BSc. Thesis Applied Mathematics and Applied Physics

One-dimensional blood flow modelling in the human arterial system with Finite Volume Methods

MARCO ROZENDAAL

Delft University of Technology

Supervisors

Prof. dr. ir. C. Vuik

Dr. S. Kenjereš

Other Committee Members

Dr. J.L.A. Dubbeldam

Drs. N. Bhattacharya

Augustus, 2017

Delft

Abstract

In this thesis, it is investigated how the blood flow changes in different physiological situations, using a one-dimensional model, that describes the blood flow in compliant vessels. The one-dimensional model, that describes the blood flow, is derived based on the physical laws of conservation of momentum and conservation of mass. A high resolution flux differencing scheme is applied to a stented artery, a tapered artery, an arterial bifurcation and a network of the 55 main arteries in the human arterial tree. It is found that inserting a stent led to an increase in the peak pressure and a dip below the equilibrium pressure, just before the stent. The nonlinearities in the model are shown using a long tapered artery, in the form of the steepening of the pulse. The current treatment of the bifurcations has led to reasonable physical reflections, in the bifurcation test cases. The network of 55 arteries showed that bifurcations play an important role in the blood flow patterns and, as such, they should not be ignored when considering a single blood vessel. Further research should focus on the implementation of a vascular prosthesis and more advanced treatments of the bifurcations.

Contents

1 Introduction

The computational power doubles every two years, as Moore's law predicts. It, therefore, only seems reasonable that complex simulations should be applied to everyday life more and more. One specific field, in which the use of complex models can be critical, is the field of biomedical engineering. One of many applications in the field of biomedical engineering is the blood flow in the human arterial tree. The problem that arises with patient specific models is the lack of suitable patient data and, still, the computational time when using the complete Navier-Stokes equations in three-dimensions (3D). For the former problem, new methods and instruments can be designed to measure the specific problem parameters. And, for the latter problem, a simplified model that describes the flow of blood in arteries, can be derived under reasonable assumptions. Formaggia et al. [10] have shown that the simplified models provide useful information for the practitioners at reasonable computational cost, thus patient specific models used for the planning of medical interventions can become a reality soon.

The aim of this thesis is the investigation of blood flow using a one-dimensional model, in different physiological situations.

As a first physiological situation, we will be investigating a treatment of atherosclerosis, the accumulation of white blood cells or plaque, using a stent, an expandable metal mesh. Atherosclerosis restricts the flow of blood. Plaques can, in extreme cases, cause irreversible damage to downstream tissue when the blood flow restriction is so severe that ischemia, the insufficient blood supply to tissues, occurs or it can detach, move into the circulation, and eventually obstruct downstream branches.

Normal arterial walls are fairly elastic, but when a stent is inserted into an artery, it becomes far stiffer. The pressure pulse from the heart will, therefore, bump into the stent which does not expand as easily and so a part of the pulse is reflected upstream. If this reflected pressure pulse is superimposed on a pressure pulse from the heart at a suture line, a tread of a stitch, of an arterial prosthesis, then the stitch can be weakened over time and can, eventually, lead to death, if left unnoticed.

The tapering of an artery is considered as a second physiological situation since most arteries are slightly tapered. The arterial tree uses branching and tapering to decrease the cross-sectional area of each blood vessel downstream in order to allow the numerous capillaries to have thin vessel walls. The hair thin vessel walls of the capillaries aid the exchange of water, oxygen, carbon dioxide and many other nutrients between the blood and the surrounding tissues.

As a final physiological situation, a model of the whole arterial tree is made. Arteries are not single isolated blood vessels as considered above, but, instead, they form a vastly complex network. Considering part of the arterial tree allows us to see the effect of the bifurcations on the blood flow.

This thesis is structured in the following way. First, a short summary of the current work in the field of one-dimensional models of blood flow is given. After that, the equations that describe the blood flow in compliant vessels are derived from the physical principles of conservation of momentum and conservation of mass. The derived equations are transformed into the more insightful characteristic equations. To make the concepts of characteristic variables and systems of PDEs more familiar, a linear test system is created and analysed. In the section after that, a stented artery and a tapered artery are subjected to the derived equations. A bifurcation and a model of the human arterial tree are considered as final test cases. Lastly, conclusions about the performed test cases are drawn.

2 Literature survey

We will first give a short summary of the current work of one-dimensional models of the human vascular system.

A system of Partial Differential Equations (PDEs) can be derived, based on the physical laws of conservation of momentum and conservation of mass, in order to model the blood flow in the human vascular system. There are two general forms of these systems of PDEs, one based on the volumetric flow Q and mean velocity u and the other one based on the cross-sectional area A and mean velocity u. The two forms are equivalent for smooth solutions, but if the solution is not continuous then the system with the variables Q and u is the proper system to use, see for example Sherwin et al. [1]. It was also shown that, under physiological conditions, the solution remains smooth.

There have been several relationships between the internal pressure of the blood vessel and the cross-sectional area of the blood vessel in literature, see Mynard [5] for a comprehensive list. The one most commonly used in literature is adapted in this thesis, see [1, 2, 5, 9].

Finite Element Methods (FEM), such as the Taylor Galerkin scheme or the Discontinuous Galerkin scheme, are often used in literature. The problem with FEMs is that they are more difficult to implement than other methods.

A Finite Volume Method (FVM), that is designed for spatially varying flux functions, is described by Bale and LeVeque in [4]. A good introduction to FVMs is given in *Finite Volume Methods For Hyperbolic Problems*, a book by LeVeque [3].

It is investigated in [1, 2, 5, 9] how stents influence the wave patterns. Sherwin et al. [1] also investigated what a tapered artery will do to the blood flow. Formaggia [2] tried modelling an endograft, a stent with closed walls, to treat an abdominal aortic aneurysm, a localized enlargement of the abdominal aortic. Several networks and physiological data are used in [1, 2, 5, 9] to model the main arteries in the human vascular system.

3 Derivation of the model

We derive a one-dimensional model for blood flow in the human arterial system. The arteries are modelled as compliant vessels. We assume that the blood vessels have a fixed cylindrical axis, the x-axis, and we assume that the cross-sections are strictly circular. Furthermore, we assume a flat velocity profile within each circular cross-section, thus the blood velocity within each circular cross-section is constant. The mean velocity of the blood across a cross-section is given by $u(x,t)$. The cross-sectional area $A(x,t)$ and mean velocity of the blood $u(x,t)$ are shown in Figure 1 from Peiró et al. [9].

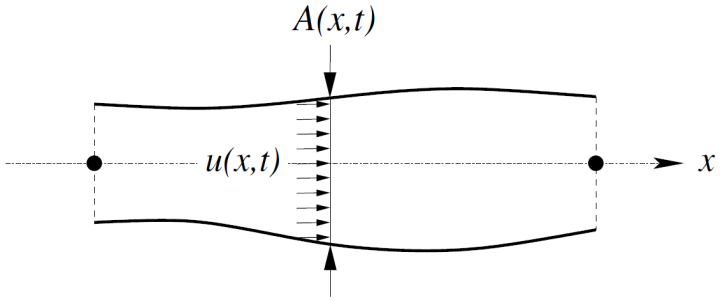

Figure 1: Schematic overview of conserved variables from Peiró et al. [9]

We will be using the Centimetre Gram Seconds (CGS) system instead of the standard SI units. A conversion Table from the CGS system to SI units can be found in appendix A.

3.1 Conservation of mass

To derive the conservation of mass equation, we consider a control volume from $x = a$ to $x = b$. The total mass inside this control volume is given by $\int_a^b \rho A \, dx$, where $A(x,t)$ is the cross-sectional area at position x at time t in cm^2 and ρ is the density in g/cm^3. It is assumed that blood can be modelled as an incompressible fluid and hence ρ is independent of space and time. The rate at which the mass inside of the control volume changes depends on the amount of mass that flows through the boundaries per unit of time i.e. the mass flux. The mass flux through the boundary at $x = a$ is $\rho A(a,t)u(a,t)$ and similarly $\rho A(b,t)u(b,t)$ at $x = b$, where $u(x,t)$ is the mean velocity of the blood at position x at time t in cm/s. Therefore

$$\frac{d}{dt}\int_a^b \rho A \, dx = \rho A(a,t)u(a,t) - \rho A(b,t)u(b,t) \tag{1}$$

We assume that A and u are sufficiently smooth. We recognise that ride hand side can be rewritten as the integral

$$\rho A(a,t)u(a,t) - \rho A(b,t)u(b,t) = -\rho \int_a^b \frac{\partial Au}{\partial x} \, dx \tag{2}$$

Bringing the derivative with respect to t inside of the integral as well as the expression above and noting that ρ is constant, gives

$$\rho \int_a^b \left(\frac{\partial A}{\partial t} + \frac{\partial Au}{\partial x} \right) dx = 0 \tag{3}$$

Since this equation is true for any a and b, it follows that the integrand has to be zero. Hence

$$\frac{\partial A}{\partial t} + \frac{\partial Au}{\partial x} = 0 \tag{4}$$

3.2 Conservation of momentum

Next, we derive the conservation of momentum equation. The argument goes analogously to that of the conservation of mass. Consider a control volume from $x = a$ to $x = b$. Because momentum is conserved,

the total momentum inside of the control volume, $\int_a^b \rho A u \, dx$, can only change due to the momentum flux through the boundaries $\rho A(a,t)u(a,t)^2 - \rho A(b,t)u(b,t)^2$ and due to the applied forces F. Therefore we have

$$\frac{d}{dt}\int_a^b \rho A u \, dx = \rho A(a,t)u(a,t)^2 - \rho A(b,t)u(b,t)^2 + F \tag{5}$$

The applied forces acting on the control volume can be decomposed into body forces, which are assumed to be zero, surface forces acting at the boundary $A(a,t)p(a,t) - A(b,t)p(b,t)$, here is $p(x,t)$ the internal pressure, a side wall pressure force $\int_a^b p\frac{\partial A}{\partial x}\, dx$ and lastly a viscous resistance force, which is taken in literature as $\int_a^b f \, dx$. Where f is the friction force per unit of length in dyne/cm and $p(x,t)$ is the internal pressure at location x and at time t in dyne/cm^2. Therefore

$$F = A(a,t)p(a,t) - A(b,t)p(b,t) + \int_a^b p\frac{\partial A}{\partial x}\, dx + \int_a^b f \, dx \tag{6}$$

Thus

$$\frac{d}{dt}\int_a^b \rho A u \, dx = \rho A(a,t)u(a,t)^2 - \rho A(b,t)u(b,t)^2 + A(a,t)p(a,t) - A(b,t)p(b,t) + \int_a^b p\frac{\partial A}{\partial x}\, dx + \int_a^b f \, dx \tag{7}$$

Noting that the $\frac{d}{dt}$ can be put inside of the integral, ρ is independent of x and t and rewriting the boundary terms as integrals gives

$$\rho\int_a^b \frac{\partial Au}{\partial t}\, dx = -\rho\int_a^b \frac{\partial Au^2}{\partial x}\, dx - \int_a^b \frac{\partial Ap}{\partial x}\, dx + \int_a^b p\frac{\partial A}{\partial x}\, dx + \int_a^b f \, dx \tag{8}$$

A little rewriting then gives

$$\rho\int_a^b \left(\frac{\partial Au}{\partial t} + \frac{\partial Au^2}{\partial x} + \frac{1}{\rho}\frac{\partial Ap}{\partial x} - \frac{p}{\rho}\frac{\partial A}{\partial x} - \frac{f}{\rho}\right) dx = 0 \tag{9}$$

This equation should hold for any a and b, thus the integrand is zero. Thus

$$\frac{\partial Au}{\partial t} + \frac{\partial Au^2}{\partial x} + \frac{1}{\rho}\frac{\partial Ap}{\partial x} - \frac{p}{\rho}\frac{\partial A}{\partial x} - \frac{f}{\rho} = 0 \tag{10}$$

This can be simplified by applying the product rule

$$\frac{\partial Au}{\partial t} = \frac{\partial A}{\partial t}u + A\frac{\partial u}{\partial t}, \quad \frac{\partial Au^2}{\partial x} = \frac{\partial (Au)u}{\partial x} = \frac{\partial Au}{\partial x}u + Au\frac{\partial u}{\partial x}, \quad \frac{\partial Ap}{\partial x} = \frac{\partial A}{\partial x}p + A\frac{\partial p}{\partial x} \tag{11}$$

Substituting this back into the equation and rearranging yields

$$\left(\frac{\partial A}{\partial t} + \frac{\partial Au}{\partial x}\right)u + A\left(\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + \frac{1}{\rho}\frac{\partial p}{\partial x} - \frac{f}{\rho A}\right) = 0 \tag{12}$$

The term in the left brackets is zero by the conservation of mass equation. Similar to Piéro et at [9], we set the resistance term f to be proportional to the fluid velocity u and the cross-sectional area A, so $f = -\rho K_R A u$. Where K_R is a strictly positive quantity and it represents the viscous resistance of the flow per unit length of the tube in poise·cm/g. Then dividing by A, which is allowed since $A > 0$ under physiological conditions, yields

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + \frac{1}{\rho}\frac{\partial p}{\partial x} + K_R u = 0 \tag{13}$$

3.3 Completing the system

The system of equations is completed, by assuming static radial equilibrium in the radial direction of the cylindrical tube and using an algebraic relationship between the pressure of the vessel p and the vessel cross-sectional area A, as in [1], as

$$p = p_{ext} + \beta(\sqrt{A} - \sqrt{A_0}) \tag{14}$$

with

$$\beta = \frac{\sqrt{\pi} h_0 E}{(1-\nu^2) A_0} \tag{15}$$

Where β is a measure of the vessel stiffness in dyne/cm^3. $A_0(x)$ is the cross-sectional area of the vessel in cm^2 and h_0 is the vessel thickness in cm both determined at equilibrium state $(p, u) = (p_{ext}, 0)$. $E(x)$ is the Young modulus in dyne/cm, p_{ext} is the external pressure in dyne/cm^2 and ν is the Poisson ratio. The Poisson's ratio for biological tissue is often taken as $\nu = \frac{1}{2}$, since biological tissue is almost incompressible.

3.4 Conservative form

So far we have derived the following set of nonlinear partial differential equations

$$\begin{cases} \dfrac{\partial A}{\partial t} + \dfrac{\partial Au}{\partial x} = 0 \\ \dfrac{\partial u}{\partial t} + u\dfrac{\partial u}{\partial x} + \dfrac{1}{\rho}\dfrac{\partial p}{\partial x} + K_R u = 0 \end{cases} \tag{16}$$

with

$$p = p_{ext} + \beta(\sqrt{A} - \sqrt{A_0}) \tag{17}$$

Where differential equations inside of the accolade represent the conservation of mass equation and the conservation of momentum equation respectively. We would like to write this system of partial differential equations in the following conservative form with source term

$$\frac{\partial U}{\partial t} + \frac{\partial F(U)}{\partial x} = S(U) \tag{18}$$

Where U is the conserved quantity, $F(U)$ is the flux function and $S(U)$ is the source term. Taking U, $F(U)$ and $S(U)$ as

$$U = \begin{bmatrix} A \\ u \end{bmatrix}, \quad F(U) = \begin{bmatrix} Au \\ \frac{1}{2}u^2 + \frac{p}{\rho} \end{bmatrix}, \quad S(U) = \begin{bmatrix} 0 \\ -K_R u \end{bmatrix} \tag{19}$$

gives the required form of (18)

$$\begin{bmatrix} A \\ u \end{bmatrix}_t + \begin{bmatrix} Au \\ \frac{1}{2}u^2 + \frac{p}{\rho} \end{bmatrix}_x = \begin{bmatrix} 0 \\ -K_R u \end{bmatrix} \tag{20}$$

The subscript denotes taking the partial derivative to the respective variable. Here is $F(U)$ the flux function that will be important when we implement a numerical scheme. To see why this form is called the conservative form, we integrate (18) from a to b to obtain

$$\int_a^b \frac{\partial U}{\partial t}\, dx + \int_a^b \frac{\partial F(U)}{\partial x}\, dx = \int_a^b S(U)\, dx \tag{21}$$

If U is sufficiently smooth, this then gives

$$\frac{\partial}{\partial t}\int_a^b U\, dx = F(U)\big|_a - F(U)\big|_b + \int_a^b S(U)\, dx \tag{22}$$

Thus the conserved quantity U in some control volume between a and b only changes in time due to the fluxes $F(U)$ through the boundaries and due to the sources $S(U)$ inside of the control volume. Hence U is conserved, if there are no sources.

4 Characteristic equations

The nonlinear system of partial differential equations (16) can be transformed into a system of characteristic equations. These characteristic equations and characteristic variables completely describe how the nonlinear system of partial differential equations (16) behaves, while providing important insights in how the system works. They are also of interest when considering the boundary conditions.

4.1 Derivation of characteristic equations

In the following sections we will assume that $p_{ext} = 0$, $\beta = \beta(x)$ and $A_0 = A_0(x)$. We note that applying the chain rule to p (17) yields

$$\frac{\partial p}{\partial x} = \frac{\partial p}{\partial A}\frac{\partial A}{\partial x} + \frac{\partial p}{\partial \beta}\frac{\partial \beta}{\partial x} + \frac{\partial p}{\partial A_0}\frac{\partial A_0}{\partial x} = \frac{\beta}{2\sqrt{A}}\frac{\partial A}{\partial x} + (\sqrt{A} - \sqrt{A_0})\frac{\partial \beta}{\partial x} - \frac{\beta}{2\sqrt{A_0}}\frac{\partial A_0}{\partial x} \tag{23}$$

Substituting this equation into the partial differential system (16) and applying the chain rule, we can write the system in quasi-linear form

$$\frac{\partial U}{\partial t} + H(U)\frac{\partial U}{\partial x} = \begin{bmatrix} A \\ u \end{bmatrix}_t + \begin{bmatrix} u & A \\ \frac{\beta}{2\rho\sqrt{A}} & u \end{bmatrix}\begin{bmatrix} A \\ u \end{bmatrix}_x = \begin{bmatrix} 0 \\ -K_R u - \frac{1}{\rho}\left\{(\sqrt{A} - \sqrt{A_0})\frac{\partial \beta}{\partial x} - \frac{\beta}{2\sqrt{A_0}}\frac{\partial A_0}{\partial x}\right\} \end{bmatrix} \tag{24}$$

Where the flux Jacobian $H(U) = \frac{\partial F(U)}{\partial U}$ is given by

$$H(U) = \begin{bmatrix} u & A \\ \frac{\beta}{2\rho\sqrt{A}} & u \end{bmatrix} \tag{25}$$

This could also be obtained by applying the chain rule to the differential system in conservative from. Since

$$\frac{\partial F(U)}{\partial x} = \frac{\partial F(U)}{\partial U}\frac{\partial U}{\partial x} + \frac{\partial F(U)}{\partial \beta}\frac{\partial \beta}{\partial x} + \frac{\partial F(U)}{\partial A_0}\frac{\partial A_0}{\partial x} \tag{26}$$

The eigenvalues of the flux Jacobian matrix H are $\lambda_{1,2} = u \pm c$ with basic wave speed $c = \sqrt{\frac{\beta}{2\rho}}A^{1/4}$. The basic wave speed c can be interpreted as the velocity of the waves as seen by an observer moving with speed u or as the velocity of the waves if the fluid is at rest, while $\lambda_{1,2}$ are the real velocities of the waves, as seen by an *observer* at rest.

In a paper by Sherwin et al. [1] is found that under physiological conditions, such as $A > 0$ and typical values of velocity u, vessel area A and the elastic parameter β, it holds that $u < c$. Thus the velocity of the blood u is strictly lower than the rest velocity of the waves c, hence the system is subcritical. Furthermore since $u < c$ it follows that $\lambda_1 > 0$ and $\lambda_2 < 0$. Thus the eigenvalues $\lambda_{1,2}$ are real and distinct, such that the system is strictly hyperbolic.

The flux Jacobian matrix H can be diagonalized and this yields

$$H = R\Lambda R^{-1} = \begin{bmatrix} \frac{A}{2c} & -\frac{A}{2c} \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix}\begin{bmatrix} u+c & 0 \\ 0 & u-c \end{bmatrix}\begin{bmatrix} \frac{c}{A} & 1 \\ -\frac{c}{A} & 1 \end{bmatrix} \tag{27}$$

Therefore can (24) be written as

$$\frac{\partial U}{\partial t} + R\Lambda R^{-1}\frac{\partial U}{\partial x} = K(U) \implies R^{-1}\frac{\partial U}{\partial t} + \Lambda R^{-1}\frac{\partial U}{\partial x} = R^{-1}K(U) \tag{28}$$

Where $K(U)$ is given by the right hand side of equation (24). We want to simplify the equations by introducing the characteristic variables $W = [w_1\ w_2]^T$. Note that $R = R(A, x)$ thus we cannot simply use $W = R^{-1}U$ since $W_x = (R^{-1}U)_x = R_x^{-1}U + R^{-1}U_x \neq R^{-1}U_x$. But there is a clever trick we can use to decompose the system into characteristic variables, we can use $\frac{\partial W}{\partial U} = R^{-1}$, as in [1]. The relationship between A, u and w_1, w_2 can then be determined from this Jacobian relation $\frac{\partial W}{\partial U} = R^{-1}$, so

$$\frac{\partial(w_1, w_2)}{\partial(A, u)} = \begin{bmatrix} \frac{\partial w_1}{\partial A} & \frac{\partial w_1}{\partial u} \\ \frac{\partial w_2}{\partial A} & \frac{\partial w_2}{\partial u} \end{bmatrix} = \begin{bmatrix} \frac{c}{A} & 1 \\ -\frac{c}{A} & 1 \end{bmatrix} \tag{29}$$

These differential equations can be integrated separately and this yields

$$w_1 = u + 4c = u + 4A^{1/4}\sqrt{\frac{\beta}{2\rho}}, \quad w_2 = u - 4c = u - 4A^{1/4}\sqrt{\frac{\beta}{2\rho}} \tag{30}$$

These relations can also be inverted and this gives

$$A = \left(\frac{w_1 - w_2}{4}\right)^4 \left(\frac{\rho}{2\beta}\right)^2, \quad u = \frac{w_1 + w_2}{2} \tag{31}$$

Thus according to the Jacobian relation $\frac{\partial W}{\partial U} = R^{-1}$, we have to choose the characteristic variables as

$$W = \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = \begin{bmatrix} u + 4c \\ u - 4c \end{bmatrix} \tag{32}$$

Substituting the Jacobian relation into equation (28) yields

$$\frac{\partial W}{\partial U}\frac{\partial U}{\partial t} + \Lambda \frac{\partial W}{\partial U}\frac{\partial U}{\partial x} = R^{-1}K(U) \tag{33}$$

Using the chain rule we have that

$$\frac{\partial W}{\partial t} = \frac{\partial W}{\partial U}\frac{\partial U}{\partial t} \tag{34}$$

and

$$\frac{\partial W}{\partial x} = \frac{\partial W}{\partial U}\frac{\partial U}{\partial x} + \frac{\partial W}{\partial \beta}\frac{\partial \beta}{\partial x} \implies \frac{\partial W}{\partial U}\frac{\partial U}{\partial x} = \frac{\partial W}{\partial x} - \frac{\partial W}{\partial \beta}\frac{\partial \beta}{\partial x} \tag{35}$$

Substituting the above identities into equation (33), it then follows that

$$\frac{\partial W}{\partial t} + \Lambda \frac{\partial W}{\partial x} = R^{-1}K + \Lambda \frac{\partial W}{\partial \beta}\frac{\partial \beta}{\partial x} \tag{36}$$

Where Λ, R^{-1} and K should be written in terms of w_1 and w_2 using equations (31) to finish the transformation from the original system to the characteristic system. The characteristic variables w_1 and w_2 are by some authors called the Riemann invariants.

If there is a region where the material properties are (almost) constant, so $\beta(x) = \beta$ and $A_0(x) = A_0$, then in that region we have that the derivatives of β and A_0 vanish. Furthermore, if we assume inviscid flow, hence there is no viscous resistance so $K_R = 0$, then it follows that in that region $K(U) = 0$. Thus if we have inviscid flow and a region with constant material properties, then the RHS of equation (36) becomes zero (or very small). Thus

$$\frac{\partial W}{\partial t} + \Lambda \frac{\partial W}{\partial x} = 0 \tag{37}$$

Since Λ is a diagonal matrix this reads componentwise

$$\frac{\partial w_1}{\partial t} + \lambda_1 \frac{\partial w_1}{\partial x} = 0, \quad \frac{\partial w_2}{\partial t} + \lambda_2 \frac{\partial w_2}{\partial x} = 0 \tag{38}$$

We note that $\lambda_1 = u + c = \frac{w_1+w_2}{2} + \sqrt{\frac{\beta}{2\rho}}\frac{w_1-w_2}{4}\sqrt{\frac{\rho}{2\beta}} = \frac{5w_1+3w_2}{8}$ and $\lambda_2 = \frac{3w_1+5w_2}{8}$. Thus the PDEs for the characteristic variables reduce to a coupled system of quasilinear PDEs where the material properties are constant. If we linearise these quasilinear PDEs, i.e. take $\lambda_{1,2}$ constant, then we would obtain two scalar transport equations moving with speed $\lambda_{1,2}$ respectively.

5 Linear system

Before we solve the strictly hyperbolic nonlinear system (16), we will first look at a linear test case. The linear test case will be built from the ground up, to mimic the nonlinear system as closely as possible. We will solve this linear system both analytically and numerically to compare the results and to obtain insight into hyperbolic systems of partial differential equations. Note that the variables defined here are all different from the ones used in the rest of the thesis.

5.1 Linear system and characteristic variables

We have chosen the following linear system for $q(x,t) = [q_1(x,t)\ q_2(x,t)]^T$

$$\frac{\partial q}{\partial t} + A\frac{\partial q}{\partial x} = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}_t + \begin{bmatrix} a & b \\ b & a \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}_x = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{39}$$

Note that matrix A is symmetric and hence it has 2 distinct real eigenvalues if $b \neq 0$, such that the system is strictly hyperbolic. The eigenvalues of A are $\lambda_{1,2} = a \pm b$. Since we want to mimic the quasilinear system we choose $b > a$ such that we have two waves travelling in opposite directions corresponding to $\lambda_1 > 0$ and $\lambda_2 < 0$. Matrix A can be diagonalized and this yields

$$A = R\Lambda R^{-1} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} a+b & 0 \\ 0 & a-b \end{bmatrix} \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \tag{40}$$

Therefore

$$q_t + R\Lambda R^{-1} q_x = 0 \Rightarrow R^{-1} q_t + \Lambda R^{-1} q_x = 0 \tag{41}$$

Now defining the characteristic variables as

$$w = R^{-1} q \tag{42}$$

does work because $w_t = R^{-1} q_t$ and $w_x = R^{-1} q_x$. Thus we get

$$w_t + \Lambda w_x = 0 \tag{43}$$

Using that Λ is a diagonal matrix the system decouples in the characteristic variables as

$$\frac{\partial w_1}{\partial t} + \lambda_1 \frac{\partial w_1}{\partial x} = 0 \text{ and } \frac{\partial w_2}{\partial t} + \lambda_2 \frac{\partial w_2}{\partial x} = 0 \tag{44}$$

These are just scalar transport equations.

5.2 Well posed boundary conditions

To obtain a well posed mathematical problem we have to take special care when dealing with the boundary conditions. As said before, we have chosen $\lambda_1 > 0$ such that it represents a wave travelling to the right and $\lambda_2 < 0$ to represent a wave travelling to the left. Thus, at the left boundary w_1 will flow into the domain while w_2 is leaving the domain. Therefore, we have to prescribe w_1 at the left boundary as it is the inflow variable. To determine which original variable we can prescribe at the left boundary, we follow [6] and compute

$$J = \frac{\partial (q_1, q_2)}{\partial w_1} = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix} \tag{45}$$

where we have used that $q = Rw$. Because we have exactly one eigenvalue greater than zero, λ_1, and because each element of J is greater than zero, we can choose either q_1 or q_2 to be prescribed at the left boundary.

The right boundary is handled in a similar way. At the right boundary, w_2 enters the domain while w_1 leaves the domain. Thus we need to prescribe w_2 at the right boundary. The same analysis as above can be done and it is found that prescribing q_1 or q_2 is allowed.

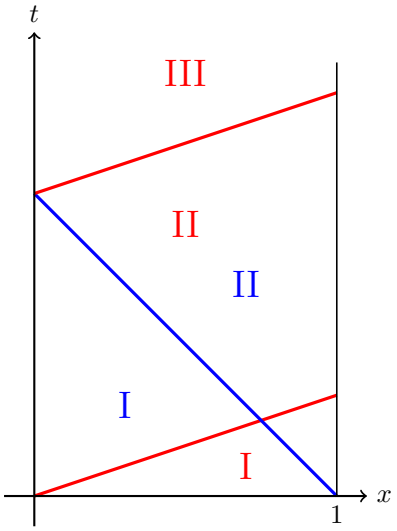

Figure 2: Domain of the solution of the linear test case with the characteristic base curves shown in red and blue for λ_1 and λ_2 respectively. The regions where the characteristic variables are determinable in a specific way are colour-coded: Red for w_1 and blue for w_2

5.3 Problem formulation

We will solve the following problem on the spatial domain $0 < x < 1$ and temporal domain $t > 0$ governed by the linear system (39), with initial conditions and boundary conditions given as

$$\text{IC: } q(x,0) = \begin{bmatrix} \overset{\circ}{q}_1(x) \\ \overset{\circ}{q}_2(x) \end{bmatrix}, \quad \text{BC}_1 : \; q_1(0,t) = \bar{q}_1(t), \quad \text{BC}_2\text{: } w_2(1,t) = 0 \tag{46}$$

A finite spatial domain and a semi-infinite temporal domain is chosen since it is similar to the domain of a blood vessel. The left boundary condition BC_1 is some prescribed value for q_1 while q_2 is determined by the domain. The right boundary condition BC_2 is a non-reflective boundary condition. It is a numerical boundary condition, not a physical one, in the sense that the computational domain cannot be infinite due to hardware limitations and therefore a border has to be drawn somewhere. This boundary condition has to be chosen such that it does not impact the domain of interest, hence a non-reflective boundary is chosen.

5.4 Analytical approach

Using the characteristic variables as described above, one can use the method of characteristics to get the following analytical solution on the domain $0 < x < 1$ and $t > 0$

$$\begin{aligned} q_1(x,t) &= \frac{1}{\sqrt{2}}\{w_1(x,t) - w_2(x,t)\} \\ q_2(x,t) &= \frac{1}{\sqrt{2}}\{w_1(x,t) + w_2(x,t)\} \end{aligned} \tag{47}$$

where

$$w_1(x,t) = \begin{cases} \frac{1}{\sqrt{2}}\left\{\mathring{q}_1(x-\lambda_1 t) + \mathring{q}_2(x-\lambda_1 t)\right\}, & 0 < t < \frac{x}{\lambda_1} \quad \text{(I)} \\ \sqrt{2}\,\bar{q}_1\left(t - \frac{x}{\lambda_1}\right) + \frac{1}{\sqrt{2}}\left\{-\mathring{q}_1\left(\lambda_2\left[\frac{x}{\lambda_1} - t\right]\right) + \mathring{q}_2\left(\lambda_2\left[\frac{x}{\lambda_1} - t\right]\right)\right\}, & \frac{x}{\lambda_1} < t < \frac{x}{\lambda_1} - \frac{1}{\lambda_2} \quad \text{(II)} \\ \sqrt{2}\,\bar{q}_1\left(t - \frac{x}{\lambda_1}\right), & t > \frac{x}{\lambda_1} - \frac{1}{\lambda_2} \quad \text{(III)} \end{cases}$$

$$w_2(x,t) = \begin{cases} \frac{1}{\sqrt{2}}\left\{-\mathring{q}_1(x-\lambda_2 t) + \mathring{q}_2(x-\lambda_2 t)\right\}, & 0 < t < \frac{x-1}{\lambda_2} \quad \text{(I)} \\ 0, & t > \frac{x-1}{\lambda_2} \quad \text{(II)} \end{cases} \tag{48}$$

Note that $\lambda_2 < 0$ such that $-\frac{1}{\lambda_2} > 0$. The domain of the solution is shown in Figure 2. In this figure, each colour-coded region plus the black lines show a region in which the characteristic variable is determinable in a specific way, as explained below. These regions also correspond to the regions mentioned in equation (48).

- Blue I: w_2 is determined by the initial conditions.
- Blue II: w_2 is determined by the right boundary condition.
- Red I: w_1 is determined by the initial conditions.
- Red II: w_1 is a combination of w_2 from I and a part of the left boundary condition.
- Red III: w_1 is determined by a part of the left boundary condition.

Once the solution for w_1 and w_2 is known analytically it can be transformed back to the original system by inverting the definition of the characteristic variables, see equation (42).

5.5 Numerical approach

The linear system with initial conditions and boundary conditions described above can also be solved numerically. The nonlinear system of PDEs (16) and the linear system of PDEs (39) are both strictly Hyperbolic PDEs. Hyperbolic PDEs describe a wide range of wave propagation and transport phenomena. In fact, hyperbolic PDEs are closely related to conservation laws. Recall that the hyperbolic system of PDEs describing the flow of blood in the arteries was derived from the physical principles of conservation of the mass and conservation of momentum. Finite Volume Methods (FVM) have the nice property that the resulting solution satisfies the conservation of quantities, such as mass, momentum or/and energy. It thus seems very natural to use FVMs for hyperbolic PDEs.

Following LeVeque [3], the spatial domain is discretized into equidistant small volumes of length Δx. These cells are centred around each node x_i, thus the i-th cell is defined on $(x_i - \frac{1}{2}\Delta x,\, x_i + \frac{1}{2}\Delta x)$ and is denoted as C_i. We get the following scheme for the cell average $Q_i^n \approx \frac{1}{\Delta x}\int_{C_i} q(x,t^n)dx$

$$Q_i^{n+1} = Q_i^n - \frac{\Delta t}{\Delta x}\left(A^+\Delta Q_{i-1/2}^n + A^-\Delta Q_{i+1/2}^n\right) \tag{49}$$

Where the superscript n denotes the n-th time step and the subscript i denotes the i-th spatial node. The entity $A^+\Delta Q_{i-1/2}^n$ is the fluctuation, and it represents the net effect of all right going waves from the border between the i-th and the $i-1$-th cell. Similarly $A^-\Delta Q_{i+1/2}^n$ is the net effect of all left going waves from the border between the i-th and the $i+1$-th cell. This equation does make sense, since every cell has two borders in this one-dimensional problem, a left and right border. The only way in which Q can change, assuming no production, is due to right moving waves coming from the left border, and left moving waves coming from the right border. Which is exactly what this equation says.

The fluctuations, in the linear case, can be calculated using

$$A^+\Delta Q_{i-1/2}^n = R\Lambda^+R^{-1}(Q_i^n - Q_{i-1}^n) \tag{50}$$

$$A^-\Delta Q_{i+1/2}^n = R\Lambda^-R^{-1}(Q_{i+1}^n - Q_i^n) \tag{51}$$

Where the matrices R and R^{-1} are implicitly defined in equation (40), $\Lambda^+ = \text{diag}(\max(\lambda, 0))$ and $\Lambda^- = \text{diag}(\min(\lambda, 0))$, so

$$\Lambda^+ = \begin{bmatrix} \lambda_1 & 0 \\ 0 & 0 \end{bmatrix}, \quad \Lambda^- = \begin{bmatrix} 0 & 0 \\ 0 & \lambda_2 \end{bmatrix} \tag{52}$$

Thus Λ^+ 'picks' the positive eigenvalues and sets the negative eigenvalues to zero. Conversely Λ^- 'picks' the negative eigenvalues and sets the positive eigenvalues to zero. We see that equation (50), the right moving fluctuations, decomposes the difference between Q_i^n and Q_{i-1}^n into the basis of matrix A's eigenvectors using R^{-1}, after which Λ^+ only allows the right moving waves to propagate and removes the left going waves and lastly we transform back to the original system using R.

5.5.1 Boundary conditions

Equation (49) describes how to handle the linear system on the interior domain. What is left to cover are the boundary conditions. We start with the right boundary condition. The easiest way to handle boundary conditions in numerical schemes is by the use of ghost cells and then using the same numerical scheme, according to LeVeque [3]. The right boundary is an outflow boundary and was chosen as $w_2(1, t) = 0$ for all t. w_2 is the characteristic variable corresponding to the left flowing waves, therefore, we want the ghost cell after the right most physical cell to have the same w_2 component as the right most physical cell. An easy way to accomplish this goal is by using zero-th order extrapolation of the right most physical cell since w_1 of the ghost cell does not influence the computational domain. Zeroth order extrapolation means that the ghost cell gets the values from the last physical cell. Let's call the right most physical cell m then the ghost cell is the m+1-th cell. For the m-th cell it holds that

$$Q_m^{n+1} = Q_m^n - \frac{\Delta t}{\Delta x}\left(A^+ \Delta Q_{m-1/2}^n + A^- \Delta Q_{m+1/2}^n\right) \tag{53}$$

We thus see that the problem arises from the $A^- \Delta Q_{m+1/2}^n$ term since we do not have a physical m+1-th cell. Using the technique described above we get $R\Lambda^- R^{-1}(Q_{m+1}^n - Q_m^n) = R\Lambda^- R^{-1}(Q_m^n - Q_m^n) = 0$. Thus nothing is flowing into the computational domain from the right boundary, as required by the right boundary condition.

The left boundary condition is a bit more difficult and we need to use characteristic decomposition to obtain the correct result. For the left boundary condition we chose to prescribe $q_1(0, t) = \bar{q}_1(t)$. We again will use the ghost cell approach but this time for the left boundary, thus for the first physical cell, $i = 1$, we have

$$Q_1^{n+1} = Q_1^n - \frac{\Delta t}{\Delta x}\left(A^+ \Delta Q_{1/2}^n + A^- \Delta Q_{3/2}^n\right) \tag{54}$$

The ghost cell will be the 0-th cell and thus we can alter the inflow fluctuations $A^+ \Delta Q_{1/2}^n = A^+(Q_1^n - Q_0^n)$ using Q_0^n. Since w_2 is determined inside of the computational domain we can only prescribe inflowing characteristic w_1 at $x = 0$. Explicitly calculating the relation between q_1 and w_1, w_2 from the definition of the characteristic variables, yields

$$w = R^{-1}q \implies q = Rw \implies \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}}(w_1 - w_2) \\ \frac{1}{\sqrt{2}}(w_1 + w_2) \end{bmatrix} \tag{55}$$

Thus $q_1 = \frac{1}{\sqrt{2}}(w_1 - w_2) \implies w_1 = w_2 + \sqrt{2}q_1$, therefore at $x = 0$ we have $w_1(0, t) = w_2(0, t) + \sqrt{2}\bar{q}_1(t)$ where $w_2(0, t)$ is determined from the domain. Thus for the ghost cell, we have $Q_0^n = R\tilde{w} = R[w_2(0, n\Delta t) + \sqrt{2}\bar{q}_1(n\Delta t) \;\; w_2(0, n\Delta t)]^T$ where $w_2(0, n\Delta t) \approx \frac{1}{\sqrt{2}}[-1 \;\; 1]Q_1^n$.

5.5.2 Stability

A necessary condition for stability is the CFL condition. The CFL condition is given by

$$\frac{|\lambda_i|\Delta t}{\Delta x} \leq 1, \quad i \in \{1, 2\} \tag{56}$$

The CFL condition basically assures that the waves, originating at the cell interface cannot pass into another cell within the given time frame Δt. Since $s = vt$, so $s = |\lambda_i|\Delta t \leq \Delta x$ rearranging would give the CFL condition.

5.6 Results of the linear test case

Having developed all the necessary theory, we can apply it to the following problem

$$\begin{bmatrix} q_1 \\ q_2 \end{bmatrix}_t + \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}_x = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{57}$$

i.e. we have chosen $a = 1$ and $b = 2$, such that the eigenvalues become $\lambda_1 = 3$ and $\lambda_2 = -1$. The initial condition and boundary conditions read

$$\text{IC: } q(x,0) = \begin{bmatrix} 1 + e^{-100(x-1/2)^2} \\ 1 \end{bmatrix}, \quad \text{BC}_1\text{: } q_1(0,t) = 1 + \frac{1}{10}\sin\left(\frac{4\pi}{3}t\right), \quad \text{BC}_2\text{: } w_2(1,t) = 0 \tag{58}$$

Thus for q_1 we have a centred Gaussian as initial condition and a sine as left boundary condition. We have chosen for such a small value for the variation, or conversely such a big value in the exponent of the Gaussian, to obtain a localised peak, which is easier to study. The solution of q_1 obtained with the numerical scheme and analytical technique are plotted in Figure 3. We are neglecting q_2 because the overall behaviour of the system can be analysed using q_1 only.

We will first look at the overall behaviour of the solution before we compare the analytical and numerical solutions. Looking at Figure 3, we see that at $t = 0$, q_1 is a Gaussian. As is the case for the wave equation, this perturbation is composed of two waves, one wave moving in the positive x direction and a wave propagating towards the negative x direction. For the scalar wave equation the perturbations move at the same velocity in both directions, but in this more general problem, the waves move at their respective eigenvalue velocity. For example, the disturbance peak travelling towards the negative x direction, starts at $x = 0.5$ and hits the boundary at $x = 0$ at $t = 0.5$ as can be seen in the lower right image from Figure 3, thus $v = \frac{\Delta x}{\Delta t} = -1$ as expected from $\lambda_2 = -1$.

We also see that the left boundary condition closely resembles a period of a sine, while the right boundary has no effect on the solution, as required by the boundary conditions.

The biggest difference between the analytical and numerical solution is that the numerical solution is a 'smeared out' version of the analytical solution. For example, the wave moving towards the negative x direction has a lower maximum than the analytical solution and the crest is less narrow in comparison to the exact solution after reflecting off the $x = 0$ boundary. This is a well known effect of using Godunov's first order method.

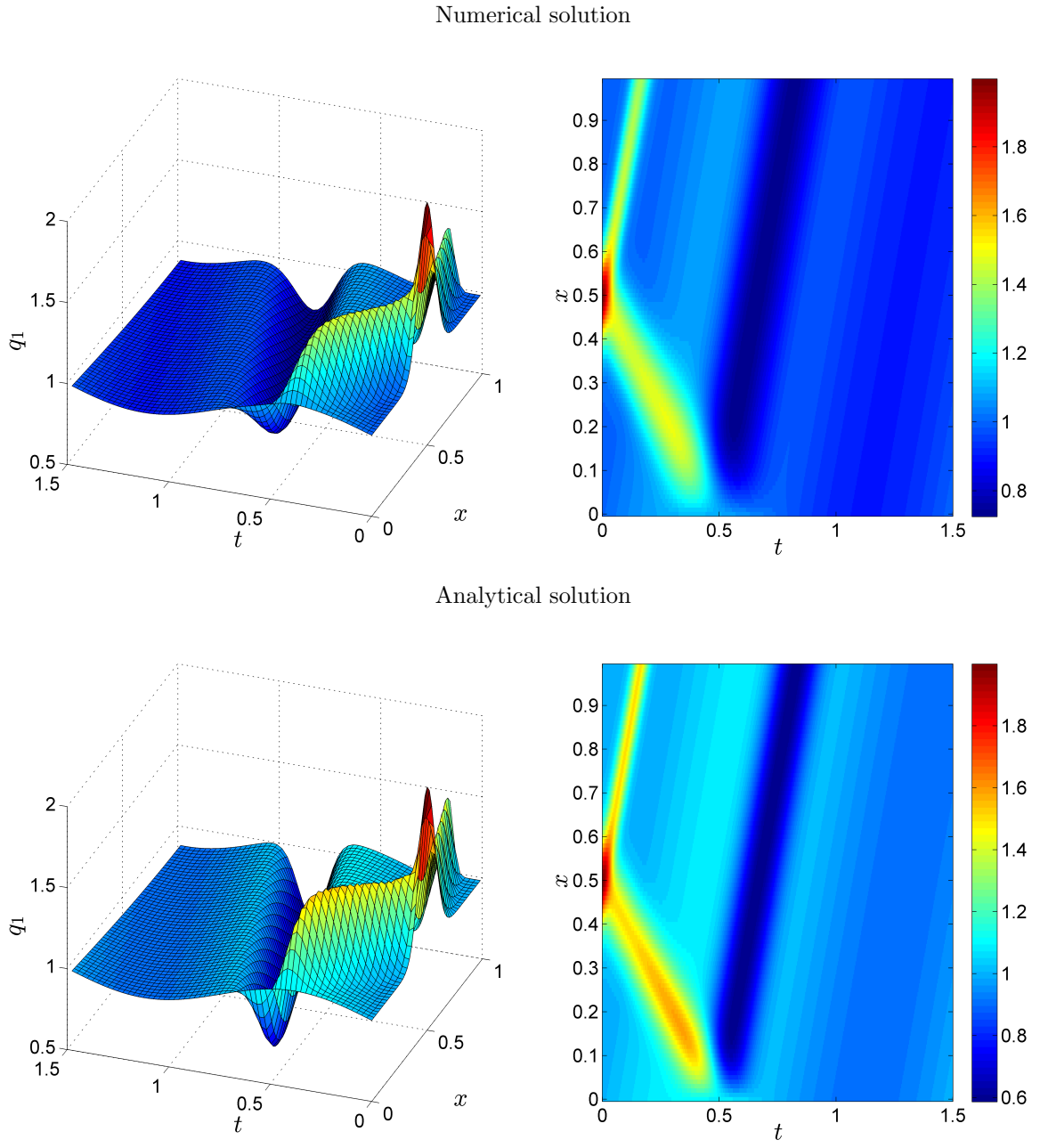

Figure 3: The q_1 component of the solution of problem (57) with IC and BC given as (58) is plotted for x and t. The first row of Figures corresponds to the numerical solution and the second row is the analytical solution. The right column of Figures shows the intensity corresponding to q_1 for the numerical solution and analytical solution respectively.

6 The nonlinear Partial Differential Equations

To be able to solve the nonlinear system PDEs we linearize the system of equations at each cell interface.

6.1 Linearisation of the nonlinear system

We assume that the friction term K_R is small and negligible. We, therefore, set K_R to zero and in conservative form (18), the source term $S(U)$ drops out. We are left with the following conservative form PDE

$$\frac{\partial U}{\partial t} + \frac{\partial F(U,x)}{\partial x} = 0, \quad \begin{bmatrix} A \\ u \end{bmatrix}_t + \begin{bmatrix} Au \\ \frac{1}{2}u^2 + \frac{p}{\rho} \end{bmatrix}_x = 0, \quad U = \begin{bmatrix} A \\ u \end{bmatrix}, \quad F(U,x) = \begin{bmatrix} Au \\ \frac{1}{2}u^2 + \frac{p}{\rho} \end{bmatrix} \tag{59}$$

Where we have made the spatial dependence of the flux function more explicit, $F(U) = F(U,x)$.

We will use a Finite Volume Method (FVM) to solve this problem, as we did with the linear test case. LeVeque recommends in his book [3], which is about solving hyperbolic PDEs with FVMs, that for solving hyperbolic PDEs with spatially varying flux functions, one should use methods based on flux difference splitting. We will therefore follow Bale and LeVeque, as in [4], to linearize (59).

Let us first reiterate some FVM terminology. The spatial domain is discretized into equidistant volumes with length Δx. These cells are centred around each node x_i, thus the i-th cell is defined on $(x_i - \frac{1}{2}\Delta x,\, x_i + \frac{1}{2}\Delta x)$ and is denoted as C_i. Furthermore the cell interface between the i-th and $i-1$-th cell be denoted by $x_{i-1/2}$. Lastly, let the average value of U in cell i at time t^n be denoted as $U_i^n \approx \frac{1}{\Delta x}\int_{C_i} U(x,t^n)\,dx$. So U without subscripts and superscripts is the continuous solution while U with subscripts is the cell average.

We will consider the Riemann problem, which is an initial value problem composed of a conservation law together with piecewise *constant* initial data with a single discontinuity. The goal of the Riemann problem is to determine the solution on a domain.

We will adjust the Riemann problem to fit the finite volume approach. Imagine that we have the i-th cell and the $i-1$-th cell. Then we have the following 'initial' data at time t^n

$$U(x,t^n) = \begin{cases} U_{i-1}^n, & x < x_{i-1/2} \\ U_i^n, & x > x_{i-1/2} \end{cases} \tag{60}$$

Where the discontinuity is located at the common cell border $x_{i-1/2}$. The aim is to use the exact solution of this Riemann problem to approximate the new cell averages Δt later, so U_{i-1}^{n+1} and U_i^{n+1}.

We will use *cell centred* flux functions. Where the flux function $F(U,x)$ is discretized to be constant within each cell and generally discontinuous at each cell interface. Let the discretization of the flux function in the i-th cell be denoted as $F_i(U) = F(U,x_i)$.

The (generalised) Riemann problem at the $i-1/2$-th cell interface is then governed by

$$\begin{aligned} \frac{\partial U}{\partial t} + \frac{\partial F_{i-1}(U)}{\partial x} = 0, \quad & \text{if } x < x_{i-1/2} \\ \frac{\partial U}{\partial t} + \frac{\partial F_i(U)}{\partial x} = 0, \quad & \text{if } x > x_{i-1/2} \end{aligned} \tag{61}$$

Where the CFL condition causes the equations above to only be valid for the $i-1$-th and i-th cell, so in the region $x_{i-3/2} < x < x_{i+1/2}$.

In Figure 4 the domain of the solution of the Riemann problem is shown between t^n and t^{n+1}. The line segment from $x_{i-3/2}$ until $x_{i-1/2}$ is the $i-1$-th cell and similarly the line segment between $x_{i-1/2}$ and $x_{i+1/2}$ represent the i-th cell. The interface between the $i-1$-th cell and the i-th cell is highlighted with a dashed line and within each suggested enclosed area the value of U constant.

Next is an explanation of the construction of Figure 4 given. We expect the flux to be conserved across the cell interface $x_{i-1/2}$ for bounded solutions. Therefore we have

$$F_{i-1}(U_{i-1/2}^l) = F_i(U_{i-1/2}^r) \tag{62}$$

Where $U_{i-1/2}^l$ and $U_{i-1/2}^r$ are U just left and right of $x_{i-1/2}$ respectively. We have that $F_{i-1}(U_{i-1/2}^l) = F_i(U_{i-1/2}^r)$ and generally that $F_{i-1}(U) \neq F_i(U)$ for all U. Thus it follows that generally $U_{i-1/2}^l \neq U_{i-1/2}^r$.

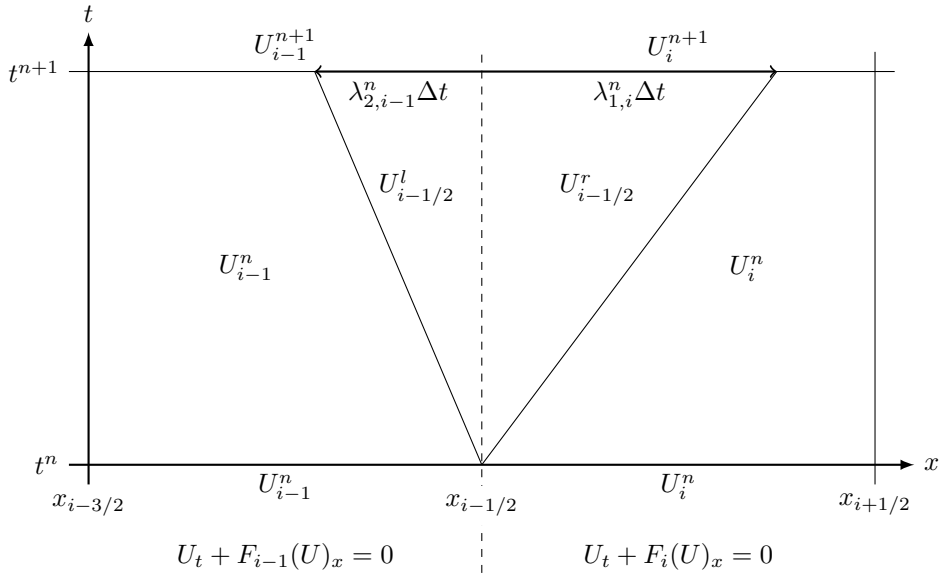

Figure 4: Domain of the solution of the Riemann problem

Hence there is generally a stationary discontinuity in U at cell interface $x_{i-1/2}$, when dealing with spatially varying flux function and using cell centred flux functions.

Note that applying the chain rule to (61) yields

$$\begin{aligned} \frac{\partial U}{\partial t} + \frac{\partial F_{i-1}(U)}{\partial U}\frac{\partial U}{\partial x} = 0, \quad \text{if } x < x_{i-1/2} \\ \frac{\partial U}{\partial t} + \frac{\partial F_i(U)}{\partial U}\frac{\partial U}{\partial x} = 0, \quad \text{if } x > x_{i-1/2} \end{aligned} \tag{63}$$

Where $\frac{\partial F_i(U)}{\partial U} = H(U, x_i) = H_i(U)$ is the flux Jacobian, see (25). The source terms from (24) drop out because in the two regions considered $\beta(x)$ and $A_0(x)$ are both constant and hence their derivatives vanish.

To linearize the problem further, we assume that the flux Jacobian is constant within each cell, so $H_i(U) = H_i(U_i^n) = H_i^n$. This way we obtain two linear systems of PDEs for this Riemann problem.

For a 2×2 strictly hyperbolic *linear* system of PDEs, we know that there are *two* characteristic variables with corresponding *constant* eigenvector and *constant* eigenvalue, as we also have seen in the linear test case. At $t = t^n$ we have piecewise constant initial data, namely U_{i-1}^n and U_i^n, and piecewise constant material properties, we then expect the characteristic variables to be piecewise constant as well. Thus the two characteristic variables both have a discontinuity at $x_{i-1/2}$ at $t = t^n$. Both the characteristic variables propagate at their respective *constant* speed for $t > t_n$. Thus the discontinuity at $x_{i-1/2}$ propagates with speed λ_1 with the first characteristic variable as well as with speed λ_2 with the second characteristic variable. Hence U, which is composed of the two characteristic variables, consists of two more discontinuities.

These discontinuities are the so called waves and they travel at their respective characteristic speed λ_j as illustrated in Figure 4 with the sloped lines which have travelled $s = v \cdot t \implies s = \lambda_j \Delta t$ in Δt time.

The eigenvectors and eigenvalues of the flux Jacobian $\frac{\partial F(U)}{\partial U} = H$ are

$$\lambda_1 = u + c, \quad r_1 = \begin{bmatrix} \frac{A}{2c} \\ \frac{1}{2} \end{bmatrix}, \quad \lambda_2 = u - c, \quad r_2 = \begin{bmatrix} -\frac{A}{2c} \\ \frac{1}{2} \end{bmatrix} \tag{64}$$

with basic wave speed $c = \sqrt{\frac{\beta}{2\rho}} A^{1/4}$, see equation (25) for the definition of the flux Jacobian H.

Since $\lambda_1 > 0$ it follows that the corresponding wave travels towards the positive x direction and thus inside the i-th cell. Thus for the first eigenvalue and vector we will be using the values of the i-th cell, so

$$\lambda_{1,i-1/2}^n = \lambda_{1,i}^n, \quad r_{1,i-1/2}^n = r_{1,i}^n \tag{65}$$

Conversely, since $\lambda_2 < 0$ it follows that the corresponding 'second' wave travels towards to negative x direction, and hence within the $i-1$-th cell. Therefore, we use the values of U^n_{i-1} for the second wave. Thus

$$\lambda^n_{2,i-1/2} = \lambda^n_{2,i-1}, \quad r^n_{2,i-1/2} = r^n_{2,i-1} \tag{66}$$

Once we have the strength of these three waves or consequently the values of U inside the cells between t^n and t^{n+1}, we have the solution of the Riemann problem of the linearised system. These effects can at t^{n+1} be averaged per cell to obtain a new estimate for the cell average at t^{n+1} and this new averaged value is the new value of the cell used for the next time step.

Finding the values of U directly left and right of to $x_{i-1/2}$ respectively $U^l_{i-1/2}$ and $U^r_{i-1/2}$, is a difficult task. We therefore will use that the flux is continuous at $x_{i-1/2}$ and instead of decomposing the difference $U_i - U_{i-1}$, we will decompose the difference in the flux $F_i(U^n_i) - F_{i-1}(U^n_{i-1})$ into so called f-waves, the flux difference splitting. The first f-wave $\mathcal{Z}^n_{1,i-1/2}$ is proportional to the first eigenvector $r^n_{1,i}$ and the second f-wave $\mathcal{Z}^n_{2,i-1/2}$ is proportional to the second eigenvector $r^n_{2,i-1}$. We therefore have

$$\begin{aligned} F_i(U^n_i) - F_{i-1}(U^n_{i-1}) &= \mathcal{Z}^n_{1,i-1/2} + \mathcal{Z}^n_{2,i-1/2} = \beta^n_{1,i-1/2} r^n_{1,i} + \beta^n_{2,i-1/2} r^n_{2,i-1} \\ &= R^n_{i-1/2}\beta^n_{i-1/2} = \frac{1}{2}\begin{bmatrix} \frac{A^n_i}{c^n_i} & -\frac{A^n_{i-1}}{c^n_{i-1}} \\ 1 & 1 \end{bmatrix} \begin{bmatrix} \beta^n_{1,i-1/2} \\ \beta^n_{2,i-1/2} \end{bmatrix} \end{aligned} \tag{67}$$

Where $\beta^n_{j,i-1/2}$ is the relative strength of the $r^n_{j,i-1/2}$ eigenvector and $R^n_{i-1/2}$ is the linearised eigenvector matrix. The vector $\mathcal{Z}^n_{j,i-1/2} = \beta^n_{j,i-1/2} r^n_{j,i-1/2}$ is the j-th f-wave.

Solving for $\beta^n_{i-1/2}$ gives

$$\begin{bmatrix} \beta^n_{1,i-1/2} \\ \beta^n_{2,i-1/2} \end{bmatrix} = \frac{2}{\frac{A^n_i}{c^n_i} + \frac{A^n_{i-1}}{c^n_{i-1}}} \begin{bmatrix} 1 & \frac{A^n_{i-1}}{c^n_{i-1}} \\ -1 & \frac{A^n_i}{c^n_i} \end{bmatrix} (F_i(U^n_i) - F_{i-1}(U^n_{i-1})) \tag{68}$$

We can incorporate the f-waves into the fluctuations formalism in the following way

$$R^+ \Delta U^n_{i-1/2} = \mathcal{Z}^n_{1,i-1/2} \tag{69}$$

$$R^- \Delta U^n_{i-1/2} = \mathcal{Z}^n_{2,i-1/2} \tag{70}$$

Thus we obtain the following explicit method, based in flux difference splitting

$$U^{n+1}_i = U^n_i - \frac{\Delta t}{\Delta x}\left(R^+ \Delta U^n_{i-1/2} + R^- \Delta U^n_{i+1/2}\right) = U^n_i - \frac{\Delta t}{\Delta x}\left(\mathcal{Z}^n_{1,i-1/2} + \mathcal{Z}^n_{2,i+1/2}\right) \tag{71}$$

6.2 High resolution methods

As we have seen before in the linear test case, the wave propagation (flux differencing algorithm) algorithm has the tendency to smear out the solution. Furthermore, if the solution contains a shock, discontinuities or sharp changes then high order spatial discretization could lead to spurious oscillations. In fact, Godunov's order barrier theorem says that linear methods cannot provide non-oscillatory solutions higher than first order [7]. In order to obtain a high resolution method, we thus have to use a nonlinear method. One way to obtain high resolution is by the use of flux limiters. Flux limiters 'switch' between two different spatial discretization schemes. If the solution contains a steep gradient then the flux limiter will use a first order scheme to prevent oscillations at that point. If the solution is smooth enough then a higher order spatial discretization can be used to obtain high resolution. The switching between the two schemes is not as binary as portrayed here, it is more like a spectrum between the two discretization schemes which depends on the strength of the neighbouring waves.

The high resolution method can be formulated as follows

$$U^{n+1}_i = U^n_i - \frac{\Delta t}{\Delta x}\left(R^+ \Delta U^n_{i-1/2} + R^- \Delta U^n_{i+1/2}\right) - \frac{\Delta t}{\Delta x}\left(\tilde{\mathcal{F}}^n_{i+1/2} - \tilde{\mathcal{F}}^n_{i-1/2}\right) \tag{72}$$

Where $\tilde{\mathcal{F}}^n_{i-1/2}$ is a correction flux and it is given by

$$\tilde{\mathcal{F}}^n_{i-1/2} = \frac{1}{2}\sum_{p=1}^{2} \text{sign}(\lambda^n_{p,i-1/2}) \left(1 - \frac{\Delta t}{\Delta x}|\lambda^n_{p,i-1/2}|\right) \tilde{\mathcal{Z}}^n_{p,i-1/2} \tag{73}$$

With $\tilde{\mathcal{Z}}^n_{p,i-1/2}$ a limited version of $\mathcal{Z}^n_{p,i-1/2}$, defined as

$$\tilde{\mathcal{Z}}^n_{p,i-1/2} = \phi(\theta^n_{p,i-1/2})\mathcal{Z}^n_{p,i-1/2} \tag{74}$$

Here is $\phi(\theta)$ the chosen flux limiter, which is preferably second order Total Variation Diminishing (TVD) to guarantee the stability of the scheme. We will be using the minmod limiter, which is second order TVD and it is defined as

$$\phi_{mm}(\theta) = \max\left\{0, \min\left\{2\theta, \frac{2+\theta}{3}, 2\right\}\right\} \tag{75}$$

The argument of the flux limiter $\theta^n_{p,i-1/2}$ is a measure of the strength of the waves to come (upwind) relative to $\mathcal{Z}^n_{p,i-1/2}$, which of course depends on the propagating direction of the waves $\lambda^n_{p,i-1/2}$. LeVeque proposed the following equation to determine this wave strength, if the solution changes rapidly then more sophisticated methods have to be used, see for example Lax and Liu [8].

$$\theta^n_{p,i-1/2} = \frac{\mathcal{Z}^n_{p,I-1/2} \cdot \mathcal{Z}^n_{p,i-1/2}}{\mathcal{Z}^n_{p,i-1/2} \cdot \mathcal{Z}^n_{p,i-1/2}} \tag{76}$$

The dot represents the dot product between two vectors. Where

$$I = \begin{cases} i-1 & \text{if } \lambda^n_{p,i-1/2} > 0 \\ i+1 & \text{if } \lambda^n_{p,i-1/2} < 0 \end{cases} \tag{77}$$

6.3 Boundary conditions

In this section we do the same analysis as in section 5.2, to determine which variables can be prescribed at which boundary, to obtain a well posed mathematical problem. Note that in that section, every variable is different from the rest of the thesis. The characteristic variable w_1 corresponds to $\lambda_1 > 0$, w_1 therefore propagates to the positive x-direction, i.e. to the right. Since $\lambda_2 < 0$ it follows that w_2 travels to the left. Thus the inflow characteristic variable at the left and right boundary are respectively w_1 and w_2. We therefore have to prescribe w_1 at the left boundary and w_2 at the right boundary. To determine which original variable, A or u, we have to prescribe at the right boundary, we compute

$$J_{w_1} = \frac{\partial(A,u)}{\partial w_1} = \begin{bmatrix} (\frac{\rho}{2\beta})^2 (\frac{w_1 - w_2}{4})^3 \\ \frac{1}{2} \end{bmatrix} \tag{78}$$

Where equation (31) is used.

Similarly, for the right boundary we compute

$$J_{w_2} = \frac{\partial(A,u)}{\partial w_2} = \begin{bmatrix} -(\frac{\rho}{2\beta})^2 (\frac{w_1 - w_2}{4})^3 \\ \frac{1}{2} \end{bmatrix} \tag{79}$$

Where we again have used (31). The upper element of J_{w_j} corresponds to A and the lower one to u. As long as such an element is nonzero, we can prescribe the corresponding variable. Therefore, at both boundaries we can prescribe A as long as $w_1 \neq w_2$ and we can always prescribe u. We have by the definition of the basic wave speed c that $c = \sqrt{\frac{\beta}{2\rho}} A^{1/4}$ and under physiological conditions it holds that $A > 0,\ \beta > 0,\ \rho > 0$ such that $c > 0$. From equation (30) we have that $w_1 = u + 4c$ and $w_2 = u - 4c$, so the requirement $w_1 \neq w_2 \implies u + 4c \neq u - 4c \implies c \neq -c \implies c \neq 0$ which is what we have shown earlier. Thus we can always prescribe either A or u and this holds for both boundaries.

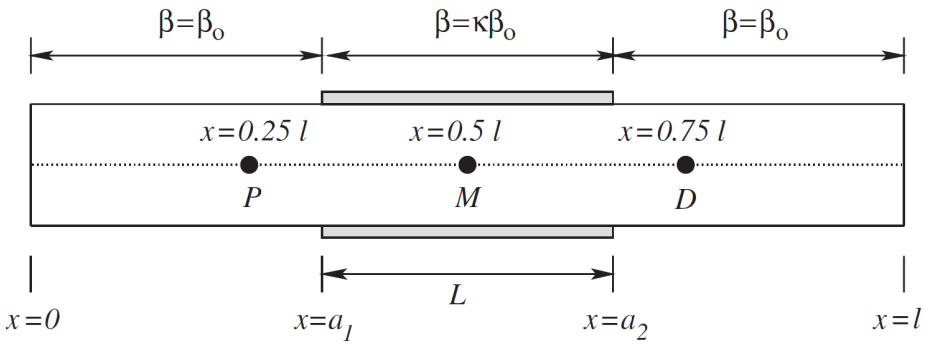

Figure 5: Schematic overview of a blood vessel with a stent from Sherwin et al. [1]

6.3.1 Boundary conditions - forward prescription

The standard method of applying boundary conditions is to choose the inflow characteristic variable such that a specific boundary condition is met. In our case that boundary condition is a prescribed pressure p, area A or velocity u. This is mathematically correct and leads to a well posed mathematical problem. But in the artery system, the heart is the source and we therefore expect a measured quantity to be propagated away from the heart. Hence this information is propagated with the characteristic variable, w_1. But to prescribe an actual quantity at the boundary, information of the *reflected* backwards propagating wave, w_2, is needed as well. Since the combination of forwards, w_1, and backwards, w_2, propagating wave uniquely describes the system. *However* the reflected backwards propagating waves are highly dependent on the system at hand. Are there discontinuous material properties such as a stent? Are there bifurcations? Where are they? When are they reached by the pressure pulse form the heart? All this information should be known quite precisely to be able to achieve the same reflected waves as in the measurements. If the reflected waves are not quite similar to the ones hidden in the measured quantity, then the inflowing wave is altered to be able to achieve the measured quantity at the boundary. Which, in turn changes the reflected waves and, as such, changes the inflowing wave even more and can thus lead to very different results.

It therefore seems more intuitive to prescribe the measured quantity through the forward propagating wave w_1 only, while assuming that the backwards propagating wave w_2 is fixed at its initial state. This is the so called *forward* prescription. Thus the *forward* prescription makes sure that the sensitive reflections do not make the inflowing wave sensitive and so a more robust solution is obtained. While *forward* prescription does not assure that at the boundary the given value is attained exactly, the difference between the real prescribed value and the *forward* prescription is at most a few percent, while enjoying good non-reflective properties, see [2, 5].

6.4 Test case - Stented artery

In order to test the algorithm described above, a test case from Sherwin et al. [1] is considered. In this test case a stent, an expandable metal mesh, is inserted into an artery. The artery has a constant cross-sectional area at rest and the blood is initially at rest. The stent is modelled as an increase in the stiffness of the blood vessel, through $\beta(x)$. See Figure 5 for a schematic overview of the test case and Figure 6 for the increase in stiffness, both from Sherwin et al. [1]. The test case is illustrated in Figure 5. The blood vessel has length l, the stent is inserted between $x = a_1$ and $x = a_2$ with length L and the increase in stiffness is κ times stiffer than the normal artery, β_0. The pressure will be measured at three points, the proximal P at $l/4$, the medial M at $l/2$ and lastly the distal D at $3l/4$. The jumps are made continuous with a sinusoid with width 2δ centred around the discontinuity, see Figure 6. See Table 1 for the numeric values used .

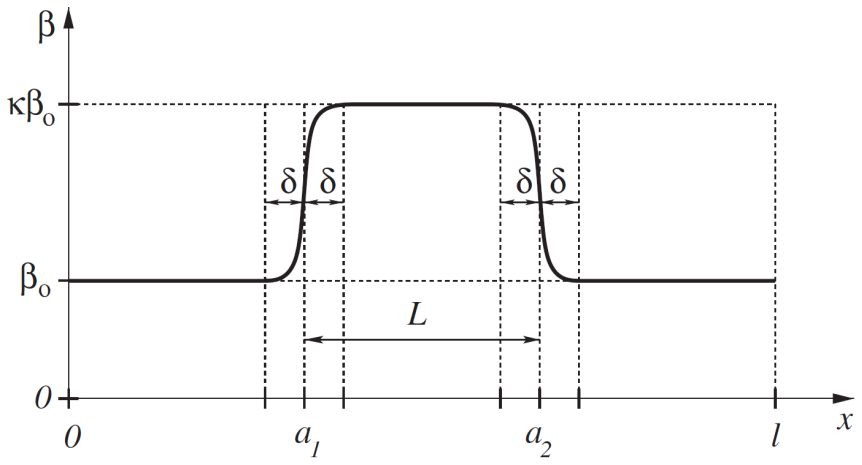

Figure 6: Smoothing of β with two sinusoids with width 2δ from Sherwin et al. [1]

Table 1: Numeric values used in the stented and normal artery test cases

$A_0(\text{cm}^2)$	$l(\text{cm})$	$a_1(\text{cm})$	$a_2(\text{cm})$	$L(\text{cm})$	$\rho(\text{g/cm}^3)$	$\delta(\text{cm})$	$\beta_0(\text{dyne/cm}^2)$	$\kappa(-)$	$T(\text{s})$
0.5	15	5	10	5	1	0.5	451352	100	0.33

6.4.1 Boundary conditions of the stented artery

The left boundary condition is a modelled pressure pulse from the heart. It is given as

$$\bar{p}(t) = 20000 \sin\left(\frac{2\pi t}{T}\right) \mathrm{H}\left(\frac{T}{2} - t\right) \tag{80}$$

Here, $\mathrm{H}(t)$ is the Heaviside step function, scaled and translated such that it allows half a period of the sine to pass. See the upper left graph of Figure 7 for this boundary condition's general shape. Using equation (17) and that $p_{ext} = 0$, we get at $x = 0$

$$\bar{p} = p_{ext} + \beta(\sqrt{A} - \sqrt{A_0}) \implies A = \left(\frac{\bar{p}}{\beta} + \sqrt{A_0}\right)^2 \tag{81}$$

We can express A in terms of w_1 and w_2 with (31), so solving for w_1 yields

$$A = \left(\frac{w_1 - w_2}{4}\right)^4 \left(\frac{\rho}{2\beta}\right)^2 \implies w_1 = w_2 + 4\sqrt{\frac{2\beta}{\rho}} A^{1/4} \tag{82}$$

Therefore we have at $x = 0$

$$w_1(0,t) = w_2(0,0) + \frac{4\sqrt{2}}{\sqrt{\rho}} \sqrt{\bar{p}(t) + \beta_0 \sqrt{A_0}} \tag{83}$$

As said in the *forward* boundary condition section, at the left boundary w_2 will be fixed at its initial value, since it is highly dependent on the model used and hence it is not known a priori. $w_2(0,0)$ will be calculated using equation (30) with the initial values of U from the first physical cell, so U_1^0. The above value for w_1 and the fixed value for w_2 are used to compute A and u using equation (31) for the inflow boundary's ghost cell.

For the right boundary a non reflective boundary condition is chosen as

$$w_2(l,t) = w_2(l,0) = \text{constant} \tag{84}$$

So, at the right boundary, w_2 is also fixed at its initial state, but this time it is because the chosen initial state is equivalent to the equilibrium state of the system. The right boundary's ghost cell is, therefore, a copy of the last physical cell.

6.4.2 Results of the stented artery

In this section, we compare the results obtained using the high resolution flux difference splitting method to the Discontinuous Galerkin method used in [1], on a normal artery and a stented artery. As a summary, the conservative form PDEs given by equation (59) are used and the initial conditions and boundary conditions are

$$\text{IC: } U(x,0) = \begin{bmatrix} A(x,0) \\ u(x,0) \end{bmatrix} = \begin{bmatrix} 0.5 \\ 0 \end{bmatrix}, \quad \text{BC}_1\text{: } \bar{p}(t) = 20000 \sin\left(\frac{2\pi t}{T}\right) \mathrm{H}\left(\frac{T}{2} - t\right), \quad \text{BC}_2\text{: } w_2(l,t) = w_2(l,0) \tag{85}$$

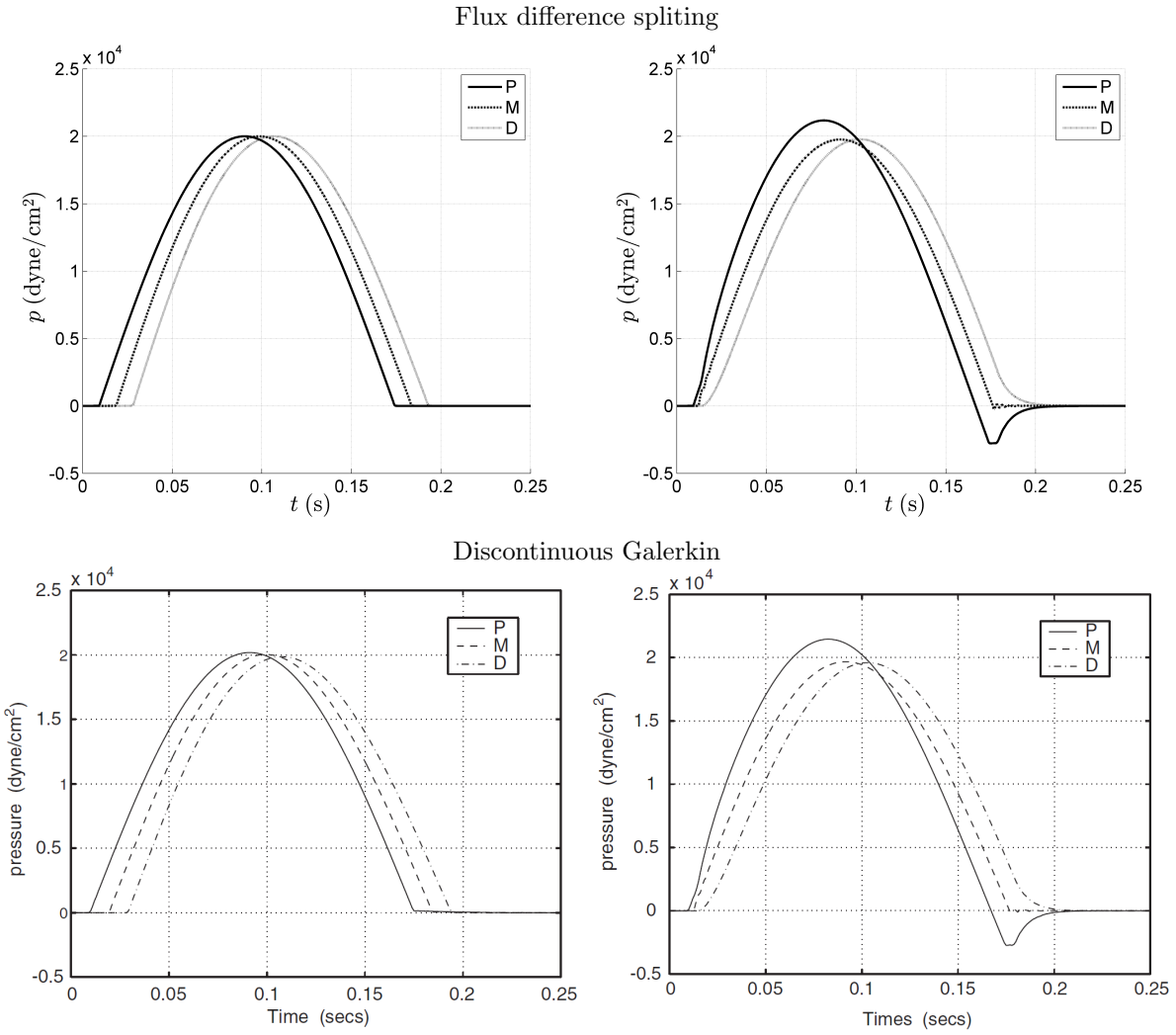

Figure 7: Result of a normal, left column, and stented, right column, blood vessel obtained using the the high resolution flux difference splitting method and the Discontinuous Galerkin method from Sherwin et al. [1]

In the left column of Figure 7 can be seen that we first applied the flux difference splitting method to an artery with no stent. The parameters are the same as in the stented blood vessel, see Table 1, but with the stiffness kept constant $\beta(x) = \beta_0$. Hence the material properties are constant. The solution shows a non distorted half sine wave propagating through the artery.

This is expected when looking at equation (38), since U is constant at $t = 0$ and the material properties are constant which imply that w_1 and w_2 are constant at $t = 0$. Using that the right boundary condition also prescribes the same constant value for w_2 for all t, it follows that w_2 is constant for all t and x. Note that indeed, PDE (38) allows for a trivial constant solution for w_2. Since w_2 is constant, it follows that the wave speed $\lambda_1 = \frac{5w_1+3w_2}{8}$, see below equation (38), does not depend on w_2.

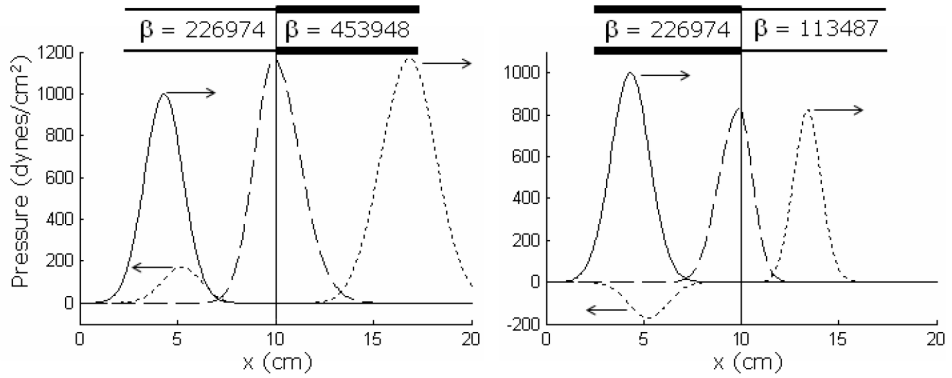

Figure 8: Figure from Mynard [5] to show the effect of a step increase and decrease of β on the wave propagation

Hence the coupled quasilinear system of PDEs for w_1 and w_2 decouples into two quasilinear PDEs for w_1 and w_2 respectively. Where the quasilinear PDE for w_1 is of interest since w_1 has a non constant boundary condition and hence w_1 is a nontrivial solution of the quasilinear PDE. However, we do not see the quasilinear behaviour in the solution because the maximum disturbance of w_1 is at most 1% and therefore is the difference in wave speed at most 1% and because the wave speed is large in comparison with the vessel length, $\frac{\lambda_1}{l} \approx 100$. So the wave has already passed the domain of interest before anything quasilinear occurs.

The pulse duration at P is about 0.16 seconds, which is in agreement with the duration of the prescribed pressure pulse at the boundary of $\frac{T}{2} = 0.165$ seconds.

It is important to note that the high resolution wave propagation algorithm (flux difference splitting method) does not 'smear out' the solution as the wave propagation algorithms did in the linear test case, see Figure 3, since the peak pulse height remains nearly constant throughout the vessel, as can be seen in Figure 7.

If a stent is inserted, then we get the right column of Figure 7. The stent is stiff with respect to the artery wall and hence it does not expand as easily. When the pressure pulse reaches the stent, the characteristic velocity of the blood λ_1 increases as a result of the increased stiffness and therefore points M and D are reached earlier than in the non-stented vessel. Furthermore, part of the pressure pulse is reflected due to the inlet of the stent and the reflected pulse is superimposed on the incoming pressure pulse leading to an increase in the pressure at P at $t = 0.8s$, as well as a dip in the pressure below the equilibrium pressure at $t = 0.18s$ due to the reflection at te outlet of the stent.

The mathematical way of looking at the right column of graphs is as follows. In the regions where the material properties are constant in x, the system is governed by equation (38) where there are no source terms and as said earlier we do not see quasilinear behaviour with the current conditions. Hence the quasilinear PDEs behave like scalar transport PDEs. The material properties are constant in the following three regions: before the stent, in the stent and after the stent. So here we expect non distorted transport as seen in the non stented artery. However at the boundaries of the *stent* the material properties are not constant and we expect the sources terms to influence the solution. At the inlet of the stent β increases, so $\frac{d\beta}{dx} > 0$, and this eventually results in a positive source term for A in the left running waves and subsequently p. While at the outlet of the stent β decreases which leads to a negative source term for A for the left running waves and thus p, as was shown by Mynard [5] with a step increase and decrease in β, see Figure 8.

The method based on flux differencing shows the same characteristics as the discontinuous Galerkin method for both the normal and stented blood vessel. Some of the characteristics are the non distorted wave propagation for the normal artery with comparable speeds, the increase in peak pressure at P just before the stent as well as the dip in pressure at P and the small (numerical) oscillations at M at $t = 0.18$ s. But the flux differencing method has a slightly lower peak pressure at P at $t = 0.08$.

A grid of 400 cells is used and the time step is chosen such that the CFL condition is just satisfied at each time t^n, so

$$\Delta t^n = \frac{\Delta x}{|\lambda^n|_{\max}} \tag{86}$$

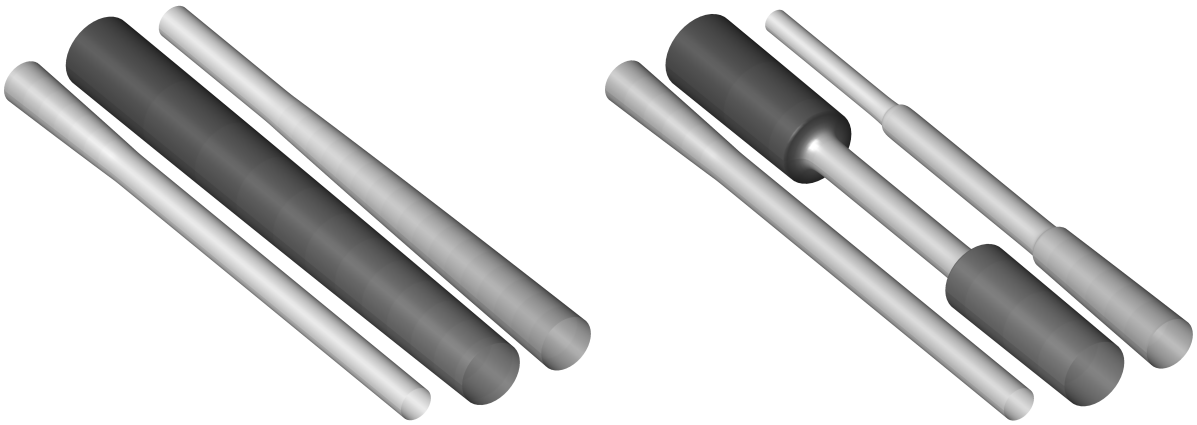

Figure 9: 3D visualisation of the normal, left, and stented, right, artery cross-sectional area A at $t = 0.01s, 0.075s, 0.175s$ from left to right respectively. The displacement from the equilibrium cross-sectional area A_0 is magnified with a factor of 30.

In Figure 9 a three-dimensional representation of the cross-sectional area A is given at times $t = 0.01$ s, $t = 0.075$ s, $t = 0.175$ s for the normal and stented artery. The displacement from the equilibrium cross-sectional area A_0 is magnified with a factor of 30 for illustration purposes.

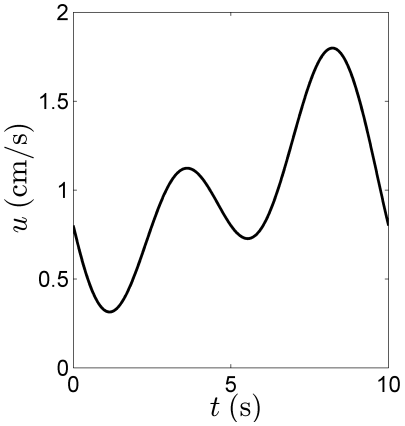

Figure 10: The inflow boundary condition $\bar{u}(t)$ is shown

6.5 Test case - Tapered artery

As a further test case from Sherwin et at [1], a normal and a tapered blood vessel are considered. A tapered artery is a blood vessel that along the length of the vessel gradually becomes smaller in cross-sectional area and hence the blood vessel is shaped as a cone. At the inlet of the artery $A_0 = 1$ cm^2 and the cross-sectional area changes linearly to the outlet cross-sectional area $A_0 = 0.5$ cm^2. The initial condition for the velocity u is $u = 1$ cm/s at the inlet and then varies linearly to $u = 2$ cm/s at the outlet such that the mass flux at the inlet and outlet are the same. While for the normal blood vessel $A_0 = 1$ cm^2 all the way through the blood vessel and the initial condition for the velocity is $u(x, 0) = 1$ cm/s such that the mass flux is comparable with that of the tapered artery. According to Sherwin et al. [1] is the basic wave speed an order of magnitude higher than the mean velocity of the blood. Thus looking at the normal artery, we want to choose β and ρ such that $c_0 = \sqrt{\frac{\beta}{2\rho}} A_0^{1/4} = 10$ cm/s. We therefore choose $\beta = 100$ dyne/cm^3 and $\rho = 0.5$ g/cm^3. Furthermore, they assumed that the spatial (linear) wavelength λ_0 (not the characteristic velocity) of the waves within this artery 100 times larger is than the vessel diameter, so $\lambda_0 = 100\sqrt{\frac{4}{\pi} A_0} \approx 100$ cm. In the linear case, we then have $\lambda_0 = c_0 T$, so $T = \frac{\lambda_0}{c_0} = 10$ s where T is the period of the wave. To be able to see the waves with a spatial wavelength of 100 cm a blood vessel of length $L = 200$ cm is considered where the inlet is located at $x = -100$ and the outlet at $x = 100$. We note that this problem is not physically accurate since there are no arteries that are 2 meters long without a bifurcation.

6.5.1 Boundary conditions of the tapered artery

The inflow boundary condition is

$$u(-100, t) = \bar{u}(t) = 1 - 0.4\sin(\omega t) - 0.4\sin(2\omega t) - 0.2\cos(2\omega t) \tag{87}$$

Here, $\omega = \frac{2\pi}{T}$ is the angular frequency with T the time period derived to be 10 s. The waveform is shown in Figure 10. With this left boundary condition we have taken into account, that further down the arterial tree the modelled half sine wave at the aortic valve given by equation (80) might have changed considerably due to reflections at the bifurcations. Using equation (31), we have for the inflowing characteristic that

$$u = \frac{w_1 + w_2}{2} \implies w_1(-100, t) = 2\bar{u}(t) - w_2(-100, 0) \tag{88}$$

As said in the *forward* boundary condition section, at the left boundary w_2 will be fixed at its initial value, since it's highly depended on the model used and hence it is not known a priori. Using the above values for w_1 and the initial value for w_2, we compute A and u for the inflow boundary's ghost cell using

equation (31) as

$$
\begin{aligned}
A(-100,t) &= \left(\frac{w_1(-100,t)-w_2(-100,0)}{4}\right)^4\left(\frac{\rho}{2\beta}\right)^2 = \left(\frac{2\bar{u}(t)-2w_2(-100,0)}{4}\right)^4\left(\frac{\rho}{2\beta}\right)^2 \\
&= \left(\frac{\bar{u}(t)-\left(u_1^0-4\sqrt{\frac{\beta}{2\rho}}(A_1^0)^{1/4}\right)}{2}\right)^4\left(\frac{\rho}{2\beta}\right)^2 \approx A_1^0 \\
u(-100,t) &= \bar{u}(t)
\end{aligned} \tag{89}
$$

where A_1^0 and u_1^0 are the values of A and u in the first physical cell at time $t=0$, and where we no longer assume that *forward* prescription holds for u, such that $u_0^1 \approx \bar{u}(t)$. A similar assumption was made by Sherwin et al. [1], who took for the ghost cell $A(-100,t) = 1$ cm^2 and $u(-100,t) = \bar{u}(t)$.

The outflow boundary condition is a non reflective boundary condition and is given by

$$w_2(100,t) = w_2(100,0) = \text{constant} \tag{90}$$

So, at the right boundary, w_2 is also fixed at its initial state, but this time it is because the chosen initial state is equivalent to the equilibrium state of the system. The right boundary's ghost cell is therefore a copy of the last physical cell.

6.5.2 Results of the non tapered artery

First the non tapered artery, the normal artery, is considered. The results obtained using the high resolution flux difference splitting will be compared with the Discontinuous Galerkin scheme used in [1].

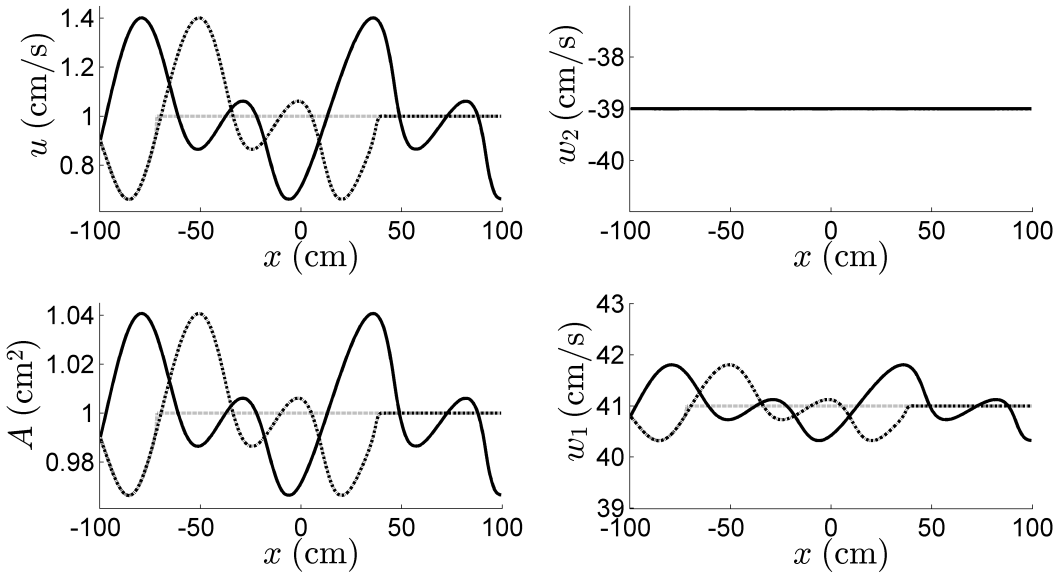

Figure 11: Results of the normal artery obtained with the high resolution flux differencing scheme at $t = 2.5s$ (dotted line), at $t = 12.5s$ (dashed line) and at $t = 20s$ (continuous line)

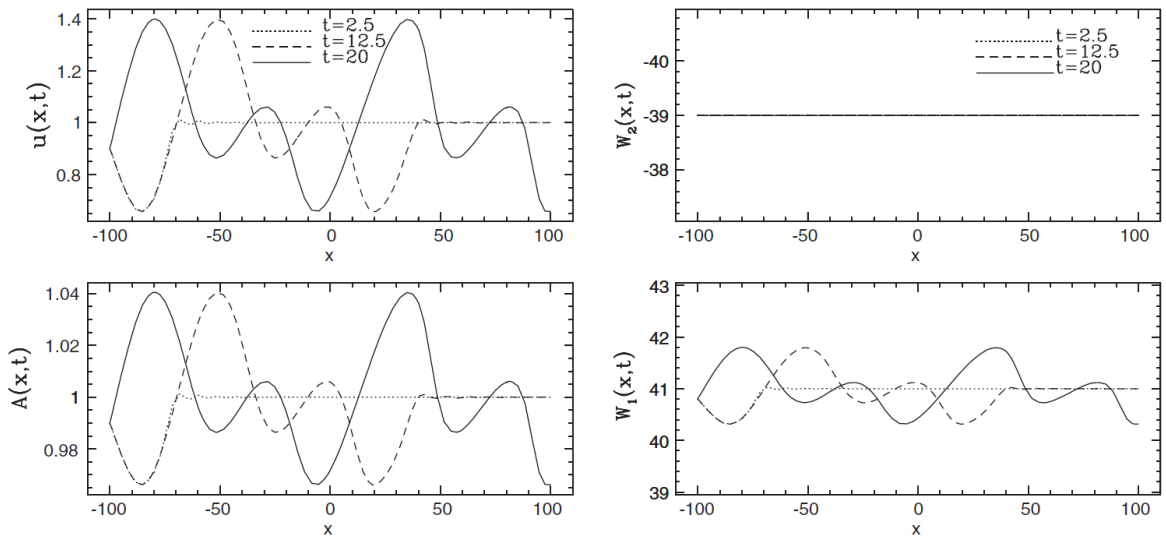

Figure 12: Results of the normal artery obtained with the Discontinuous Galerkin scheme at $t = 2.5s$ (dotted line), at $t = 12.5s$ (dashed line) and at $t = 20s$ (continuous line) from Sherwin et al. [1]

Note that in Figure 11, in contrast to the stented artery, where the temporal wave was shown, the spatial wave is shown here. It is clearly visible that the prescribed blood velocity from equation (87) is propagated throughout the normal artery, albeit a bit lower than the prescribed value due to the approximation made at the inflow boundary. The normal propagation is expected since the material properties are constant.

Because we are considering a long blood vessel, the non linearities of the system can also be seen in the form of the steepening of the pulse. This effect can be seen for the $t = 20$ s continuous line, compare for example the steepness of the pulse at $x = 20$ cm with the steepness of the pulse at $x = -90$ cm.

Furthermore, for the continuous line ($t = 20$ s) for either A or u, we expect in the linear case to see $\frac{L}{\lambda_0} = 2$ pulses in the spatial domain (λ_0 is the linear wavelength), but since the actual forward travelling wave speed is $\lambda_1 = u + c$, it follows that $\lambda_a = \lambda_1 T > \lambda_0 T$, since $u > 0$ for all t. So the actual wavelength λ_a is larger than in the linear case and, hence, less than 2 wavelengths are shown in Figure 11.

Moreover, since w_2 is constant, it follows with equation (31) that u and A in Figure 11 are of similar shape.

The results from the flux differencing and the Discontinuous Galerkin are exactly the same, as shown in Figure 11 and Figure 12 respectively. The only small differences that can be noticed by the naked eye are the small oscillation that occur for the Discontinuous Galerkin when the pulse first disturbs the initial state, which can be seen for the dotted line at $x = -60$ cm and for the dashed line at $x = 50$ cm.

6.5.3 Results of the tapered artery

In this section the results of the tapered artery acquired with the high resolution flux differencing method are compared with the Discontinues Galerkin method from [1].

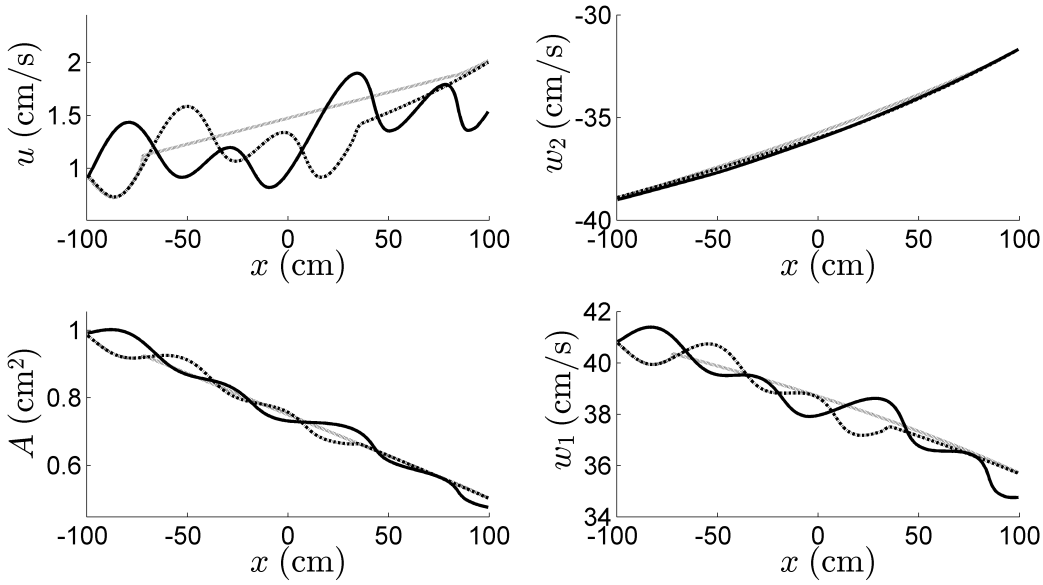

Figure 13: Results of the tapered artery obtained with the high resolution flux difference splitting method at $t = 2.5s$ (dotted line), at $t = 12.5s$ (dashed line) and at $t = 20s$ (continuous line)

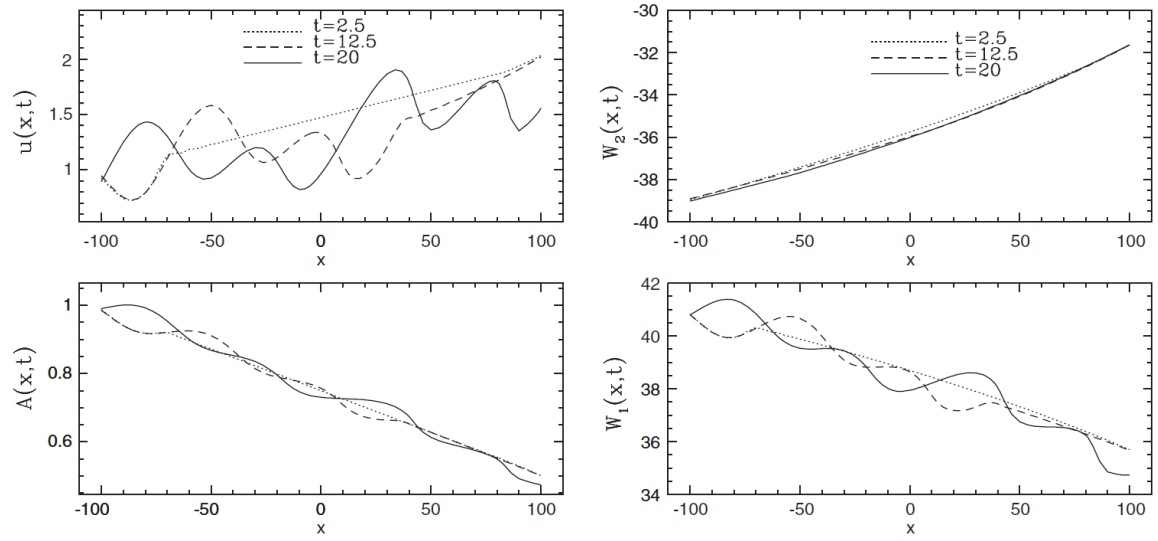

Figure 14: Results of the tapered artery obtained with the Discontinuous Galerkin scheme at $t = 2.5s$ (dotted line), at $t = 12.5s$ (dashed line) and at $t = 20s$ (continuous line) from Sherwin et al. [1]

We note that in Figure 13 for the continuous line ($t = 20$ s) of either A or u, a tiny bit more than two wavelengths can be seen compared to the slightly less than 2 wavelengths in Figure 11. This is to be expected, since the basic wave speed $c_0 = \sqrt{\frac{\beta}{2\rho}} A_0^{1/4}$ decreases over the length of the blood vessel, because the blood vessel is tapered. Furthermore, w_2 is almost constant in time and the waves are getting a bit steeper further down the artery with respect to the normal artery.

The two methods again show exactly the same solution and the oscillation of the DG are less visible.

A grid size of 400 is used for both the non tapered and the tapered blood vessel, and the time step is chosen such that the CFL condition is just satisfied at each instant in time.

In Figure 15 a three-dimensional representation of A is given at $t = 2.5s, 12.5s, 20s$. The displacement from the equilibrium cross-sectional area area A_0 is magnified by a factor of 10 and the x-axis is scaled

with a factor of 1/20.

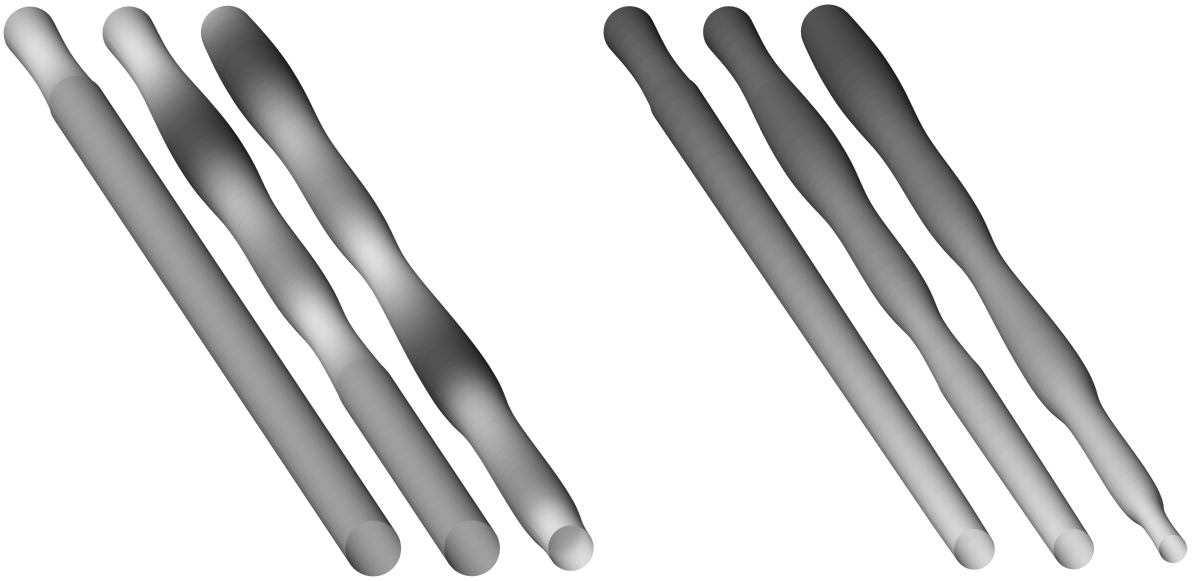

Figure 15: 3D visualisation of the normal, left, and tapered, right, artery cross-sectional area A at $t = 2.5s, 12.5s, 20s$ from left to right respectively. The displacement from the equilibrium cross-sectional area A_0 is magnified with a factor of 10

7 The human arterial tree

In this section we will be modelling the human arterial tree, but before we can do that we first have to look at a bifurcation or the branching of an artery.

7.1 Treatment of a bifurcation

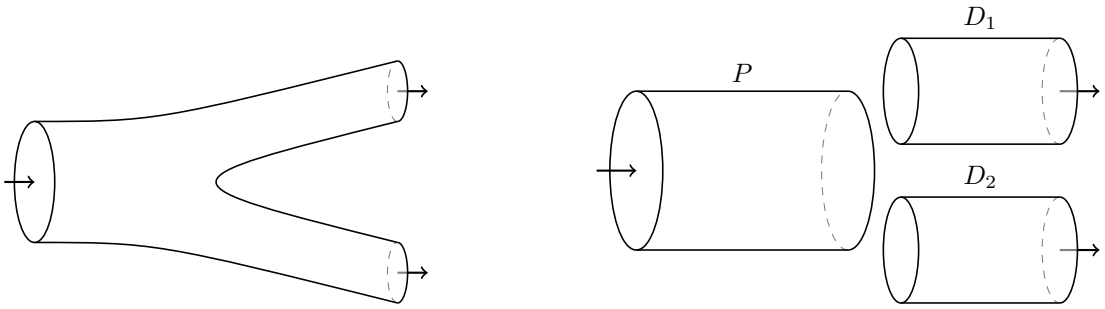

Figure 16: Schematic overview of a bifurcation

See Figure 16 for a schematic overview of a bifurcation. The blood vessel that branches is called the parent vessel P and the vessels that the parent branches into are called the daughter vessels D_1 and D_2. Since we have a 2x2 system of PDEs, we only have two waves propagating in the solution. We can therefore not add an extra f-wave to the flux differencing scheme since we would obtain 3 unknowns for 2 equations. We could use other conditions to complete the system but that would cause us to solve a nonlinear system of equations, see for example Sherwin et al. [1]. Here, we have chosen for another approach. We model the two daughter vessels as one big blood vessel and the resulting f-wave is distributed according to the cross-sectional areas at rest. To model the two daughter vessels as one big artery D we will use the following assumptions:

The cross-sectional area of D is given by the sum of the daughter vessel's cross-sectional areas

$$A_D^n = A_{D_1}^n + A_{D_2}^n \tag{91}$$

The equilibrium cross-sectional area of D is given by the sum of the equilibrium cross-sectional areas of the daughter vessels

$$A_{0,D}^n = A_{0,D_1}^n + A_{0,D_2}^n \tag{92}$$

The resulting f-wave $\mathcal{Z}_D^n$ is distributed according to

$$\mathcal{Z}_{D_1}^n = \alpha \mathcal{Z}_D^n, \quad \mathcal{Z}_{D_2}^n = \gamma \mathcal{Z}_D^n \tag{93}$$

Where

$$\alpha = \frac{A_{0,D_1}}{A_{0,D_1} + A_{0,D_2}}, \quad \gamma = \frac{A_{0,D_2}}{A_{0,D_1} + A_{0,D_2}} \tag{94}$$

are the relative cross-sectional area fractions at rest.

The mean velocity of the blood in D is given by the special averaging

$$u_D^n = \alpha u_{D_1}^n + \gamma u_{D_2}^n \tag{95}$$

The stiffness β of D is also given by the special averaging

$$\beta_D = \alpha \beta_{D_1} + \gamma \beta_{D_2} \tag{96}$$

Lastly the basic wave speed c at D is obtained using the new base units and is given by

$$c_D^n = \sqrt{\frac{\beta_D}{2\rho}} {A_D^n}^{1/4} \tag{97}$$

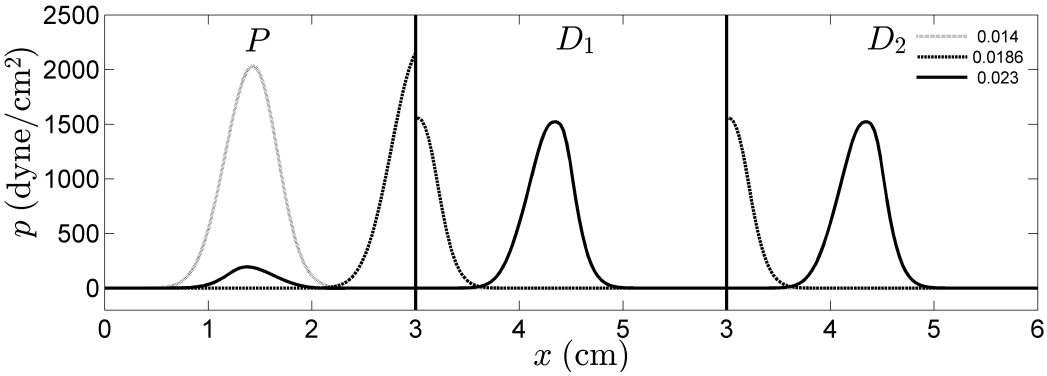

Figure 17: Result of the equal A_0 and equal β for the daughter vessels test case, shown for $t = 0.014$ s, $t = 0.0186$ s and $t = 0.023$ s

Using the above assumptions for artery D, we get the following flux differencing at interface $i - 1/2$

$$F_D(U_D^n) - F_P(U_P^n) = \mathcal{Z}_{1,D}^n + \mathcal{Z}_{2,P}^n = \beta_{1,i-1/2}^n r_{1,D}^n + \beta_{2,i-1/2}^n r_{2,P}^n = R_{i-1/2}^n \beta_{i-1/2}^n \tag{98}$$

$$= \frac{1}{2} \begin{bmatrix} \frac{A_D^n}{c_D^n} & -\frac{A_P^n}{c_P^n} \\ 1 & 1 \end{bmatrix} \begin{bmatrix} \beta_{1,i-1/2}^n \\ \beta_{2,i-1/2}^n \end{bmatrix} \tag{99}$$

Solving for $\beta_{i-1/2}^n$ gives

$$\begin{bmatrix} \beta_{1,i-1/2}^n \\ \beta_{2,i-1/2}^n \end{bmatrix} = \frac{2}{\frac{A_D^n}{c_D^n} + \frac{A_P^n}{c_P^n}} \begin{bmatrix} 1 & \frac{A_P^n}{c_P^n} \\ -1 & \frac{A_D^n}{c_D^n} \end{bmatrix} (F_D(U_D^n) - F_P(U_P^n)) \tag{100}$$

The f-wave can then be found from its definition $\mathcal{Z}_D^n = \beta_{1,i-1/2}^n r_D^n$ and equation (93) can be used for the f-waves in the two daughter vessels. We use the linear wave propagation algorithm (71) (flux difference splitting algorithm) at the bifurcations and for the rest of the blood vessels we use the high resolution flux differencing algorithm given by equation (72).

7.2 Test case - Bifurcation

We apply the high resolution flux differencing scheme with the above treatment of the bifurcations to three bifurcation test cases to investigate its behaviour and to study the reflected waves caused by the current treatment of the bifurcations. The test cases are called respectively 'equal A_0 and equal β', 'equal A_0 and different β' and 'different A_0 and equal β' for the daughter vessels.

To model a bifurcation of the ascending aorta, the parent equilibrium cross-sectional area $A_{0,P}$ is taken to be 6 cm^2, see Table 4 in appendix B for the physiological data from Sherwin et al. [1]. The stiffness of the parent vessel β_p has an estimated value of 10^5 dyne/cm^3 and the length of each vessel is 3 cm. The blood is initially at rest and the initial cross-sectional area is the equilibrium cross-sectional area in each vessel. The left boundary condition is a quick and small pulse given as

$$\bar{A}(t) = A_{0,P} + 0.1e^{-10^6(t-0.1)^2} \tag{101}$$

to minimize the effects of the non-linearities in the solution. The outflow boundary conditions are of the non reflective type.

7.2.1 Results of the equal A_0 and equal β for the daughter vessels

As was illustrated in Figure 16, we normally expect that the equilibrium cross-sectional area just before the bifurcation $A_{0,P}$ to be comparable to the sum of the equilibrium cross-sectional areas just after the

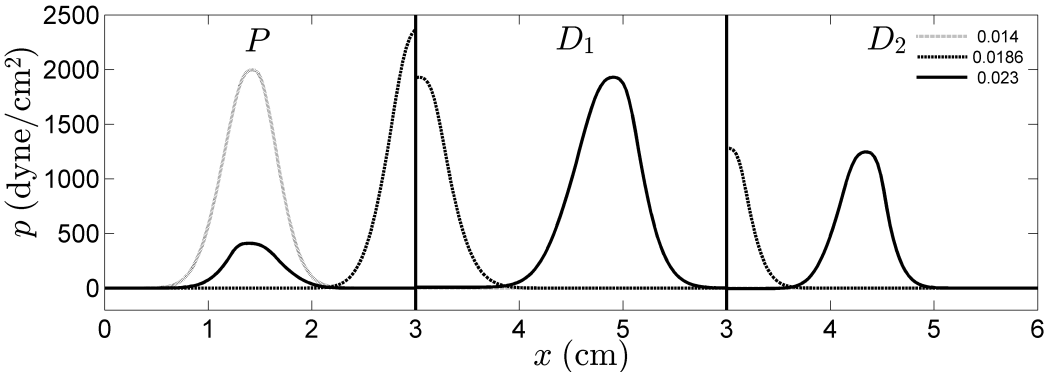

Figure 18: Result of the equal A_0 and different β for the daughter vessels test case, shown for $t = 0.014$ s, $t = 0.0186$ s and $t = 0.023$ s. Here is $\beta_{D_1} = 2\beta_P = 2\beta_{D_2}$

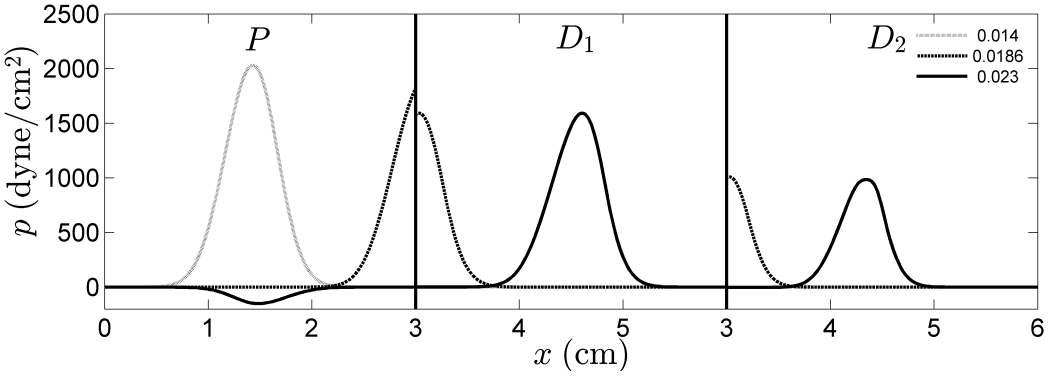

Figure 19: Result of the different A_0 and equal β for the daughter vessels test case, shown for $t = 0.014$ s, $t = 0.0186$ s and $t = 0.023$ s. Here is $A_{0,P} = 6$ cm^2, $A_{0,D_1} = 6$ cm^2 and $A_{0,D_2} = 3$ cm^2

bifurcation $A_{0,D_1} + A_{0,D_2}$. Therefore, was chosen for $A_{0,D_1} = A_{0,D_2} = \frac{1}{2}A_{0,P} = 3$ cm^2 in this test case. The arterial stiffness β is constant and the same in each vessel to minimize the reflections, so $\beta_{D_1} = \beta_{D_2} = \beta_P$.

The results of the equal test case are found in Figure 17. At $t = 0.014$ s, we can see that the pulse is travelling through the parent vessel. The bifurcation from parent P to daughters D_1 and D_2 is illustrated by the black lines located at $x = 3$ cm and $x = 6$ cm. The pulse hits the bifurcation at $t = 0.0186$ s, where the peak of the pulse in the parent vessel increases slightly and the wave is evenly split between the two daughter vessels. At $t = 0.023$ s, a small pulse is reflected back into the parent vessel and the pulse propagates undistorted in the daughter vessels. We expected no reflected pulse with these conditions since the material properties and the cross-sectional areas are equal before and after the bifurcation. The reflected pulse is about 10% the height of the original pressure pulse, which is acceptable.

Note that the pressures in the daughter vessels do not add up to the pressure in the parent vessel, this is due to that the pressure is given by $p = p_{ext} + \beta(\sqrt{A} - \sqrt{A_0})$, so it is dependent on $A_0(x)$. We will do a small calculation to show this effect. A small amplitude pulse W given above steady equilibrium, is of the form $A = A_0 + W$. Therefore

$$p = \beta(\sqrt{A_0 + W} - \sqrt{A_0}) = \beta\left(\sqrt{A_0}\sqrt{1 + \frac{W}{A_0}} - \sqrt{A_0}\right) \approx \beta\left(\sqrt{A_0}\left(1 + \frac{W}{2A_0}\right) - \sqrt{A_0}\right) = \frac{\beta W}{2\sqrt{A_0}} \tag{102}$$

where we have used the approximate Taylor expansion $\sqrt{1+x} \approx 1 + \frac{x}{2}$ for small x. After substitution the values $\beta = 10^5$ dyne/cm^3, $W = 0.1$ cm^2, $A_0 = 6$ cm^2, we get $p_P = 2.04 \cdot 10^3$ dyne/cm^2 for the peak pressure in the parent vessel. Using that approximately half of the pulse above equilibrium reaches the daughter vessels, so $W = 0.05$ cm^2 and $A_0 = 3$ cm^2, we get $P_{D_i} = 1.44 \cdot 10^3$ dyne/cm^2 for the peak pressure in the daughter vessels. These calculations match with what we see in Figure 17. Thus we see that the pressure scales linearly with the pulse 'height', but does take the equilibrium cross-sectional area into account whenever the relative pulse 'height' is not to big $\frac{W}{A_0} \ll 1$.

7.2.2 Results of the equal A_0 and different β for the daughter vessels

Same conditions as test case 'equal A_0 and equal β' for the daughter vessels but this time with the stiffness of daughter 1 twice as high as in the previous test case, so $\beta_{D_1} = 2\beta_{D_2} = 2 \cdot 10^5$ dyne/cm^3.

The results can be found in Figure 18. The reflected pulse height is about 20% of the original pulse height. The transmitted wave in daughter vessel D_1 at $t = 0.0186$ s has travelled further than the transmitted pulse in daughter D_2 in the same amount of time, hence its velocity it higher. This is logical since the basic wave speed $c = \sqrt{\frac{\beta}{2\rho}}$ scales with β. The ratio of the pulses in the daughter vessels is $R = \frac{P_{D_1}}{P_{D_2}} = 1.55$ instead of $\frac{\beta_{D_1}}{\beta_{D_2}} = 2$, see equation (102), which is likely due to the fact that, indeed, D_1 and D_2 get a 1:1 split of the flux from the parent vessel, but they can independently propagate flux backwards into the parent vessel. Since the stiffness is relatively high in the D_1 vessel, it is likely that more flux is propagated backwards than in the lower stiffness vessel D_2 and hence the ratio R is lower. This behaviour is physically more realistic since we expect the stiffness to play a role in the flux distribution. Since the stiffer the artery, the less it wants to expand, hence, a lower proportion of the flux is propagated into the stiff artery.

7.2.3 Results of the different A_0 and equal β for the daughter vessels

Figure 19 shows the results if we keep all the conditions the same as in the 'equal A_0 and equal β' for the daughter vessels test case, but this time is the equilibrium cross-sectional area of daughter D_1 taken as $A_{0,D_1} = 6$ cm^2 and for daughter D_2 we have $A_{0,D_2} = 3$ cm^2.

The reflected pressure pulse is about 7.5% of the size of the inlet pressure pulse and it is negative, which is caused by the fact that the sum of the equilibrium cross-sectional areas is greater than that of the parent vessel and hence a bit more of the pulse is transmitted and therefore there is a negative pressure pulse or decrease in the cross-sectional area below the equilibrium cross-sectional area in the parent vessel. Also note that this time the flux is split according to $2:1$, therefore more flux is propagated into D_1 which in turn leads to a higher pressure than in D_2, such that more flux will be propagated backwards, which is visible in the transmitted pulses in the daughter vessels.

Table 2: Parent pointer of Figure 20 and Figure 21. The top row is the index of the node and the bottom row is the corresponding parent. Parent 0 means that that node is a root

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28
0	1	1	3	3	4	4	7	7	9	9	5	5	2	2	15	15	14	14	19	19	21	21	23	23	18	18	27

29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55
27	29	29	30	30	28	28	35	35	37	37	39	39	41	41	42	42	44	44	46	46	43	43	50	50	52	52

7.3 Human arterial tree

Having developed all the tools necessary we will proceed to model the human arterial tree. The arterial tree is modelled as a network of 55 blood vessels. Their spatial dependence is given by Figure 20 from [1]. The spatial dependence of the arteries in Figure 20 is isomorphic to a full binary tree, in which each node in the tree has 0 or 2 children. Each artery will be a node and the edges represents the spatial connections. The root of the tree is artery 1. Artery 1 branches into artery 2 and artery 3. Hence an edge will be drawn from node 1 to node 2 and from node 1 to node 3. Blood vessel 2 bifurcates to artery 14 and artery 15. Artery 3 splits into artery 4 and artery 5 and so forth. Using this technique we obtain the graph in Figure 21.

There is a convenient data structure to capture this branching tree behaviour, the parent pointer data structure. The parent pointer is an array or vector where each index corresponds to the respective node (artery) and the value at each index represents which node is its parent. See Table 2 for the parent pointer of Figure 20 and 21. Artery 1 is a root, so it has no parents and hence it has parent 0. Artery 2 and artery 3 are the children of artery 1, so artery 1 is their parent. The parent of arteries 14 and 15 is artery 2.

The physiological data for the 55 main arteries in the human arterial tree is provided by Sherwin et al. [1] and is shown in Table 4 and Table 5 in appendix B. This data is altered to minimize the reflections at the bifurcations for the forward running waves. Blood flows from the heart via the arteries to the capillaries where the oxygen transfer takes place. We are interested in the results in the larger arteries and not so much in the numerous small capillaries where the blood flow slows down significantly. But the network of blood vessel further down the arterial tree will generate backwards travelling waves. To incorporate these reflections in to the model of 55 arteries is chosen for a *terminal* resistance coefficient R_t at the *terminal* vessels, defined as

$$R_t = -\frac{\Delta w_2}{\Delta w_1} = -\frac{w_2^{n+1} - w_2^0}{w_1^{n+1} - w_1^0} \tag{103}$$

where w_j^0 is the j-th characteristic variable fixed at its initial value. The terminal resistance coefficient will be applied to each vessel end that is not connected to other blood vessels, the outflow blood vessels. The lowest row of nodes of Figure 21 represent the outflow blood vessels. It follows that w_2 has to be prescribed at the outflow blood vessels, we therefore solve for w_2 and this yields

$$w_2^{n+1} = w_2^0 - R_t(w_1^{n+1} - w_1^0) \tag{104}$$

We see that $R_t = 0$ represents a non reflective boundary condition such that the wave can leave without disturbing the domain of interest. Setting $R_t = 1$ represents a perfect blockage of the pulse and thus the pulse is fully send back into the computational domain. For the terminal resistance coefficient it holds that $R_t \in [0, 1]$.

The inflow boundary conditions has to be prescribed at the start of artery 1 and we will use the *forward* prescription. The inflow boundary condition is taken as

$$\bar{A}(t) = A_{0,1} + 1.587 \cdot \delta(t)\,\mathrm{H}(\delta(t)), \quad \delta(t) = \sin(\omega t + 0.628) - 0.588 \tag{105}$$

Where $A_{0,1} = 5.983$ cm^2 is the equilibrium cross-sectional area of the first artery, see Table 4, H(t) is the Heaviside step function, $\omega = \frac{2\pi}{T}$ rad/s is the angular frequency with $T = 1$ s. The inflow boundary condition is plotted in Figure 22. The density of blood was taken to be $\rho = 1.021$ g/cm^3 and the blood is initially at rest and the cross-sectional area is at the equilibrium value.

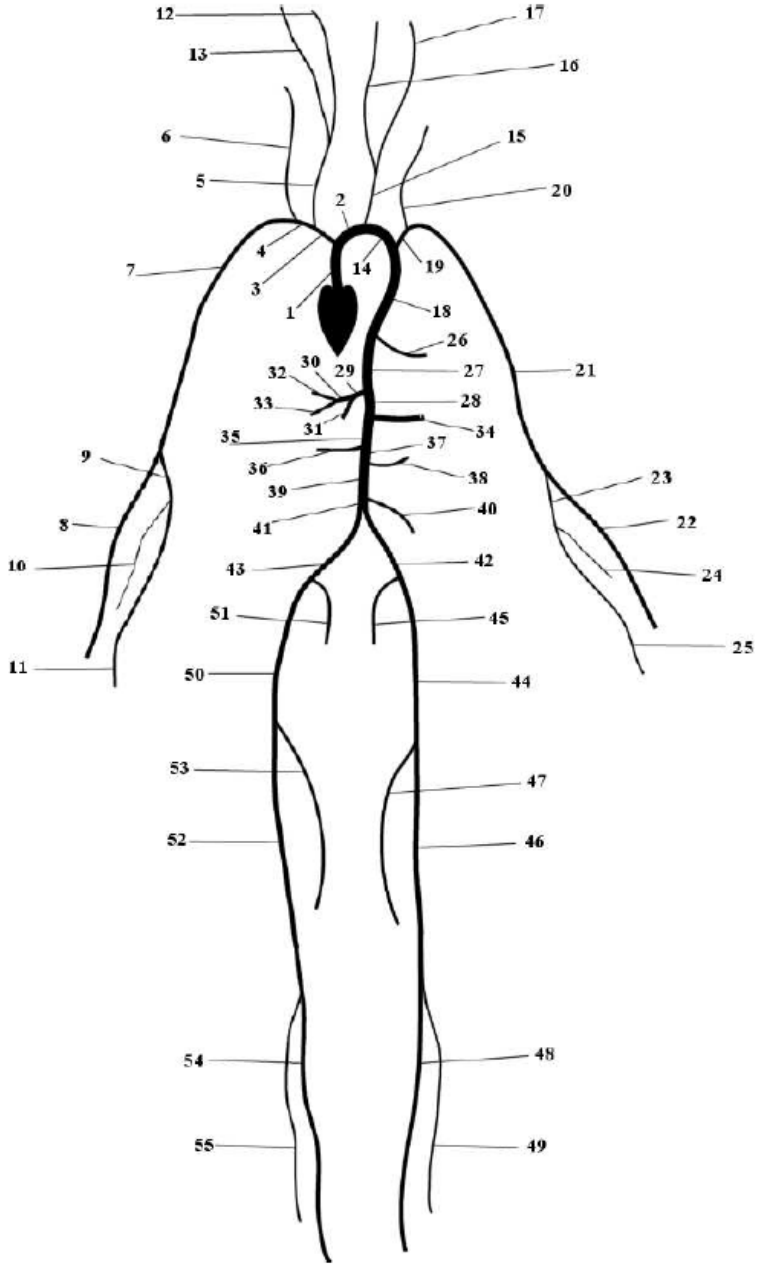

Figure 20: Connectivity of the 55 main arteries in the human arterial system from Sherwin et al. [1]

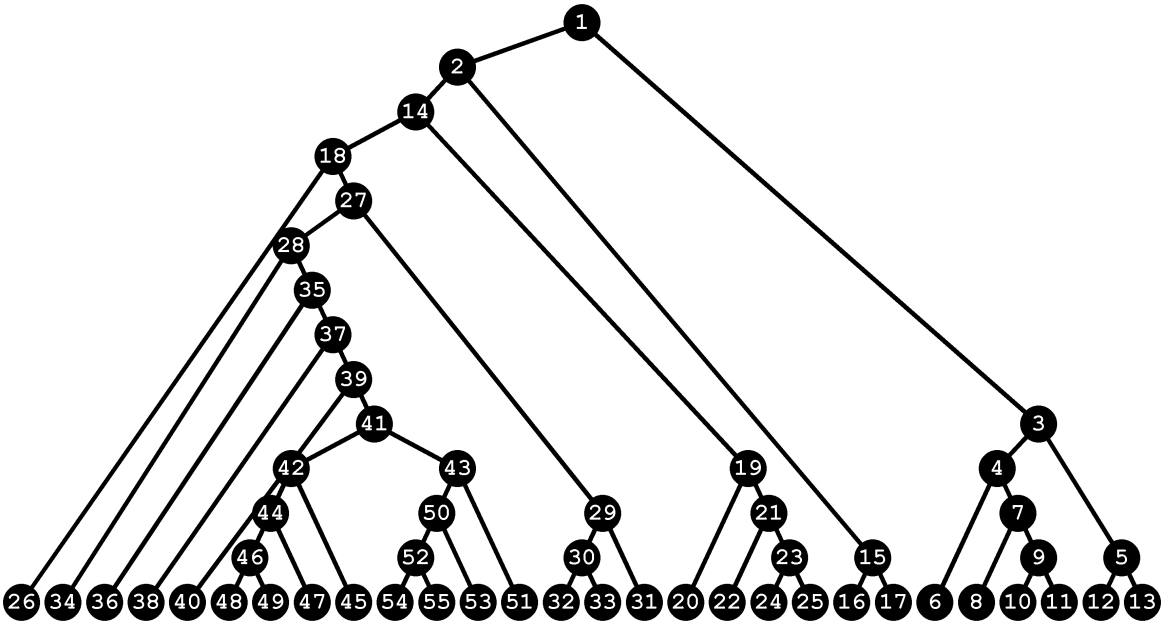

Figure 21: Graph of the connectivity of the 55 main arteries in the human arterial system

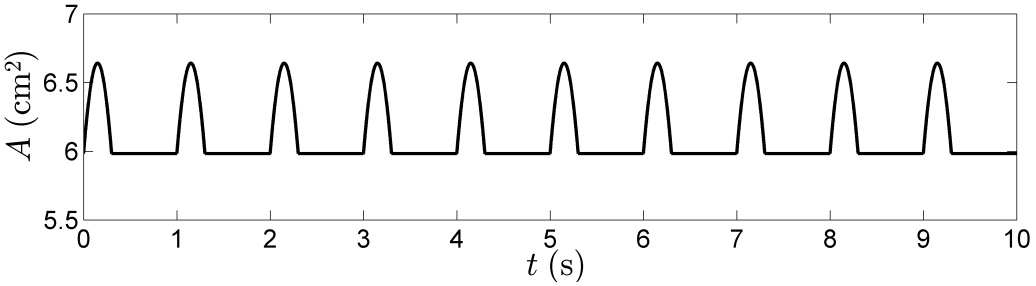

Figure 22: Plot of the inflow boundary condition A, see equation (105)

7.4 Results of the human arterial tree

The results of the simulations of the 55 main arteries in the human arterial tree will be given in this section.

Figures 23, 24, 25 and 25 show A, u, w_1 and w_2 for one period $T = 1$ s at the start of the ascending aorta (artery 1) and at the start of the l. anterior tibial (artery 49) with and without terminal resistance ($R_t = 0$). On the left, the results obtained using the high resolution flux differencing method are shown and on the right, the results of the Discontinuous Galerkin form Sherwin et al. [1] are shown. Each second, a pulse is sent into the arterial network. In order to let the wave patters develop in the arterial tree, we will wait for 9 pulses before looking at a single cycle.

The data of the network of 55 arteries is adjusted such that there should be no (or small) reflections at the bifurcations for forward running waves according to Sherwin et al. [1] . This, however, is not the case for both the flux differencing method and the Discontinuous Galerkin as can be seen in Figure 23, where the cross-sectional area A and mean velocity u do not immediately return to the equilibrium values when the incoming pulse has passed. This implies that there is a reflected wave w_2 originating from the bifurcations since there is no terminal resistance. In Figure 24, it can be seen that, indeed, no pulse is reflected at the terminal vessels, since w_2 is constant.

The general behaviour of the DG and FD is similar but the actual waveforms are quite different as displayed in Figure 23. The mean velocity u and cross-sectional area A also differ quite a lot in value between the two methods. They both show the shape of the prescribed pulse until $t = 9.3$ s. Although the FD method has lower mean velocity u and a greater cross-sectional area A. After the pulse has passed, so $t > 9.3$ s, u shows a steep decline followed by a steep increase, while for the DG the process of returning to equilibrium is a lot slower and smoother. This is likely caused by the different approaches of handling the bifurcations. It seems that the approach we used, where we saw the daughter vessels as one big vessel, causes a greater portion of the pulse to be reflected.

When taking the terminal resistance into account at the terminal vessels, we obtain Figures 25 and 26 at the inlet of the ascending aorta and at the start of l. anterior tibial respectively. The amount of waves changes significantly due to terminal resistance. Since, the forward running waves reflect of the terminal vessels and become backwards running waves that reflect of the bifurcation and, as such, create even more small waves.

In the left column of Figure 25, it can be seen that the reflected waves have become stronger since the dip in u and the recovery of u are both a bit more pronounced. Also, the cross-sectional area A at $t = 9.5$ s becomes smaller than the equilibrium value due to the stronger reflected wave.

It can be seen that the wave has reflected of the terminal vessel end in Figure 26. Compare for example w_2 of Figure 26 with w_2 of Figure 24. The axis are the same, but it can be seen that w_2 is no longer constant and, hence, has reflected of the vessel end.

When terminal resistance is considered, then the the FD and the DG both show qualitatively the same results but quantitatively, the results differ substantially. Note the different scale on the axis. This is, again, likely caused by the different treatment of the bifurcations.

A grid size of 2000 cells was used, and the time step Δt^n is determined by the maximum allowed time step by the CFL condition at each time t^n.

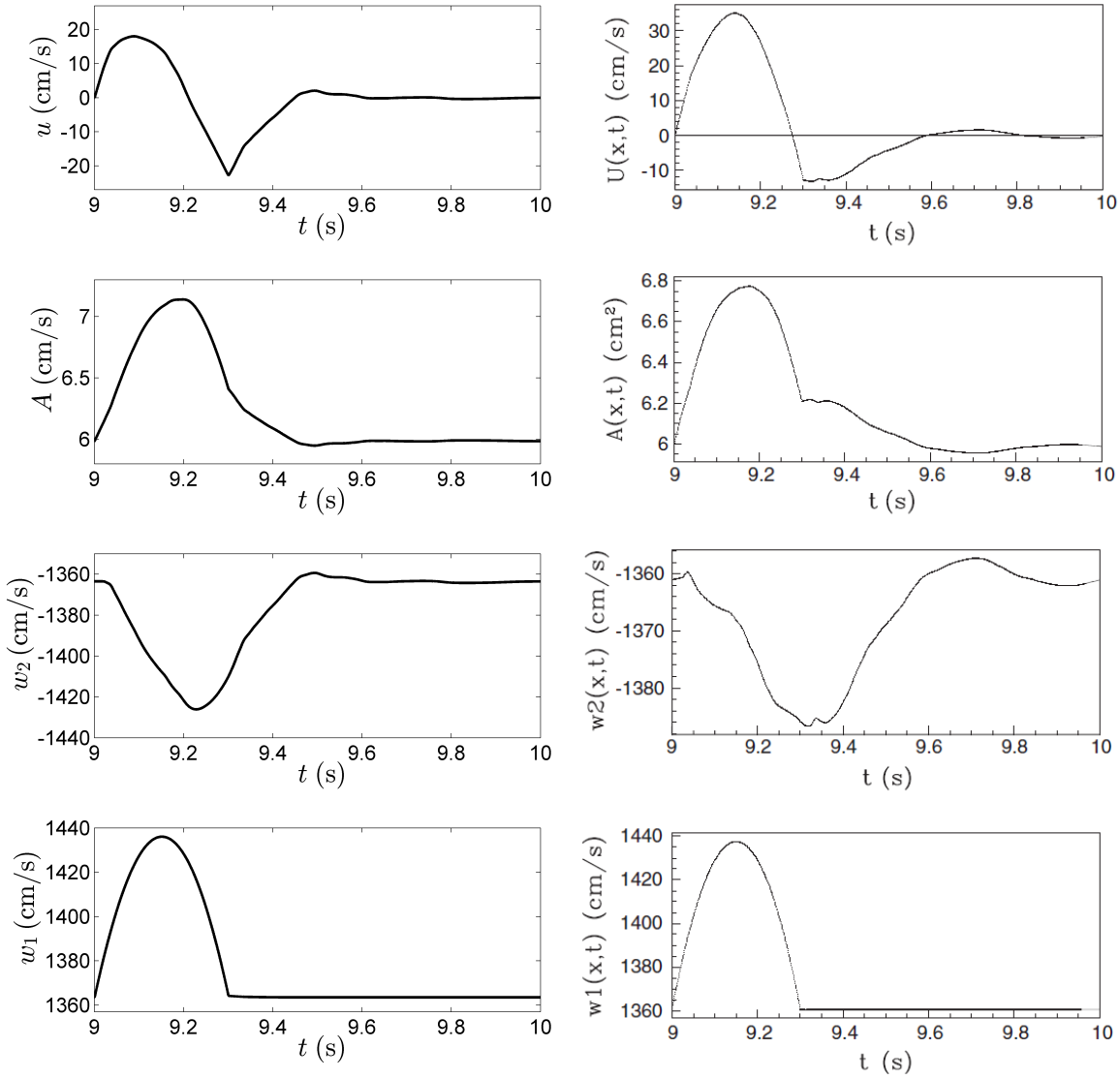

Figure 23: Results of the network of 55 arteries considering no terminal resistance shown at the start of the ascending aorta (artery 1) obtained using the high resolution flux differencing method, left, and Discontinuous Galerkin method, right, from Sherwin et al. [1]

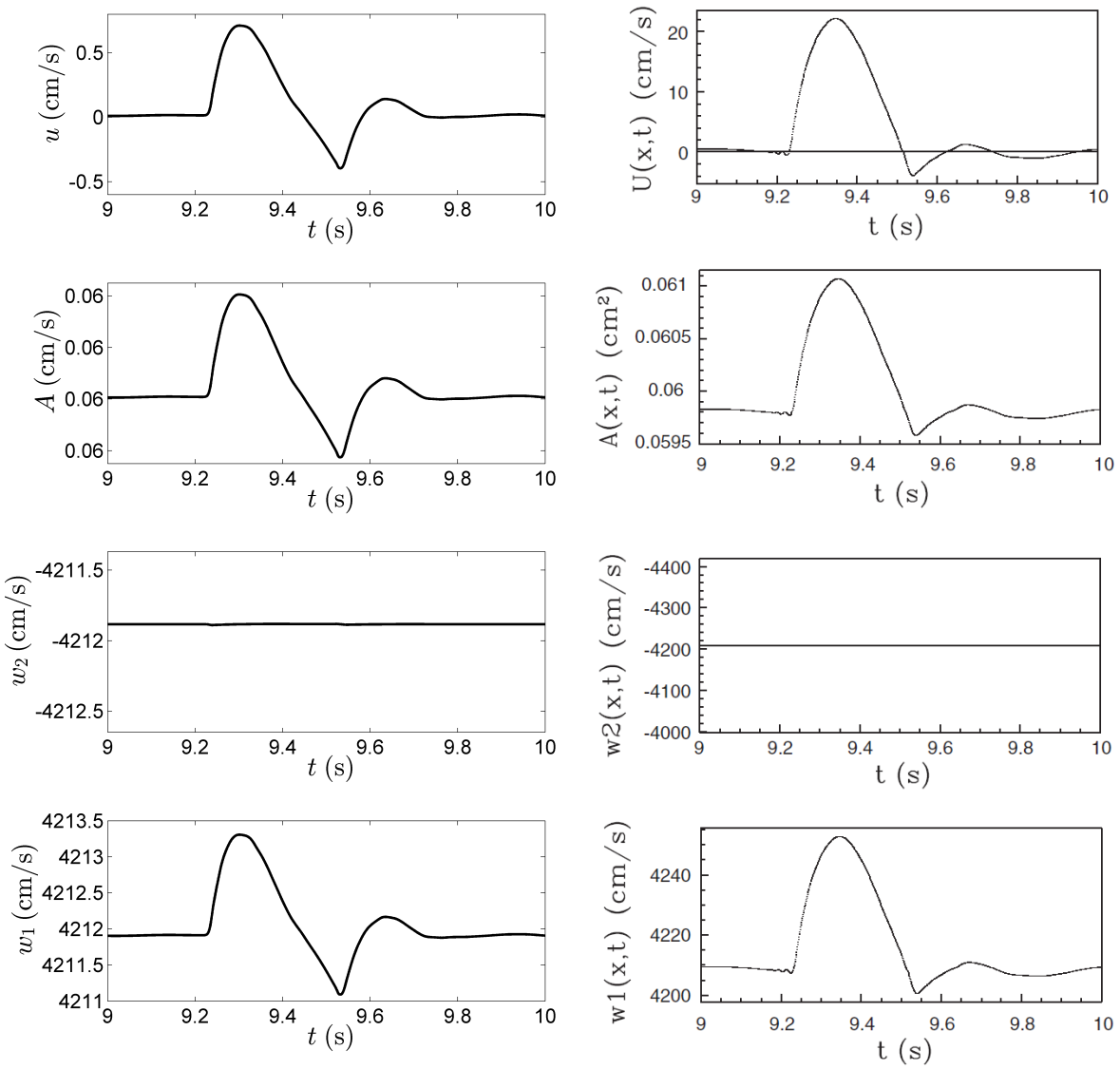

Figure 24: Results of the network of 55 arteries considering no terminal resistance shown at the start of the l. anterior tibial (artery 49, terminal vessel) using the high resolution flux differencing, left, and Discontinuous Galerkin, right, from Sherwin et al. [1]

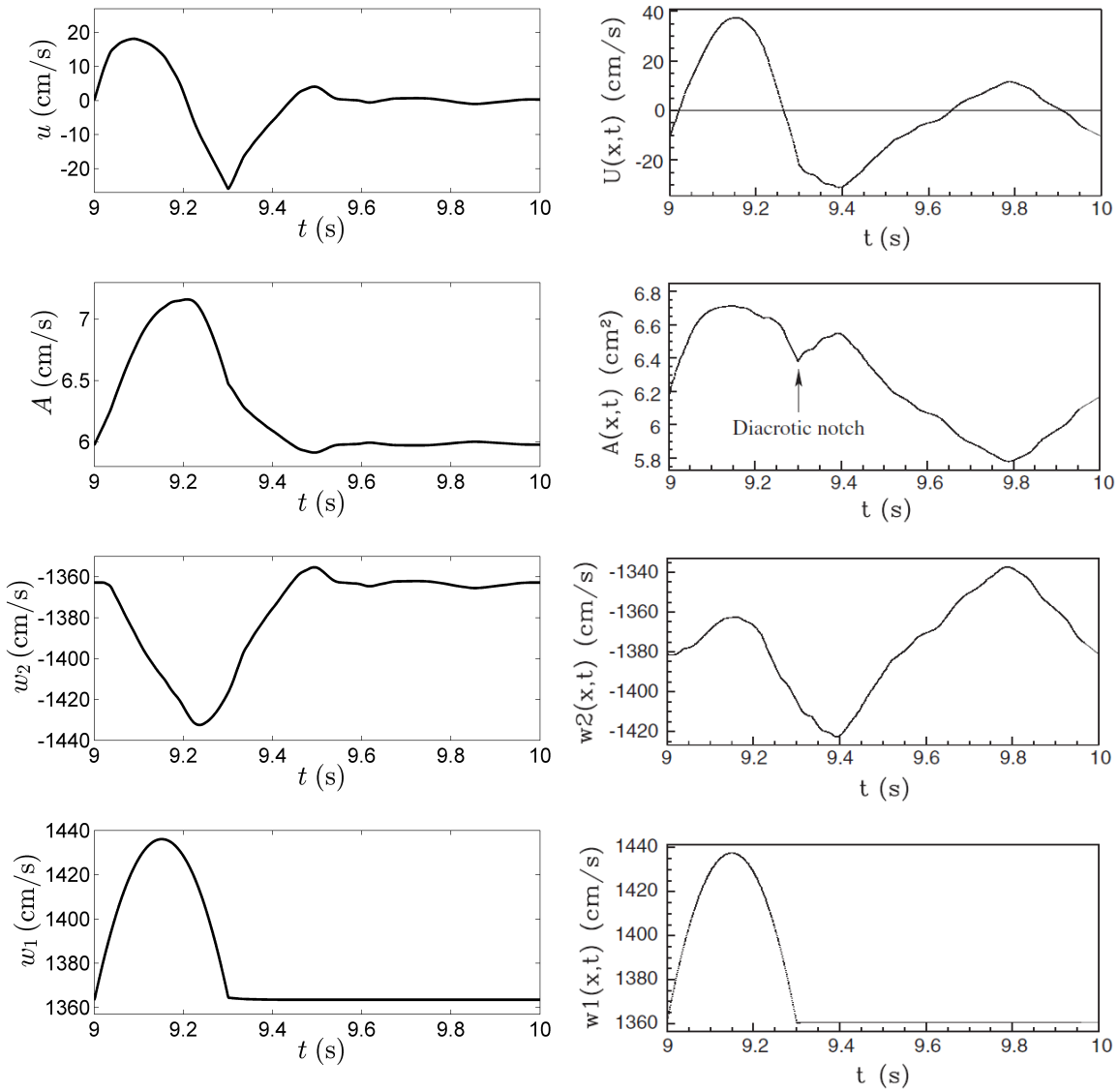

Figure 25: Results of the network of 55 arteries considering terminal resistance shown at the start of the ascending aorta (artery 1) using the high resolution flux differencing, left, and Discontinuous Galerkin, right, from Sherwin et al. [1]

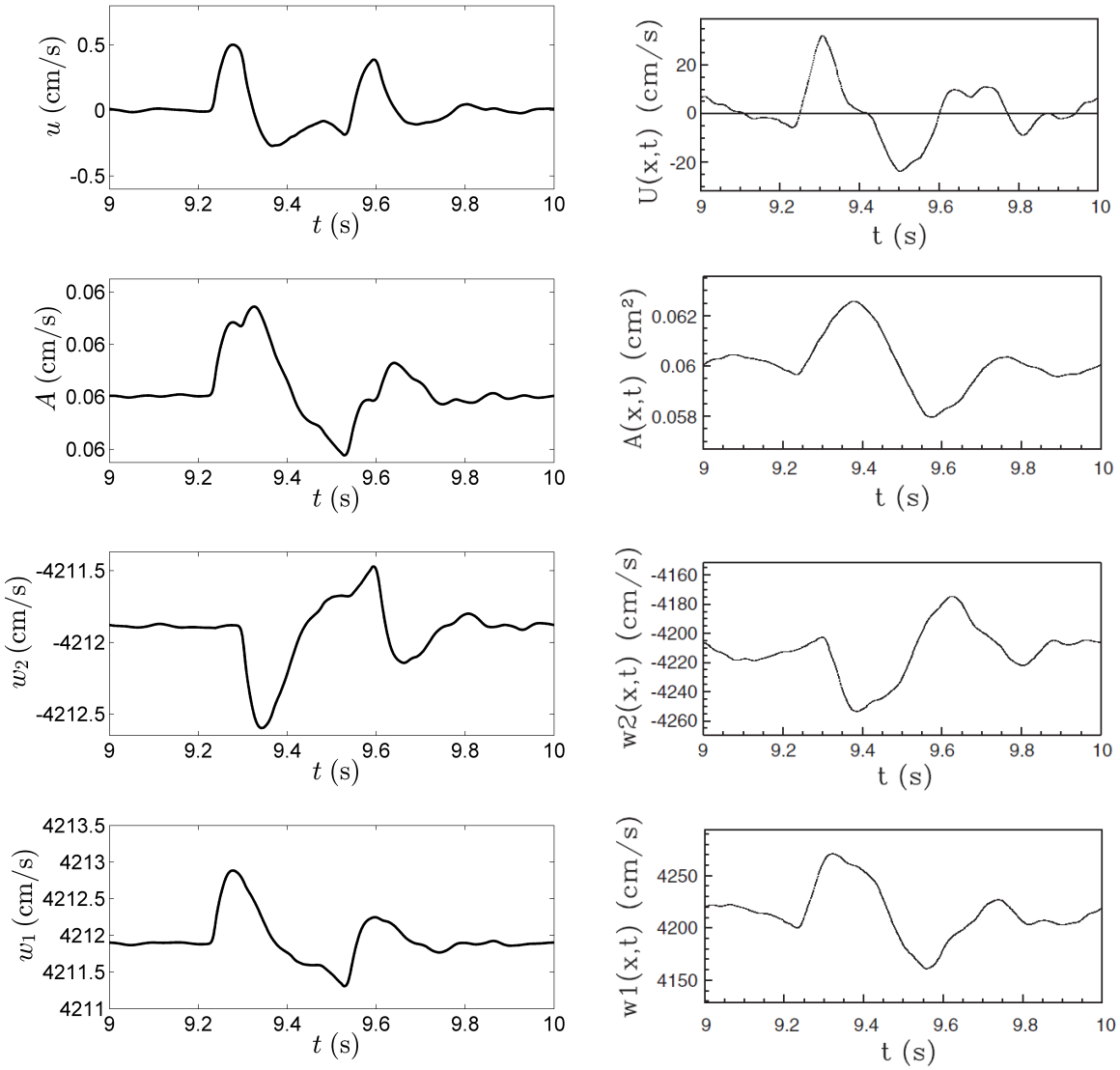

Figure 26: Results of the network of 55 arteries considering terminal resistance shown at the start of the l. anterior tibial (artery 49, terminal vessel) using the high resolution flux differencing, left, and Discontinuous Galerkin, right, from Sherwin et al. [1]

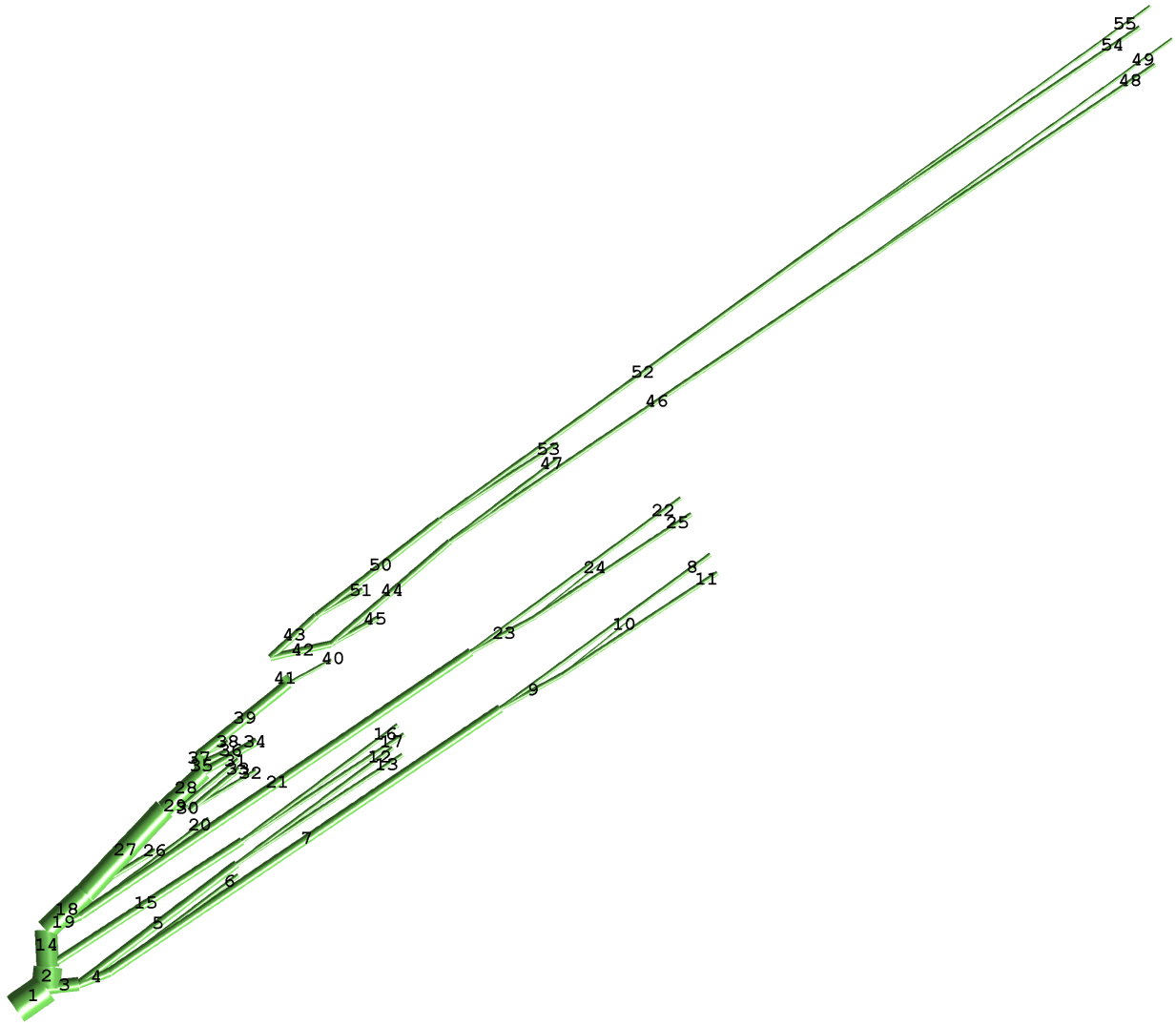

Figure 27: Two-dimensional representation of the 55 main arteries in the arterial system where the length of the arteries is on the same scale as the vessel diameters. The physiological data is given in Table 4 and Table 5

Figure 27 shows a spatial representation of the 55 main arteries in the human arterial tree at rest. The figure is made such that the vessel diameters are on the same scale as the vessel lengths. Also each artery is labelled with its corresponding number in graph 21. The normal blood vessels have their number in the centre of the blood vessel and the terminal vessels at the end of the vessel. An algorithm determined each artery's needed 'width' and using that information, the artery's length and if it should go up or down, the corresponding angle and position are calculated. Some arteries are too short for the calculated width and it therefore seems like that these arteries are not connected, but this is only a visual consequence of the used algorithm. Artery 41 for example, does not seem to be connected to arteries 42 and 43.

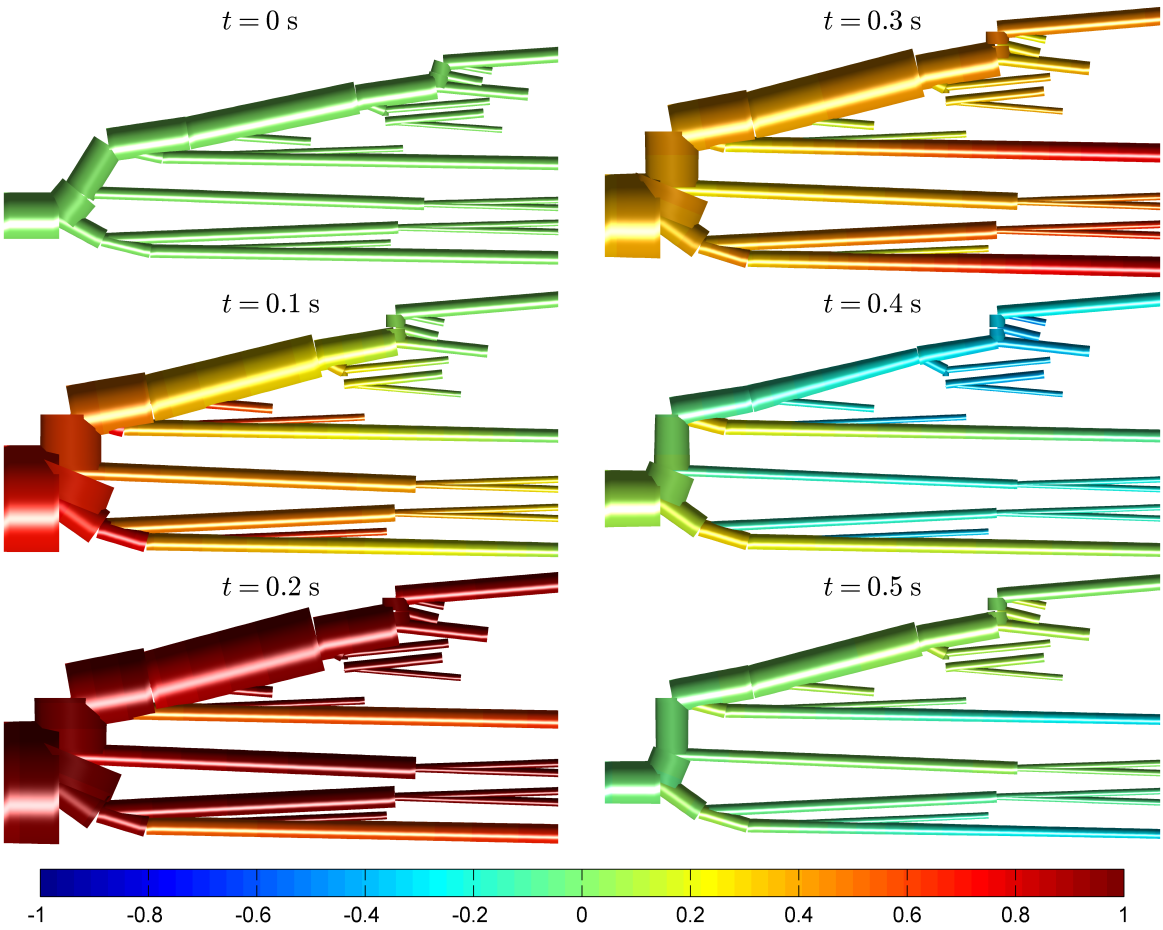

Figure 28: Part of the 55 arterial network shown for $t = 0$ s, $t = 0.1$ s, $t = 0.2$ s, $t = 0.3$ s, $t = 0.4$ s and $t = 0.5$ s. The displacement relative to equilibrium cross-sectional area A_0 is magnified with a factor of 30. The colour is relative to the maximum pressure per vessel. Maximum pressure is shown in red, no pressure in green and negative pressure is shown in blue, as illustrated by the colour bar

The start of the 55 arterial network can be found in Figure 28 for $t = 0$ s, $t = 0.1$ s, $t = 0.2$ s, $t = 0.3$ s, $t = 0.4$ s and $t = 0.5$ s. To see the expansion of the blood vessels more clearly, we have magnified the displacement of the cross-sectional area with a factor of 30. The arterial tree is in its equilibrium state at $t = 0$ s. For $t > 0$, it can be seen how the vessel tree responds to a pulse given by equation (105). After 0.3 seconds the pulse has stopped at the inlet (artery 1) and hence the system returns to its equilibrium state once all the reflections have left the domain. Those reflections also cause negative pressure at some bifurcations, as shown by the blue colour.

8 Conclusions

The results of the wave propagation algorithm were compared with the exact solution for a linear flux function. The wave propagation algorithm showed many similarities with the exact solution, but it was also found to be very diffusive.

As a further test case, the results of the high resolution flux difference splitting method were compared with the results of the Discontinuous Galerkin method for an idealised stented artery using a spatially varying nonlinear flux function. The flux differencing algorithm proved to give similar results to Discontinuous Galerkin's, the most noteworthy similarities were an increase in the peak pressure and a dip below the equilibrium pressure just before the stent. It was also shown that with constant material properties, the nonlinear system simplifies to a quasilinear system, which behaved like scalar transport equations under the considered conditions. The high resolution flux differencing method proved to be not very diffusive under these conditions, in contrast with the linear wave propagation algorithm.

The nonlinear behaviour of the flux function was shown using a long tapered artery. The nonlinearity expressed itself, in this case, by the steepening of the pulse. An artery without bifurcations that is this long, is unrealistic. Thus, luckily, no shock waves will form in normal arteries. It was shown that the actual speed at which the waves travel, is not the basic wave speed, but is instead the characteristic velocities.

Furthermore, an approach to the branching of arteries was investigated and it showed reasonable physical reflections. If the parent and daughter arteries are equally stiff and if the sum of the daughter vessel's cross-section area is comparable to that of the parent, then the magnitude of the reflected pressure pulse is in the order of 10%.

As a final test case, a simplified model consisting of the 55 main arteries in the human arterial tree is considered, to study the reflections at the bifurcations in the arterial tree. To simulate the missing arteries a simplified model of terminal resistance is used. The high resolution flux differencing algorithm showed the same qualitative results, but the quantitative results differed considerably, when compared to the results of the Discontinuous Galerkin. The discrepancy is likely caused by the treatment of the bifurcations. It can be concluded that bifurcations play an important role in the overall behaviour of blood flow and, as such, cannot be neglected when considering a single artery. The results from the Discontinuous Galerkin are from Sherwin et al. [1].

Future research could try a more advanced treatment of the bifurcations in order to get more comparable results to the Discontinuous Galerkin's. Moreover, further research can focus on the wave pattern created by an endovascular prosthesis or a surgical bypass, since this would be a straightforward extension of the methods presented in this thesis.

References

[1] Sherwin S, Formaggia L, Peiró J, Franke V. Computational modelling of 1D blood flow with variable mechanical properties and its application to the simulation of wave propagation in the human arterial system. *International Journal for Numerical Methods in Fluids*. 2003;43(6-7):673-700. doi:10.1002/fld.543.

[2] Formaggia L, Veneziani A. Reduced and multiscale model for the human cardiovascular system. *Lecture notes VKI Lecture Series*. Brussels, Belgium; 2003.

[3] LeVeque R. *Finite Volume Methods For Hyperbolic Problems*. Cambridge [u.a.]: Cambridge Univ. Press; 2011.

[4] Bale D, LeVeque R, Mitran S, Rossmanith J. A Wave Propagation Method for Conservation Laws and Balance Laws with Spatially Varying Flux Functions. *SIAM Journal on Scientific Computing*. 2003;24(3):955-978. doi:10.1137/s106482750139738x.

[5] Mynard J. *One-dimensional Blood Flow Modelling with the Locally Conservative Galerkin (LCG) Method*. [master's thesis]. Swansea, UK: Swansea University; 2007.

[6] Guaily A, Epstein M. Boundary conditions for hyperbolic systems of partial differentials equations. *Journal of Advanced Research*. 2013;4(4):321-329. doi:10.1016/j.jare.2012.05.006.

[7] High-resolution scheme. *Enwikipediaorg*. 2017. Available at: `https://en.wikipedia.org/wiki/High-resolution_scheme`. Accessed August 20, 2017.

[8] Lax P, Liu X. Solution of Two-Dimensional Riemann Problems of Gas Dynamics by Positive Schemes. *SIAM Journal on Scientific Computing*. 1998;19(2):319-340. doi:10.1137/s1064827595291819.

[9] Peiró J, Sherwin S, Parker K et al. Numerical simulation of arterial pulse propagation using one-dimensional models. In: Collins M, Pontrelli G, Atherton M, ed. *Wall-Fluid Interactions In Physiological Flows*. WIT Press; 2004:1-36.

[10] Formaggia L, Nobile F, Quarteroni A, Veneziani A. Multiscale modelling of the circulatory system: a preliminary analysis. Computing and Visualization in Science. 1999;2(2-3):75-83. doi:10.1007/s007910050030.

[11] Centimetre–gram–second system of units. *Enwikipediaorg*. 2017. Available at: `https://en.wikipedia.org/wiki/Centimetre%E2%80%93gram%E2%80%93second_system_of_units`. Accessed August 20, 2017.

Appendices

A The CGS system

The conversion from the CGS system to the standard SI system is shown in Table 3, which is an adaptation from [11].

Table 3: Conversion Table from the CGS system to the SI units

Quantity	CGS unit	Definition	SI
length	centimetre (cm)	1/100 of metre	$=10^{-2}$ m
mass	gram (g)	1/1000 of kilogram	$=10^{-3}$ kg
time	second (s)	1 second	=1 s
velocity	centimetre per second (cm/s)	cm/s	$=10^{-2}$ m/s
acceleration	gal (Gal)	cm/s^2	$=10^{-2}$ m/s^2
force	dyne (dyn)	$g{\cdot}cm/s^2$	$=10^{-5}$ N
energy	erg (erg)	$g{\cdot}cm^2/s^2$	$=10^{-7}$ J
power	erg per second (erg/s)	$g{\cdot}cm^2/s^3$	$=10^{-7}$ W
pressure	barye (Ba)	$g/cm{\cdot}s^2$	$=10^{-1}$ Pa
dynamic viscosity	poise (P)	g/cm·s	$=10^{-1}$ Pa·s
kinematic viscosity	stokes (St)	cm^2/s	$=10^{-4}$ m^2/s

B Physiological data of the 55 arteries

The physiological data from Sherwin et al. [1] for the 55 main arteries in the human arterial system can be found in Table 4 and Table 5.

Table 4: Physiological data of the 55 main arteries from [1]

#	Artery	Length (cm)	Area (cm^2)	β($kg\ s^{-2}\ cm^{-2}$)	R_t
1	Ascending Aorta	4.0	5.983	97	—
2	Aortic Arch I	2.0	5.147	87	—
3	Brachiocephalic	3.4	1.219	233	—
4	R. Subclavian I	3.4	0.562	423	—
5	R. Carotid	17.7	0.432	516	—
6	R. Vertebral	14.8	0.123	2590	0.906
7	R. Subclavian II	42.2	0.510	466	—
8	R. Radial	23.5	0.106	2866	0.82
9	R. Ulnar I	6.7	0.145	2246	—
10	R. Interosseous	7.9	0.031	12894	0.956
11	R. Ulnar II	17.1	0.133	2446	0.893
12	R. Internal Carotid	17.6	0.121	2644	0.784
13	R. External Carotid	17.7	0.121	2467	0.79
14	Aortic Arch II	3.9	3.142	130	—
15	L. Carotid	20.8	0.430	519	—
16	L. Internal Carotid	17.6	0.121	2644	0.784
17	L. External Carotid	17.7	0.121	2467	0.791
18	Thoracic Aorta I	5.2	3.142	124	—
19	L. Subclavian I	3.4	0.562	416	—
20	Vertebral	14.8	0.123	2590	0.906
21	L. Subclavian II	42.2	0.510	466	—
22	L. Radial	23.5	0.106	2866	0.821
23	L. Ulnar I	6.7	0.145	2246	—
24	L. Interosseous	7.9	0.031	12894	0.956
25	L. Ulnar II	17.1	0.133	2446	0.893
26	Intercostals	8.0	0.196	885	0.627
27	Thoracic Aorta II	10.4	3.017	117	—
28	Abdominal I	5.3	1.911	167	—
29	Celiac I	2.0	0.478	475	—
30	Celiac II	1.0	0.126	1805	—
31	Hepatic	6.6	0.152	1142	0.925
32	Gastric	7.1	0.102	1567	0.921
33	Splenic	6.3	0.238	806	0.93
34	Superior Mesenteric	5.9	0.430	569	0.934
35	Abdominal II	1.0	1.247	227	—
36	L. Renal	3.2	0.332	566	0.861
37	Abdominal III	1.0	1.021	278	—
38	R. Renal	3.2	0.159	1181	0.861
39	Abdominal IV	10.6	0.697	381	—
40	Inferior Mesenteric	5.0	0.080	1895	0.918
41	Abdominal V	1.0	0.578	399	—
42	R. Common Iliac	5.9	0.328	649	—
43	L. Common Iliac	5.8	0.328	649	—
44	L. External iliac	14.4	0.252	1493	—
45	L. Internal Iliac	5.0	0.181	3134	0.925
46	L. Femoral	44.3	0.139	2559	—
47	L. Deep Femoral	12.6	0.126	2652	0.885
48	L. Posterior Tibial	32.1	0.110	5808	0.724
49	L. Anterior Tibial	34.3	0.060	9243	0.716

Table 5: Physiological data of the 55 main arteries from [1] continued

#	Artery	Length (cm)	Area (cm^2)	β ($kg\,s^{-2}\,cm^{-2}$)	R_t
50	R. External Iliac	14.5	0.252	1493	—
51	R. Internal Iliac	5.1	0.181	3134	0.925
52	R. Femoral	44.4	0.139	2559	—
53	R. Deep Femoral	12.7	0.126	2652	0.888
54	L. Posterior Tibial	32.2	0.110	5808	0.724
55	R. Anterior Tibial	34.4	0.060	9243	0.716

C Numerical implementation

All the created Matlab scripts can be found at `https://tinyurl.com/BEP-Marco-Rozendaal` (original link `https://drive.google.com/drive/folders/0B7N-e57lbo5FOFcySFpBUUxpeVk?usp=sharing`). The bifurcation script is included below:

```
clc; clear all; close all;
tic

figs_path = 'C:\Users\Marco\Documents\TU Delft\BEP\LaTeX BEP\Figures';
font = 15;
linewidth = 2;

save = 1;



%%
%Bifurcation
% parameters
load('55_art_data.mat')
p_ext = 0;
rho = 1.021;

if save==1||save==2
    %           1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21
       22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
       43 44 45 46 47 48 49 50 51 52 53 54 55
    parent_p = [0 1 1 3 3 4 4 7 7 9  9  5  5  2  2  15 15 14 14 19 19
       21 21 23 23 18 18 27 27 29 29 30 30 28 28 35 35 37 37 39 39 41
       41 42 42 44 44 46 46 43 43 50 50 52 52];
    L = L(1:length(parent_p))';
    A_0_s =  A_0_s(1:length(parent_p))';
    beta_s = 1000*beta_s(1:length(parent_p))';
    if save==1
        R_t = zeros(size(parent_p));
    else
        R_t = R_t(1:length(parent_p))';
    end
elseif save==3
    L = [3 3 3]; %[4 2 3.4 3.4 17.7 14.8 42.2 23.5 6.7 7.9 17.1 17.6
       17.7];%[4 2 3.4 3.4 17.7];
    A_0_s =  [6 3 3]; %[5.983 5.147 1.219 0.562 0.432 0.123 0.51 0.106
       0.145 0.031 0.133 0.121 0.121]; %[5.983 5.147 1.219 0.562
       0.432];
```

```
    beta_s = [1 1 1]*10^5; %100000; [97 87 233 423 516 2590 466 2866
        2246 12894 2446 2644 2467];%[97 87 233 423 516];
    R_t = [0 0 0];
elseif save==4
    L = [3 3 3];
    A_0_s =  [6 3 3];
    beta_s = [1 2 1]*10^5;
    R_t = [0 0 0];
elseif save==5
    L = [3 3 3];
    A_0_s =  [6 6 3];
    beta_s = [1 1 1]*10^5;
    R_t = [0 0 0];
end

if save==1||save==2
    %network 55 arteries
    T_end = 10;
    N = 2000; %2000 spacesize

    % left BC - prescribed A(0,t)
    T_p = 1;
    delta_A = @(t) sin(2*pi*t/T_p+0.628)-0.588;
    A_BC = @(t) A_0_s(1) + 1.597*delta_A(t) .* (delta_A(t)>=0);

    TPLOT = linspace(0,T_end,10000);
    plot(TPLOT,A_BC(TPLOT),'k','linewidth',linewidth)%,TPLOT,delta_A(
        TPLOT))
    ylabel('$A$ (cm$^2$)','Interpreter','LaTeX','fontsize',font)
    xlabel('$t$ (s)','Interpreter','LaTeX','fontsize',font)
    set(gca,'fontsize',font)
    set(gcf, 'PaperUnits', 'centimeters');
    set(gcf, 'PaperPosition', [0 0 25 6]);
    print(fullfile(figs_path,'network_55_A_inflow_bc'),'-dpng','-r500',
        '-opengl')
else
    % test cases
    T_end = 0.1;
    N = 300;
    parent_p = [0 1 1];
    % left BC - prescribed A(0,t)
    T_p = 0.02;
    A_BC = @(t) A_0_s(1) + 0.1*exp(-(t-0.01).^2*10^6);
end
%ADJ = [0 1 1 0 0 0 0 0 0; 0 0 0 0 0 0 0 0 0; 0 0 0 1 1 0 0 0 0; 0 0 0
    0 0 1 1 0 0; 0 0 0 0 0 0 0 0 0; 0 0 0 0 0 0 0 0 0; 0 0 0 0 0 0 0 1
    1; 0 0 0 0 0 0 0 0 0; 0 0 0 0 0 0 0 0 0];%[0 1 1 0 0; 0 0 0 0 0; 0 0
     0 1 1; 0 0 0 0 0; 0 0 0 0 0];


%parent pointer -> adj matrix
for i=1:length(parent_p)
    for j=1:length(parent_p)
        if parent_p(j)==i
            ADJ(i,j) = 1;
        else
```

```
            ADJ(i,j) = 0;
        end
    end
end

dx = sum(L)/(N+1);
x_pos = (dx:dx:(sum(L)-dx))';

disp(['dx<min(L):' num2str(dx<min(L))])

% Changing A_0, beta_0 such that they contain N elements
K_r = [diag(ones(N,1),0); zeros(1,N)];
K_l = [zeros(1,N); diag(ones(N,1),0)];
K_c = [diag(ones(N,1),0); zeros(1,N)];
K_ibc = [diag([1 zeros(1,N-1)]); zeros(1,N)];
K_obc = zeros(N+1,N);


K_z1_up = diag(ones(N,1),-1); %v interfaces > z_1 interfaces
K_z2_up = diag(ones(N,1),1);
K_z1 = [diag(ones(N,1),0) zeros(N,1)]; % Fundamental different, it
   relates Z's/interfaces to the cells
K_z2 = [zeros(N,1) diag(ones(N,1),0)]; % ^


A_0 = [];
beta = [];
rest = 0;
for i=1:1:length(A_0_s)
    cells_unr = (L(i) + rest - dx*(i==length(A_0_s)))/dx;
    rest = (cells_unr - floor(cells_unr))*dx;
    cells(i) = fix(cells_unr)*(i~=length(A_0_s)) + round(cells_unr)*(i
       ==length(A_0_s));
    A_0 = [A_0; repmat(A_0_s(i),cells(i),1)];
    beta = [beta; repmat(beta_s(i),cells(i),1)];
end

%sum over ADJ
for i=1:length(A_0_s)
    total_area = ADJ(i,:)*A_0_s';
    if total_area == 0
        total_area = 1;
    end
    K_obc( sum(cells(1:i))+1 , sum(cells(1:i)) ) = ( sum(ADJ(i,:))==0 )
       ;
    for j=1:length(A_0_s)
        % Coeff of bifurcations
        alpha(i,j) = A_0_s(j)*ADJ(i,j)/total_area;
        % fixing K's
        K_r( sum(cells(1:i))+1 , sum(cells(1:(j-1)))+1 ) = ADJ(i,j);
        K_c( sum(cells(1:i))+1 , sum(cells(1:(j-1)))+1 ) = alpha(i,j);

        K_z1( sum(cells(1:(j-1)))+1 , sum(cells(1:i))+1 ) = alpha(i,j);
        K_z1_up( sum(cells(1:(j-1)))+2 , sum(cells(1:i))+1 ) = ADJ(i,j)
           ;
```

```
        K_z2_up( sum(cells(1:i))+1 , sum(cells(1:(j-1)))+2 ) = ADJ(i,j)
            ;
    end
end

Lin = ones(N+1,1);
Lin(cumsum(cells)+1) = (sum(ADJ,2)==0);
%%
M = 60000; % Guess for time dimension

% functions
c = @(A,j) sqrt(beta(j)/(2*rho)).*A.^(1/4);
p = @(A,j) p_ext + beta(j) .* ( sqrt(A) - sqrt(A_0(j)) );
F = @(A, u, beta, A_0) [ A.*u  1/2*u.^2 + ( p_ext + beta.*(sqrt(A)-sqrt
    (A_0)) )/rho ];

% limiters
% minmod
phi = @(theta) max(0, min(1,theta));

% initialisation
U = zeros(N,2,M); % [A u]
U(:,:,1) = [A_0 zeros(N,1)]; % ICs

beta_ibc = K_ibc*beta;
beta_obc = K_obc*beta;

A_0_obc = K_obc*A_0;
A_0_ibc = K_ibc*A_0;

A_0_r = K_r*A_0 + A_0_obc;
A_0_l = K_l*A_0 + A_0_ibc;

beta_r = K_c*beta + beta_obc;
beta_l = K_l*beta + beta_ibc;



dt = 10^(-4);
t = zeros(1,M);

i = 1;
while t(i)<=T_end
    if mod(i,500)==0
        fprintf('i=%i, t=%6.4f, elapsedtime=%5.2f, eta=%5.2f\n', i, t(i)
            , toc/60, (T_end-t(i))/t(i)*toc/60)
    end

    % Inlet BC
    W_2 = 1/2*U(1,2,1) - 2*c(U(1,1,1),1); % fixed at initial state
    W_1 = W_2 + 4*sqrt(beta(1)/(2*rho))*A_BC(t(i))^(1/4);
    U_ibc = [((W_1-W_2)/4)^4*(2*rho/beta(1))^2 W_1+W_2; zeros(N,2)];

    % Outlet BCs - copy pasta
    U_obc = zeros(N,2);
```

```
U_obc(cumsum(cells),:) = [U(cumsum(cells),1,i) (ones(length(A_0_s)
    ,1)-R_t').*(U(cumsum(cells),2,1)+U(cumsum(cells),2,i) + 4*( c(U(
    cumsum(cells),1,i),cumsum(cells)) - c(U(cumsum(cells),1,1),
    cumsum(cells))) ) - U(cumsum(cells),2,i)];
% filtering
U_obc =  K_obc*U_obc;


% Some simplyfing steps
U_h = U(:,:,i);

A_r = K_r*U_h(:,1) + U_obc(:,1);
A_l = K_l*U_h(:,1) + U_ibc(:,1);

u_r = K_c*U_h(:,2) + U_obc(:,2);
u_l = K_l*U_h(:,2) + U_ibc(:,2);

%Real code - Flux difference - matrix
F_r = F(A_r, u_r, beta_r, A_0_r);
F_l = F(A_l, u_l, beta_l, A_0_l);
delta = F_r-F_l;



r_r = A_r.^(3/4)./sqrt(beta_r/(2*rho));
r_l = A_l.^(3/4)./sqrt(beta_l/(2*rho));

det = r_r + r_l;

beta_1 = 1./det.*(delta(:,1) + r_l.*delta(:,2));
beta_2 = 1./det.*(-delta(:,1) + r_r.*delta(:,2));


% Waves
Z_1 = bsxfun(@times, beta_1, [r_r ones(N+1,1)]);
Z_2 = bsxfun(@times, beta_2, [-r_l ones(N+1,1)]);


% High-res based on Leveque
theta_1 = dot(K_z1_up*Z_1, Z_1, 2)./dot(Z_1, Z_1, 2);
theta_2 = dot(K_z2_up*Z_2, Z_2, 2)./dot(Z_2, Z_2, 2);

theta_1(~isfinite(theta_1)) = 0;
theta_2(~isfinite(theta_2)) = 0;

Z_1_t = bsxfun(@times, phi(theta_1), Z_1);
Z_2_t = bsxfun(@times, phi(theta_2), Z_2);


s_1 = U_h(:,2) + c(U_h(:,1),1:N);
s_2 = U_h(:,2) - c(U_h(:,1),1:N);
s_1_i = K_c*s_1;
s_2_i = K_l*s_2;

F_c = 1/2*( bsxfun(@times, sign(s_1_i).*(1-dt/dx*abs(s_1_i)), Z_1_t
    )  +  bsxfun(@times, sign(s_2_i).*(1-dt/dx*abs(s_2_i)), Z_2_t) )
```

```
        ;

    % Make it linear at bifurcations
    F_c = bsxfun(@times, F_c, Lin);

    % Update cell avg
    U(:,:,i+1) = U(:,:,i) - dt/dx*( K_z1*Z_1 + K_z2*Z_2 ) - dt/dx*(
        K_z2*F_c - K_z1*F_c ); %K_z2 and K_z1 good


    % Update time
    t(i+1) = t(i) + dt;

    %determine dt for cfl
    lambda_max(i) = max(abs([s_1; s_2]));

    dt_cfl(i) = dx/lambda_max(i);
    dt = dt_cfl(i);
    i = i+1;
end

U(:,:,~any(any(U)))=[];
t(t==0)=[];
t = [0 t];
toc
indeces = repmat((1:N)',1,length(t));
p_min = min(min( p(squeeze(U(:,1,:)), indeces) ));
p_max = max(max( p(squeeze(U(:,1,:)), indeces) ));
u_min = min(min( U(:,2,:) ));
u_max = max(max( U(:,2,:) ));
A_min = min(min( U(:,1,:) ));
A_max = max(max( U(:,1,:) ));
%% reflection plots
if save==3 || save==4 || save == 5
    close(figure(9))
    figure(9)
    t_stamp = [0.014 0.0186 0.023];
    linetype = {'k:' 'k--' 'k'};
    ax_y_lims = [-200 2500];
    ax_x_lims = [0 sum(L)];

    x_b = reshape(bsxfun(@times, cumsum(L), ones(3,length(cells)))
        ,1,[]);%
    y_b = repmat([ax_y_lims nan], 1,length(cells));

    for i=1:length(t_stamp)
        index_t = find(t>t_stamp(i),1);
        p_h = p(U(:,1,index_t),1:N);
        p_plot = nan(1,N+2);
        x_plot = [];
        for j=1:length(cells)
            p_plot( sum(cells(1:j-1))+j:sum(cells(1:j))+j-1 ) = p_h(sum
                (cells(1:j-1))+1:sum(cells(1:j)));
            x_plot = [x_plot linspace(sum(L(1:j-1)), sum(L(1:j)), cells
                (j)) nan];
        end
```

```
        plot(x_plot(1:end-1), p_plot, linetype{i}, 'linewidth',
            linewidth);
        hold all
    end
    plot(x_b(1:end-2), y_b(1:end-2),'k', 'linewidth',linewidth)
    ylim(ax_y_lims)
    xlim(ax_x_lims)
    ylabel('$p$ (dyne/cm$^2$)','Interpreter','LaTeX','fontsize',font)
    xlabel('$x$ (cm)','Interpreter','LaTeX','fontsize',font)
    set(gca,'xtick',0:sum(L))
    set(gca,'xticklabel',{'0' '1' '2' '3' '4' '5' '3' '4' '5' '6'})
    set(gca,'fontsize',font)
    legend({num2str(t_stamp')},'fontsize',10)
    text(cumsum(L)-L/2-0.15, ones(1,length(L))*0.9*ax_y_lims(2), {'$P$'
        '$D_1$' '$D_2$'},'Interpreter','LaTeX','fontsize',font)
    legend(gca,'boxoff')

    set(gcf, 'PaperUnits', 'centimeters');
    set(gcf, 'PaperPosition', [0 0 25 8]);
    if save == 3
        print(fullfile(figs_path,'testcase_bifurcation_equal'),'-dpng',
            '-r500','-opengl')
    end
    if save == 4
        print(fullfile(figs_path,'
            testcase_bifurcation_equal_A0_diff_beta'),'-dpng','-r500','-
            opengl')
    end
    if save == 5
        print(fullfile(figs_path,'
            testcase_bifurcation_diff_A0_equal_beta'),'-dpng','-r500','-
            opengl')
    end
end
%% Connectivity Graph
close(figure(13))
if save==1 || save==2
    eps = 0.0001;
    figure(13)
    treeplot(parent_p);
    h = findobj(gca,'Type','line');
    set(h,'color','black','linewidth',linewidth);
    [x_c,y_c] = treelayout(parent_p);

    hold on
    plot(x_c,y_c,'o','color',[eps 0 0],'markersize',7,'linewidth',9)
    text(x_c(1:9),y_c(1:9), num2str((1:9)'),'color',[1-eps 1 1], '
        FontName','FixedWidth', 'fontweight','bold', 'VerticalAlignment'
        ,'middle', 'HorizontalAlignment','center');
    h=text(x_c(10:end),y_c(10:end), num2str((10:length(x_c))'),'color'
        ,[1-eps 1 1], 'FontName','FixedWidth', 'fontweight','bold', '
        VerticalAlignment','middle', 'HorizontalAlignment','center');

    set(gca,'visible','off')
    set(gca,'position',[0 0 1 1]);
    set(gcf, 'PaperUnits', 'centimeters');
```

```
    set(gcf, 'PaperPosition', [0 0 18 10]);

    print(fullfile(figs_path,'network_55_connection_graph'),'-dpng','-
        r500','-opengl')
end

%% Ascending aorta en 49
if save == 1 || save==2
    artery = [1 49];
    T_start = 9;
    t_plot = t(t>=T_start);
    index_t = length(t)-length(t_plot)+1:length(t);
    y_label = {'$u$ (cm/s)' '$A$ (cm/s)' '$w_2$ (cm/s)' '$w_1$ (cm/s)'
        };
    x_label = {'$t$ (s)'};
    y_lim = [-27 27; 5.8 7.3; -1440 -1350; 1357 1440;   -0.6 0.8;
        0.059975 0.060045; -4212.65 -4211.37; 4211 4213.5];
    axis_lim = [ones(length(artery)*4,1)*T_start ones(length(artery)
        *4,1)*T_end y_lim];


    for i=1:length(artery)
        close(figure(i))
        figure(i)

        index_x = sum(cells(1:artery(i)-1))+1;
        data = [squeeze(U(index_x,2,index_t)) squeeze(U(index_x,1,
            index_t)) squeeze(U(index_x,2,index_t)) - 4*c(squeeze(U(
            index_x,1,index_t)),index_x) squeeze(U(index_x,2,index_t)) +
             4*c(squeeze(U(index_x,1,index_t)),index_x)];
        size(data)

        for j=1:4
            subplot(4,1,j)
            plot(t_plot, data(:,j),'k','linewidth', linewidth)
            ylabel(y_label(j),'Interpreter','LaTeX','fontsize',font)
            xlabel(x_label ,'Interpreter','LaTeX','fontsize',font)
            set(gca,'fontsize',font)
            axis(axis_lim(j+4*(i-1),:))
            if j==1
                POS = get(gca,'position');
                POS = POS + [0.05 0.0045 -0.01 0];
                delta_y = [0 -0.235 0 0];
            end
            set(gca,'position', POS +  delta_y*(j-1))
        end

        set(gcf, 'PaperUnits', 'centimeters');
        set(gcf, 'PaperPosition', [0 0 15 30]);
        if save==1
            print(fullfile(figs_path,['network_55_artery_'  num2str(
                artery(i)) '_4']), '-dpng','-r500','-opengl')
        elseif save == 2
            print(fullfile(figs_path,['network_55_artery_'  num2str(
                artery(i)) '_4_R_t']), '-dpng','-r500','-opengl')
        end
```

```
    end
end
%% Make 2d tree structure
%load('U_55_art.mat');
d = 0.6;%0.8 of 0.9 |0.5
s = 0.5; %0.3 en 0.7|0.4
width = zeros(1,length(parent_p));
rel_width = zeros(1,length(parent_p));
%----------  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19
   20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
    43 44 45 46 47 48 49 50 51 52 53 54 55
direction = [-1  1 -1 -1  1  1 -1  1 -1  1 -1  1 -1  1 -1  1 -1  1 -1
   1 -1  1 -1  1 -1 -1  1  1 -1 -1  1 -1  1 -1  1 -1  1 -1  1 -1  1 -1
    1 1  -1 -1  1 -1  1 1  -1  1 -1 -1  1];%;[(-1).^(1:25) -(-1)
   .^(26:55)] - Switch(28); % (-1).^(1:length(parent_p))
direction = direction(1:length(parent_p));

% ONLY BIFURCATIONS
%Aslang as there are zero width vessels:
while ~all(width(2:end)) %excluding the first vessel cuz zero width
    for j=2:length(parent_p)
        if width(j)==0
            childs = [];
            childsHaveWidth = true;

            for i=1:length(parent_p)
                if j==parent_p(i)
                    %i.e. j is connected to i
                    childs = [i childs];
                    if width(i)==0
                        %not assigned yet
                        childsHaveWidth = false;
                    end
                end
            end

            if isempty(childs)
                % no childs or a leaf
                width(j) = direction(j)*d;
                rel_width(j) = width(j) + direction(j)*s/2;
            elseif childsHaveWidth
                %bifurcations -> 2 childs only
                if direction(j) == -direction(childs(1))
                    width(j) = -rel_width(childs(1));
                    rel_width(j) = rel_width(childs(2)) - rel_width(
                       childs(1));
                else %-1
                    width(j) = -rel_width(childs(2));
                    rel_width(j) = rel_width(childs(1)) - rel_width(
                       childs(2));
                end
            end
        end
    end
end
L_e = max((cells-1)*dx,abs(width));
```

```
phi = asin(width./L_e);
x_off = 0;
y_off = 0;

for i=2:length(parent_p)
    x_off = [x_off x_off(parent_p(i)) - L_e(parent_p(i))*sin(phi(
        parent_p(i)))];
    y_off = [y_off y_off(parent_p(i)) + L_e(parent_p(i))*cos(phi(
        parent_p(i)))];
end
z_off = zeros(size(parent_p));

%% plots of part of 3d network
if save==1||save==2
    close(figure(5))
    res = 10;
    stretch = 30;
    t_stamp = [0 0 0.1 0.2 0.3 0.4 0.5];
    terminal_ves = (sum(ADJ,2)==0)';
    normal_ves = ones(size(parent_p))-terminal_ves;

    for i=1:length(t_stamp)
        t_index = find(t>t_stamp(i),1);
        x2 = [];
        y2 = [];
        z2 = [];
        c2 = [];
        for j=1:length(parent_p)
            x_i = sum(cells(1:j-1))+1:sum(cells(1:j));

            [x1, y1, z1, c1] = makeVessel(U(x_i, 1, t_index), A_0(x_i),
                beta(x_i), res, phi(j), dx, stretch);
            x2 = [x2; nan(1,res+1); x1 + x_off(j)];
            y2 = [y2; nan(1,res+1); y1 + y_off(j)];
            z2 = [z2; nan(1,res+1); z1 + z_off(j)];
            c2 = [c2; nan(1,res+1); c1 / p(A_max_ves(j), sum(cells(1:j
                -1))+1 )]; %new
        end

        figure(5)
        %K=flipud(gray);%gray;
        %colormap(K(3:end-23,:))
        %colormap('JET')
        K=jet;
        %colormap(K(30:end,:))
        surf(x2, y2, z2, c2, 'LineStyle', 'none', 'FaceColor', 'interp'
            )
        shading interp
        light
        lighting phong%gouraud;%
        %material shiny

        caxis([-1 1])
        axis equal

        if i==1
```

```
            view(55,90)
            frac = normal_ves*1/2 + terminal_ves*0.9;
            text(x_off(1:9) - frac(1:9).*L(1:9).*sin(phi(1:9)) , y_off
                (1:9) + frac(1:9).*L(1:9).*cos(phi(1:9)), sqrt(res*max(
                A_0_s)/pi) + ones(size(parent_p(1:9))), num2str((1:9)'),
                'fontsize',7,'FontName','FixedWidth', 'fontweight','bold
                ', 'VerticalAlignment','middle', 'HorizontalAlignment','
                center');
            text(x_off(10:end) - frac(10:end).*L(10:end).*sin(phi(10:
                end)) , y_off(10:end) + frac(10:end).*L(10:end).*cos(phi
                (10:end)), sqrt(res*max(A_0_s)/pi) + ones(size(parent_p
                (10:end))), num2str((10:length(parent_p))'),'fontsize'
                ,7,'FontName','FixedWidth', 'fontweight','bold', '
                VerticalAlignment','middle', 'HorizontalAlignment','
                center');
            zoom(1.3)
            set(gcf, 'PaperUnits', 'centimeters');
            set(gcf, 'PaperPosition', [0 0 15 13]);
        else
            view(90,90)
            axis([-13 3.7  0 33 0 100])
            text(-11.5, 33/2-3.5, 0, ['$t=' num2str(t_stamp(i)) '$ s']
                ,'Interpreter','LaTeX','fontsize',font,'
                VerticalAlignment','middle');
            set(gcf, 'PaperUnits', 'centimeters');
            set(gcf, 'PaperPosition', [0 0 15 7]);
        end

        xlabel('x')
        ylabel('y')
        zlabel('z')
        axis off
        grid off

        set(gca,'position',[0 0 1 1]);
        if save == 1
            print(fullfile(figs_path,['network_55_virtual3d_' num2str(i
                )]),'-dpng', '-r500','-opengl')
        end
    end
    %%
    close(figure(6))
    figure(6)
    set(gca,'visible','off');
    caxis([-1 1])
    h = colorbar('SouthOutside');
    set(h, 'Position', [0.05 .45 0.9 .3])
    set(gcf, 'PaperUnits', 'centimeters');
    set(gcf, 'PaperPosition', [0 0 20 1.5]);
    if save == 1
        print(fullfile(figs_path,'network_55_virtual3d_colorbar'),'-
            dpng', '-r500','-opengl')
    end
end

%% virtual-3d netwerk test
```

```
close(figure(4))
res = 10;
stretch = 30;
%local collering

for i=1:length(parent_p)
    index = sum(cells(1:i-1))+1:sum(cells(1:i));
    A_max_ves(i) = max(max( U(   index , 1 , : ) ));
end

A_max = max(max(U(:,1,:)));
tic
for i=1:15:length(t) %163 for 1:1 secs
    x2 = [];
    y2 = [];
    z2 = [];
    c2 = [];
    for j=1:length(parent_p)
        x_i = sum(cells(1:j-1))+1:sum(cells(1:j));

        [x1, y1, z1, c1] = makeVessel(U(x_i, 1, i), A_0(x_i), beta(x_i)
            , res, phi(j), dx, stretch);
        x2 = [x2; nan(1,res+1); x1 + x_off(j)];
        y2 = [y2; nan(1,res+1); y1 + y_off(j)];
        z2 = [z2; nan(1,res+1); z1 + z_off(j)];
        c2 = [c2; nan(1,res+1); c1 / p(A_max_ves(j), sum(cells(1:j-1))
            +1 )]; %new
    end


    %plotting
    if i==1
        figure(4)
        pause(0.00001);
        frame_h = get(handle(gcf),'JavaFrame');
        set(frame_h,'Maximized',1);
        pl = surf(x2, y2, z2, c2, 'LineStyle', 'none', 'FaceColor', '
            interp');
        view(90, 90);

        colormap('JET')%summer
        shading interp
        light
        lighting gouraud;%phong
        %material shiny


        axis equal

        caxis([-1 1])%[-0.1*10^4 0.8*10^4]
        zoom(3)
        text(x_off - 1/2*L.*sin(phi) , y_off + 1/2*L.*cos(phi), sqrt(
            A_0_s/pi) + ones(size(parent_p)), num2str((1:length(parent_p
            ))') );

        %xlabel('x')
```

```
        %ylabel('y')
        %zlabel('z')
        axis off
        grid off
        colorbar
    end


    set(pl, 'XData', x2, 'YData', y2, 'ZData', z2 , 'CData', c2);
    %text(min(xlim), mean(ylim),['t = ' num2str(t(i)) ', i = ' num2str(
       i)])
    title(['t = ' num2str(t(i)) ', i = ' num2str(i)],'Position',[min(
       xlim)+80 min(ylim)+10 1])
    %hold off
    drawnow

end
toc
%% p and/or u
if 1==1
    q=4 ;%1 2 3 4

    figure(10)


    x_pos2 = reshape(bsxfun(@times, cumsum(L), ones(3,size(L))),1,[]);
    y_pos = repmat([p_min p_max nan], 1,length(L));
    y_pos2 = repmat([u_min u_max nan], 1,length(L));

    subplot(q,1,1)
    %adj -> parent_p
%     parent = [0];
%     for i=1:length(A_0_s)
%         parent = [parent i*ones(1,sum(ADJ(i,:)))];
%     end
    treeplot(parent_p)
    [x_c,y_c] = treelayout(parent_p);
    text(x_c,y_c, num2str((1:length(x_c))'));



    for i=1:5:length(t) %10
        if q >= 2
            subplot(q,1,2)
            %plot(x_pos2,y_pos,':k', x_pos,A_0,'r', x_pos,squeeze(U
               (:,1,i)),'b')
            plot(x_pos2,y_pos,':k', x_pos,p(squeeze(U(:,1,i)),1:N),'b')
            text(cumsum(L)-L,((p_max-p_min)*0.9+p_min)*ones(1,length(L)
               ), num2str((1:length(x_c))') );
            xlabel('x')
            ylabel('p')
            title(['t = ' num2str(t(i)) ', i = ' num2str(i)])
            axis([0 sum(L) p_min p_max])
        end
        if q >= 3
            subplot(q,1,3)
```

```
            plot(x_pos2,y_pos2,':k', x_pos,squeeze(U(:,2,i)))
            text(cumsum(L)-L/2,((u_max-u_min)*0.9+u_min)*ones(1,length(
               L)), num2str((1:length(x_c))') );
            xlabel('x')
            ylabel('u')
            title(['t = ' num2str(t(i)) ', i = ' num2str(i)])
            axis([0 sum(L) u_min u_max])
        end
        if q >= 4
            subplot(q,1,4)
            plot(x_pos2,y_pos2,':k', x_pos,squeeze(U(:,1,i)))
            text(cumsum(L)-L/2,((A_max-A_min)*0.9+A_min)*ones(1,length(
               L)), num2str((1:length(x_c))') );
            xlabel('x')
            ylabel('A')
            title(['t = ' num2str(t(i)) ', i = ' num2str(i)])
            axis([0 sum(L) A_min A_max])
        end
        drawnow
        %pause(100/1000)
    end
end
```