{aligned} \tag{B.10}$$

After that, we can substitute the parametrization (B.3) and perform the integration in (B.10) analytically. For the sake of readability, we introduce the following quantities:

$$\begin{aligned}\overline{z}_i &:= \int_0^1 z_i(\theta)\,\mathrm{d}\theta = \frac{z_{i,L}^e + z_{i,R}^e}{2} \\ \widehat{z}_i^2 &:= \int_0^1 z_i^2(\theta)\,\mathrm{d}\theta = \frac{(z_{i,L}^e)^2 + z_{i,L}^e\, z_{i,R}^e + (z_{i,R}^e)^2}{3} \end{aligned} \qquad \forall i \in \{1,2\}\ . \tag{B.11}$$

Hence, we can further manipulate (B.10) and obtain the following expression for $\widetilde{\mathrm{B}}$ and $\widetilde{\mathrm{C}}$:

$$\begin{aligned}\widetilde{\mathrm{B}} &= \begin{bmatrix} 2\overline{z}_1 & 0 \\ \overline{z}_2 & \overline{z}_1 \end{bmatrix}\,, \\ \widetilde{\mathrm{C}} &= \begin{bmatrix} \overline{z}_2 & \overline{z}_1 \\ \dfrac{\beta}{\rho A_{\mathrm{ref}}} \widehat{z}_1^2 & 2\alpha \overline{z}_2 \end{bmatrix}\,. \end{aligned} \tag{B.12}$$

Finally, we can find the linearized Jacobian matrix $\widetilde{\mathrm{H}}$ by applying the relation (B.7):

$$\begin{aligned}\widetilde{\mathrm{H}} = \widetilde{\mathrm{C}}\widetilde{\mathrm{B}}^{-1} &= \begin{bmatrix} 0 & 1 \\ \dfrac{\beta}{2\rho A_{\mathrm{ref}}} \dfrac{\widehat{z}_1^2}{\overline{z}_1} - \alpha \left(\dfrac{\overline{z}_2}{\overline{z}_1}\right)^2 & 2\alpha \dfrac{\overline{z}_2}{\overline{z}_1} \end{bmatrix} \\ &:= \begin{bmatrix} 0 & 1 \\ \widetilde{c}^2 - \alpha \left(\dfrac{\widetilde{Q}}{\widetilde{A}}\right)^2 & 2\alpha \dfrac{\widetilde{Q}}{\widetilde{A}} \end{bmatrix}\,, \end{aligned} \tag{B.13}$$

where Roe's averages $\widetilde{A}$, $\widetilde{Q}$, $\widetilde{c}$ are equal to:

$$\begin{aligned}
\widetilde{A} &= \left(\frac{\sqrt{A_L^e} + \sqrt{A_R^e}}{2} \right)^2 , \\
\widetilde{Q} &= \frac{\sqrt{\widetilde{A}}}{2} \left(\frac{Q_L^e}{\sqrt{A_L^e}} + \frac{Q_R^e}{\sqrt{A_R^e}} \right) , \\
\widetilde{c} &= \sqrt{\frac{\beta}{2\rho A_{\text{ref}}} \frac{A_L^e + \sqrt{A_L^e A_R^e} + A_R^e}{3\sqrt{\widetilde{A}}}} \neq c(\widetilde{A}) ,
\end{aligned} \tag{B.14}$$

as already reported in (3.35)–(3.36). It is worth noting that the expression for $\widetilde{\mathrm{H}}$ (B.13) is equivalent to the one of the original Jacobian matrix $\mathrm{H}(\mathbf{U})$ (2.11), evaluated at Roe's averages, i.e. $\widetilde{\mathrm{H}} = \mathrm{H}(\widetilde{A}, \widetilde{Q}, \widetilde{c})$.

C | Solution of the Riemann problem with Roe's linearization

In this appendix, we show the solution of the Riemann problem in the case of Roe's linearization, as depicted in Figure C.1. Both the intermediate state $\mathbf{U}^{*e}$ and the numerical flux $\mathbf{F}^{*}$ are derived, as proposed in [24].

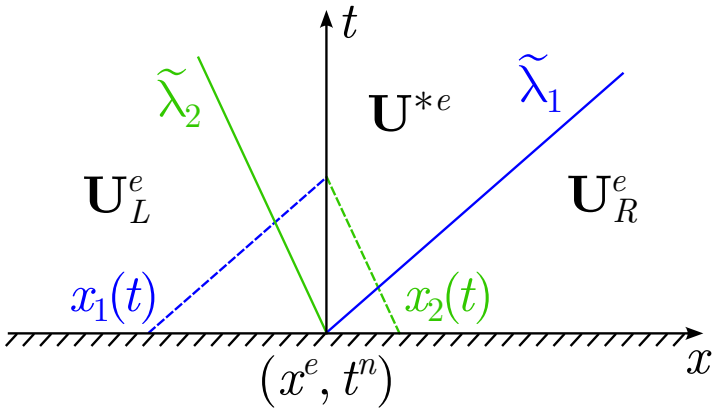


Figure C.1: Graphical representation of the Riemann problem applied to (C.1). The domain is identified by the dashed oblique line. $\mathbf{U}_L^e$ and $\mathbf{U}_R^e$ represent the left and right states. $\mathbf{U}^{*e}$ is the solution within the intermediate (or *star*) region. $\widetilde{\lambda}_1$ and $\widetilde{\lambda}_2$ are the eigenvalues of the linearized Jacobian matrix $\widetilde{\mathrm{H}}$ associated with the right- and left-running waves, respectively. $x_1(t)$ and $x_2(t)$ are the characteristic lines along which $\widetilde{W}_1$ and $\widetilde{W}_2$ propagate unchanged.

We recall the expression for Roe's linearization in its quasi-linear form (3.34):

$$\frac{\partial \mathbf{U}}{\partial t} + \widetilde{\mathrm{H}} \frac{\partial \mathbf{U}}{\partial x} = \mathbf{0} \,, \tag{C.1}$$

where the linearized Jacobian matrix $\widetilde{\mathrm{H}}$ is a constant-coefficient matrix, equivalent to the original Jacobian matrix evaluated at Roe's averages (3.36), as provided by (3.35). The linearized system (C.1) can be solved by employing the method of characteristics

[23]. In particular, the left and right eigenvectors of $\widetilde{\mathrm{H}}$, which we denote by $\widetilde{\mathbf{l}}_{1,2}$ and $\widetilde{\mathbf{r}}_{1,2}$, are equivalent to the ones introduced in (2.19) evaluated at Roe's averages (3.36); we recall that these left and right eigenvectors are chosen to be mutually orthonormal, formally $\widetilde{\mathbf{l}}_i \cdot \widetilde{\mathbf{r}}_i = \delta_{ij}$ (with δ_{ij} being the Kronecker delta). By applying the definition of characteristic variables (2.21) to the linearized case (C.1), we can find the following approximation for the characteristic variables $W_{1,2}$:

$$W_{1,2} \approx \widetilde{W}_{1,2} = \widetilde{\mathbf{l}}_{1,2} \cdot \mathbf{U} \ . \tag{C.2}$$

Let $\widetilde{\mathrm{L}}$ and $\widetilde{\mathrm{R}}$ be the matrices collecting the left and right eigenvectors $\widetilde{\mathbf{l}}_{1,2}$ and $\widetilde{\mathbf{r}}_{1,2}$, which are equivalent to the matrices introduced in (2.20) evaluated at Roe's averages (3.36). We remark that $\widetilde{\mathrm{L}}$ and $\widetilde{\mathrm{R}}$ are chosen such that $\widetilde{\mathrm{L}}\,\widetilde{\mathrm{R}} = \mathbb{I}$, with $\mathbb{I}$ being the identity matrix. By rewriting (C.2) in a vector form:

$$\widetilde{\mathbf{W}} = \begin{pmatrix} \widetilde{W}_1 \\ \widetilde{W}_2 \end{pmatrix} = \widetilde{\mathrm{L}}\,\mathbf{U} \ , \tag{C.3}$$

which is equivalent to:

$$\mathbf{U} = \widetilde{\mathrm{R}}\,\widetilde{\mathbf{W}} \ . \tag{C.4}$$

Then we proceed as already shown in Section 3.4.1 and allow the characteristic variables $\widetilde{W}_{1,2}$ to propagate unchanged along the characteristic lines $x_{1,2}(t)$ (as illustrated in Figure C.1). In other words, this is equivalent to substituting the approximate characteristic variables $\widetilde{W}_{1,2}$ into the system (3.31):

$$\begin{cases} \widetilde{W}_1(\mathbf{U}^{*e}) = \widetilde{W}_1(\mathbf{U}_L^e) \\ \widetilde{W}_2(\mathbf{U}^{*e}) = \widetilde{W}_2(\mathbf{U}_R^e) \end{cases} . \tag{C.5}$$

By substituting the definition of $\widetilde{W}_{1,2}$ (C.2) into (C.5):

$$\begin{cases} \widetilde{\mathbf{l}}_1 \cdot \mathbf{U}^{*e} = \widetilde{\mathbf{l}}_1 \cdot \mathbf{U}_L^e \\ \widetilde{\mathbf{l}}_2 \cdot \mathbf{U}^{*e} = \widetilde{\mathbf{l}}_2 \cdot \mathbf{U}_R^e \end{cases} , \tag{C.6}$$

which is equivalent to the vector form:

$$\widetilde{\mathrm{L}}\,\mathbf{U}^{*e} = \begin{pmatrix} \widetilde{\mathbf{l}}_1 \cdot \mathbf{U}_L^e \\ \widetilde{\mathbf{l}}_2 \cdot \mathbf{U}_R^e \end{pmatrix} \ . \tag{C.7}$$

By inverting $\widetilde{\mathrm{L}}$, (C.7) becomes:

$$\mathbf{U}^{*e} = \widetilde{\mathrm{R}} \begin{pmatrix} \widetilde{\mathrm{l}}_1 \cdot \mathbf{U}_L^e \\ \widetilde{\mathrm{l}}_2 \cdot \mathbf{U}_R^e \end{pmatrix} . \tag{C.8}$$

By applying further algebraic manipulations to (C.8), it is possible to derive the following expression for the solution within the intermediate region $\mathbf{U}^{*e}$:

$$\mathbf{U}^{*e} = \mathbf{U}_L^e + \left[\widetilde{\mathrm{l}}_2 \cdot (\mathbf{U}_R^e - \mathbf{U}_L^e) \right] \widetilde{\mathbf{r}}_2 , \tag{C.9}$$

as already reported in (3.38).

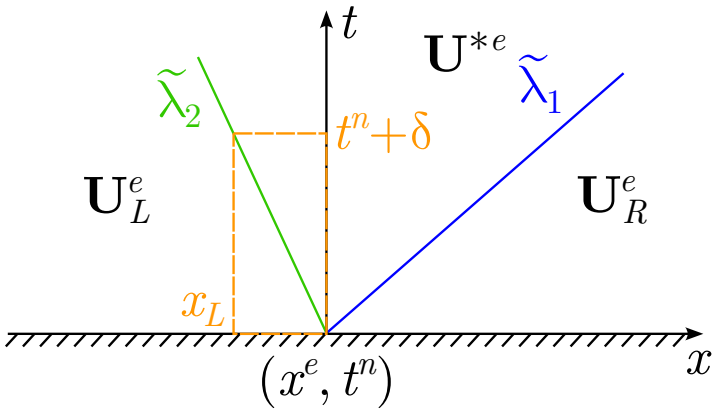


Figure C.2: Graphical representation of the Riemann problem applied to (C.1). The domain is identified by the dashed oblique line. $\mathbf{U}_L^e$ and $\mathbf{U}_R^e$ represent the left and right states. $\mathbf{U}^{*e}$ is the solution within the intermediate (or *star*) region. $\widetilde{\lambda}_1$ and $\widetilde{\lambda}_2$ are the eigenvalues of the linearized Jacobian matrix $\widetilde{\mathrm{H}}$ associated with the right- and left-running waves, respectively. The orange box denotes the integration domain to compute the approximate numerical flux $\mathbf{F}^*$.

After that, we proceed to compute the numerical flux $\mathbf{F}^*$ for Roe's approximate Riemann solver. We recall the conservative form for the Riemann problem:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}(\mathbf{U})}{\partial x} = \mathbf{0} , \tag{C.10}$$

which is equivalent to (2.13), except for the reaction term $\mathbf{S}(\mathbf{U})$, which is null at the interface between two consecutive cells, as already discussed in Section 3.4. Then we integrate (C.10) along the domain $[x_L, x^e] \times [t^n, t^n + \delta]$ (as illustrated in Figure C.2), resulting in:

$$\int_{x_L}^{x^e} \int_{t^n}^{t^n+\delta} \frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}(\mathbf{U})}{\partial x} \, \mathrm{d}t \, \mathrm{d}x = \mathbf{0} , \tag{C.11}$$

where $x_L := x^e + \widetilde{\lambda}_2\,\delta$, with $\widetilde{\lambda}_2$ being the second eigenvalue of the linearized Jacobian matrix $\widetilde{\mathrm{H}}$, equivalent to (2.17) evaluated at Roe's averages (3.36). For the sake of simplicity, we will consider $x_e = 0\,\mathrm{m}$ and $t^n = 0\,\mathrm{s}$ hereafter, but the following procedure can be readily extended for the case $x_e \neq 0\,\mathrm{m}$ and $t^n \neq 0\,\mathrm{s}$. Hence, (C.11) becomes:

$$\int_{x_L}^{0}\int_{0}^{\delta} \frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}(\mathbf{U})}{\partial x}\,\mathrm{d}t\,\mathrm{d}x = \mathbf{0} \tag{C.12}$$

with $x_L = \widetilde{\lambda}_2\,\delta$. By carrying out the integrals appearing in (C.12), we obtain:

$$\int_{x_L}^{0} \big[\mathbf{U}(x,\delta) - \mathbf{U}(x,0)\big]\mathrm{d}x + \int_{0}^{\delta} \big[\mathbf{F}(0,t) - \mathbf{F}(x_L,t)\big]\mathrm{d}t = \mathbf{0}\,. \tag{C.13}$$

By comparing the terms appearing in (C.13) with Figure C.2, the following equalities can be noticed:

$$\begin{aligned}
\mathbf{U}(x,\delta) \equiv \mathbf{U}^{*e} = \mathrm{const} \qquad & \forall x \in (x_L, 0)\,,\\
\mathbf{U}(x,0) \equiv \mathbf{U}^{e}_{L} = \mathrm{const} \qquad & \forall x \in (x_L, 0)\,,\\
\mathbf{F}(0,t) \equiv \mathbf{F}^{*} = \mathrm{const} \qquad & \forall t \in (0, \delta)\,,\\
\mathbf{F}(x_L,t) \equiv \mathbf{F}(\mathbf{U}^{e}_{L}) = \mathrm{const} \qquad & \forall t \in (0, \delta)\,.
\end{aligned} \tag{C.14}$$

By substituting (C.14) into (C.13), we obtain:

$$x_L\left[\mathbf{U}^{e}_{L} - \mathbf{U}^{*e}\right] + \delta\left[\mathbf{F}^{*} - \mathbf{F}(\mathbf{U}^{e}_{L})\right] = \mathbf{0}\,. \tag{C.15}$$

By isolating the numerical flux $\mathbf{F}^{*}$ and recalling that $x_L = \widetilde{\lambda}_2\,\delta$:

$$\mathbf{F}^{*}(\mathbf{U}^{e}_{L}, \mathbf{U}^{e}_{R}) = \mathbf{F}(\mathbf{U}^{e}_{L}) + \widetilde{\lambda}_2\left[\mathbf{U}^{*e} - \mathbf{U}^{e}_{L}\right]\,. \tag{C.16}$$

Finally, we introduce the approximation for the intermediate state (C.9) into (C.16), thus yielding the following expression for the numerical flux in the case of Roe's linearization:

$$\mathbf{F}^{*}(\mathbf{U}^{e}_{L}, \mathbf{U}^{e}_{R}) = \mathbf{F}(\mathbf{U}^{e}_{L}) + \widetilde{\lambda}_2\left[\widetilde{\mathbf{l}}_2 \cdot (\mathbf{U}^{e}_{R} - \mathbf{U}^{e}_{L})\right]\widetilde{\mathbf{r}}_2\,, \tag{C.17}$$

as already reported in (3.39).

D | Non-reflecting boundary condition for the 3D FSI problem

Given the hyperbolic nature of 3D FSI problems, boundary conditions should be prescribed carefully, in order to avoid spurious wave reflections. For this purpose, we derive a non-reflecting boundary condition to be applied to the fluid subproblem from the FSI formulation (5.1), based on the linearization of the characteristic variables (2.24), as proposed in [3].

Under the assumption of an inviscid flat velocity profile (i.e. $\alpha = 1$ and $K_R = 0$ (2.8)), a non-reflecting boundary condition can be obtained by requiring that the incoming characteristic variable W_{in} is preserved at the outlet, as already expressed in (3.48). For simplicity, let W_2 be the incoming characteristic variable at the outlet, which entails:

$$W_2(\mathbf{x}, t) = \text{const} \qquad \forall \mathbf{x} \in \Gamma_{f,\text{out}},\ \forall t \in (0, t_f]\ . \tag{D.1}$$

We recall that, for a flat velocity profile, we have at our disposal the analytical expression for the characteristic variable W_2 (2.24). We proceed to linearize W_2 around the reference state appearing in the tube law (2.3), i.e. we substitute $(A, Q)^\top \approx (A_{\text{ref}} + a', q')^\top$ into (2.24), yielding the following linearized version for the characteristic variable W_2:

$$W_2 \approx \frac{q'}{A_{\text{ref}}} - \frac{c_{\text{ref}}}{A_{\text{ref}}} a' \tag{D.2}$$

with a' and q' being sufficiently small perturbations with respect to the reference values, while the relation $c_{\text{ref}} = c(A_{\text{ref}})$ is expressed by (2.12). Furthermore, the linearization of the tube law (2.3) around the reference state $(A_{\text{ref}}, P_{\text{ref}})$ yields:

$$p' = \frac{\rho_f\, c_{\text{ref}}^2}{A_{\text{ref}}}\, a'\ . \tag{D.3}$$

By substituting (D.3) into (D.2), we find the following linearized expression for the incoming characteristic variable:

$$W_2 \approx \frac{q'}{A_{\text{ref}}} - \frac{p'}{\rho_f \, c_{\text{ref}}} \, . \tag{D.4}$$

Since the characteristic variables are defined up to an arbitrary additive constant, (D.1) is equivalent to requiring:

$$W_2(\mathbf{x}, t) = 0 \qquad \forall \mathbf{x} \in \Gamma_{f,\text{out}}, \; \forall t \in (0, t_f] \, . \tag{D.5}$$

By substituting (D.4) into (D.5):

$$p' = \frac{\rho_f \, c_{\text{ref}}}{A_{\text{ref}}} \, q' \qquad \forall \mathbf{x} \in \Gamma_{f,\text{out}}, \; \forall t \in (0, t_f] \, . \tag{D.6}$$

Finally, (D.6) can be translated onto the 3D FSI model by considering the initial conditions as the reference state around which the linearization is performed (i.e. $A_{\text{ref}} \equiv A_0$), leading to:

$$P_{3\text{D}} = R_t \, Q_{3\text{D}} \qquad \forall \mathbf{x} \in \Gamma_{f,\text{out}} \, , \tag{D.7}$$

where R_t represents the terminal resistance and is equal to:

$$R_t = \frac{\rho_f \, c_0}{A_0} \, , \tag{D.8}$$

as already reported in (5.8)–(5.9).

E | Simulation setup for verification against benchmark solutions

Parameter	Units	*inviscid Gaussian pulse*	*viscous Gaussian pulse*	*common carotid artery*	*upper thoracic aorta*
Vessel length L	m	10	10	0.126	0.24137
Number of cells M	–	500	500	1	2
Polynomial degree p	–	4	4	4	4
Final time t_f	s	1.5	1.5	10	20.055
Time step Δt	s	10^{-4}	10^{-4}	10^{-3}	5×10^{-4}
Order of the AB scheme s	–	2	2	2	2
Initial area A_0	m^2	$\pi \times 10^{-4}$	$\pi \times 10^{-4}$	2.2038×10^{-5}	4.5239×10^{-4}
Initial flow rate Q_0	m^3/s	0	0	0	0
Blood density ρ	kg/m^3	1050	1050	1060	1060
Blood viscosity μ	Pa s	0	4×10^{-3}	4×10^{-3}	4×10^{-3}
Velocity profile	–	Flat ($\alpha = 1$)	Blunt $\left(\gamma = 9, \alpha = \frac{11}{10}\right)$	Parabolic $\left(\gamma = 2, \alpha = \frac{4}{3}\right)$	Blunt $\left(\gamma = 9, \alpha = \frac{11}{10}\right)$
Young modulus E	Pa	4×10^5	4×10^5	7×10^5	4×10^5
Poisson ratio ξ	–	0.5	0.5	0.5	0.5
Wall thickness h	m	1.5×10^{-3}	1.5×10^{-3}	3×10^{-4}	1.2×10^{-3}
Reference area A_{ref}	m^2	$\pi \times 10^{-4}$	$\pi \times 10^{-4}$	2.8274×10^{-5}	4.5239×10^{-4}
Reference flow rate Q_{ref}	m^3/s	10^{-6}	10^{-6}	10^{-5}	10^{-4}
Reference pressure P_{ref}	Pa	0	0	1.0933×10^4	9.46×10^3
Windkessel proximal resistance R_1	$Pa\,s/m^3$	–	–	2.4875×10^8	1.1752×10^7
Windkessel compliance C	m^3/Pa	–	–	1.7529×10^{-10}	1.0163×10^{-8}
Windkessel distal resistance R_2	$Pa\,s/m^3$	–	–	1.8697×10^9	1.1167×10^8
Windkessel outflow pressure P_{out}	Pa	–	–	0	0

Table E.1: Simulation setup for verification against benchmark solutions, taken from [5]. The parameters used for *inviscid Gaussian pulse*, *viscous Gaussian pulse*, *common carotid artery*, *upper thoracic aorta* test cases are reported.

Parameter	Units	*aortic bifurcation* (*abdominal aorta*)	*aortic bifurcation* (*common iliac arteries*)
Vessel length L	m	0.086	0.085
Number of cells M	–	1	1
Polynomial degree p	–	2	2
Final time t_f	s	29.7	29.7
Time step Δt	s	10^{-3}	10^{-3}
Order of the AB scheme s	–	2	2
Initial area A_0	m^2	1.8062×10^{-4}	$0.947\,87 \times 10^{-4}$
Initial flow rate Q_0	m^3/s	0	0
Blood density ρ	kg/m^3	1060	1060
Blood viscosity μ	Pa s	4×10^{-3}	4×10^{-3}
Velocity profile	–	Blunt $\left(\gamma = 9, \alpha = \frac{11}{10}\right)$	Blunt $\left(\gamma = 9, \alpha = \frac{11}{10}\right)$
Young modulus E	Pa	5×10^5	7×10^5
Poisson ratio ξ	–	0.5	0.5
Wall thickness h	m	1.032×10^{-3}	0.7200×10^{-3}
Reference area A_{ref}	m^2	2.3235×10^{-4}	1.1310×10^{-4}
Reference flow rate Q_{ref}	m^3/s	8×10^{-5}	2.5×10^{-5}
Reference pressure P_{ref}	Pa	9.46×10^3	9.46×10^3
Windkessel proximal resistance R_1	$Pa\,s/m^3$	–	6.8123×10^7
Windkessel compliance C	m^3/Pa	–	3.6664×10^{-10}
Windkessel distal resistance R_2	$Pa\,s/m^3$	–	3.1013×10^9
Windkessel outflow pressure P_{out}	Pa	–	0

Table E.2: Simulation setup for verification against benchmark solutions, taken from [5]. The parameters used for *aortic bifurcation* test case are reported.

F | Simulation setup for 3D-1D coupled simulations

Parameter	Units	*pressure Gaussian pulse*
Vessel length L	m	0.30
Number of cells M	–	15
Polynomial degree p	–	4
Final time t_f	s	0.1
Time step Δt	s	10^{-4}
Order of the AB scheme s	–	2
Order of the BDF scheme	–	2
Initial area $A_{1D,0}$	m^2	$\pi \times 10^{-4}$
Initial flow rate $Q_{1D,0}$	m^3/s	0
Initial velocity $\mathbf{u}_{3D,0}$	m/s	**0**
Initial pressure $P_{3D,0}$	m/s	0
Blood density ρ_f	kg/m^3	1060
Blood viscosity μ	Pa s	4×10^{-3}
Velocity profile	–	Blunt $\left(\gamma = 9, \alpha = \frac{11}{10}\right)$
Structure density ρ_s	kg/m^3	1000
Young modulus E	Pa	7×10^5
Poisson ratio ξ	–	0.3
Wall thickness h	m	10^{-3}
Reference area A_{ref}	m^2	$\pi \times 10^{-4}$
Reference flow rate Q_{ref}	m^3/s	10^{-4}
Reference pressure P_{ref}	Pa	0
Terminal resistance R_t	$Pa\,s/m^3$	2.0494×10^7

Table F.1: Simulation setup for *pressure Gaussian pulse* test case, used for 3D and 1D standalone simulations. The 1D mesh is reduced to 20 cm for the coupled simulation, while keeping the same mesh size (i.e. $h = 0.02$ m).

List of Figures

List of Tables

List of Algorithms