UNIVERSITY OF CALIFORNIA
RIVERSIDE

High Order Numerical Methods for Hyperbolic Balance Laws: Well-Balanced Discontinuous Galerkin Methods and Adjoint-Based Inverse Algorithms

A Dissertation submitted in partial satisfaction
of the requirements for the degree of

Doctor of Philosophy

in

Mathematics

by

Jolene A. Britton

June 2020

Dissertation Committee:

Dr. Yulong Xing, Chairperson
Dr. Weitao Chen
Dr. Mark Alber

The Dissertation of Jolene A. Britton is approved:

Committee Chairperson

University of California, Riverside

Acknowledgments

First, I would like to extend great thanks to my dissertation committee, Dr. Yulong Xing from The Ohio State University, as well as Dr. Weitao Chen and Dr. Mark Alber from the University of California, Riverside. Thank you Dr. Xing for your willingness to meet so often, for your insightful guidance on projects, for the opportunities of networking you provided, and for your support and encouragement in my development as a mathematician. I am honored to have you as my advisor. Thank you Dr. Chen and Dr. Alber for your support and career advice. I would also like to thank Dr. Yat Tin Chow for all of your invaluable help with research. I am so grateful for the many, many hours you sat with me and for the many encouragements you shared.

Second, I would like to thank the many mathematical mentors I had outside of UCR. Thank you Dr. Matthew Reynolds at the National Renewable Energy Laboratory for selecting me as an intern and giving me the opportunity to work with you for two summers. I am eternally grateful for the many opportunities you provided me with. I would like to thank Dr. Jason Wilson and all the mathematics faculty at my undergraduate institution, Biola University, for encouraging me to apply to graduate school and for believing in me.

Third, I would like to acknowledge my colleagues. To my office mates and fellow applied math graduate students Josh Buli, Mikhal Banwarth-Kuhn, Kevin Tsai, Daniel Collister, Christian Michael, Cecelia Duran, Christina Knox, Jenifer Rangel Ambriz, and Alysha Toomey. It was a great honor being your colleague and I am thankful I could spend my graduate career with each of you. I would also like to thank the UCR faculty and the department chair Dr. Yat Sun Poon for generous travel and fellowship funding. Many

thanks to the many UCR math administrative staff. Your assistance and continual guidance is very appreciated.

Finally, I would like to thank my family for their support throughout graduate school. Sam, you are the best study partner and source of unconditional support. Additionally, I could not have made it this far without the rest of my family: Janet, Keith, Daniel, Tyler, Becky, Cameron, Frank, Sue, Joel, Kate, Hannah, Eli, Seth, and Charlie.

The text of this dissertation, in part or in full, is a reprint of the material as it appears in [16] (Jolene Britton and Yulong Xing. "High Order Still-Water and Moving-Water Equilibria Preserving Discontinuous Galerkin Methods for the Ripa Model". In: *Journal of Scientific Computing* 82.2 (Feb. 2020)) and [17] (Jolene Britton and Yulong Xing. "Well-Balanced Discontinuous Galerkin Methods for the One-Dimensional Blood Flow through Arteries Model with Man-at-Eternal-Rest and Living-Man Equilibria". In: *Computers & Fluids* 203 (2020), p. 104493). The co-author, Yulong Xing, directed and supervised the research which forms the basis for this dissertation as well as provided ideas, technical expertise, and aid in the preparation of the listed manuscripts.

To Sam and my family.

ABSTRACT OF THE DISSERTATION

High Order Numerical Methods for Hyperbolic Balance Laws: Well-Balanced Discontinuous Galerkin Methods and Adjoint-Based Inverse Algorithms

by

Jolene A. Britton

Doctor of Philosophy, Graduate Program in Mathematics
University of California, Riverside, June 2020
Dr. Yulong Xing, Chairperson

Hyperbolic balance laws are a special class of partial differential equations that represent various physical phenomena. Numerous scientific and engineering problems can be modeled using balance laws. This thesis contains three parts dedicated to the advancement of numerical methods for hyperbolic balance laws.

In the first and second parts, we construct high-order accurate well-balanced discontinuous Galerkin methods for two different systems of hyperbolic balance laws. The first part is devoted to the Ripa model and the second to the arterial blood flow model. The schemes were developed to preserve zero velocity and non-zero velocity steady states. The methods are an extension of the schemes used for the shallow water equations. Special attention is paid to the projection of the initial conditions into piecewise polynomial space, the approximation of the source term, and the construction of the numerical fluxes. High order and well-balanced methods have been previously developed for zero velocity steady states, however little work has been done on the more general non-zero velocity steady states. Numerical examples are given to demonstrate the well-balanced property, accuracy,

non-oscillatory behavior at discontinuities, and ability to resolve small perturbations to steady states. This approach can be generalized to other balance laws.

In the third part, we develop an adjoint approach for recovering the topographical function included in the source term of one dimensional hyperbolic balance laws. We focus on a specific system, namely the shallow water equations, in an effort to recover the riverbed topography. The novelty of this work is the ability to robustly recover the bottom topography using only noisy boundary data from one measurement event and the inclusion of two regularization terms in the iterative update scheme. The adjoint scheme is determined from a linearization of the forward system and is used to compute the gradient of a cost function. The bottom topography function is recovered through an iterative process given by a three-operator splitting method which allows the feasibility to include two regularization terms. Numerous numerical tests demonstrate the robustness of the method regardless of the choice of initial guess and in the presence of discontinuities in the solution of the forward problem.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Hyperbolic Balance Laws

Hyperbolic conservation laws are a class of time-dependent partial differential equations (PDEs) that have a simple structure and are used to model various physical phenomena. A one-dimensional scalar conservation law can be expressed by the following notation,

$$\frac{\partial}{\partial t}u(x,t) + \frac{\partial}{\partial x}f(u(x,t)) = 0. \tag{1.1}$$

The variable u is the *state variable* or *conserved variable*. It represents a conserved quantity, common examples include mass, momentum, and energy. A state variable is conserved if the total quantity in a fixed spatial domain remains constant over time. In one dimension, the total quantity of u in the interval $[x_0, x_L]$ at time t is represented by the integral $\int_{x_0}^{x_L} u(x,t)\ dx$. The quantity of u can neither be created nor destroyed in an isolated system, the only manner in which the quantity of u can vary is due to flow across the

spatial boundaries x_0 and x_L. This is represented by the flux term $f(u)$, thus the rate of change in the total quantity of u with respect to time is

$$\frac{d}{dt}\int_{x_0}^{x_L} u(x,t)\ dx = f(u(x_0,t)) - f(u(x_L,t)). \tag{1.2}$$

Integrating in time from t_1 to t_2 and rearranging terms results in the equivalent formulation

$$\int_{x_0}^{x_L} \left[u(x,t_2) - u(x,t_1)\right]\ dx = \int_{t_1}^{t_2} \left[f(u(x_0,t)) - f(u(x_L,t))\right]\ dt. \tag{1.3}$$

Under the assumption that $u(x,t)$ is differentiable in both space and time, (1.3) is equivalent to

$$\int_{t_1}^{t_2}\int_{x_0}^{x_L} \left[\frac{\partial}{\partial t}u(x,t) + \frac{\partial}{\partial x}f(u(x,t))\right]\ dxdt = 0. \tag{1.4}$$

Lastly, the integrand itself must be zero since this integral equation holds for all values of x_0, x_L and t_1, t_2, thus the conservation law (1.1) is satisfied.

More generally, one can express a system of m conservation laws in arbitrary dimensions as follows,

$$\partial_t \boldsymbol{U} + \nabla \cdot \boldsymbol{F}(\boldsymbol{U}) = 0, \tag{1.5}$$

where $\boldsymbol{U} = [\boldsymbol{U}_1, \boldsymbol{U}_2, ..., \boldsymbol{U}_m]^T$ is a vector and $\boldsymbol{F}$ is a vector-valued function. We will only examine one-dimensional systems throughout this thesis, therefore we will only use one-dimensional notation from this point on.

Conservation laws are representative of physical properties that do not change over time in isolated systems. However, if the system is not isolated and sources or sinks are present, the system (1.1) is no longer valid because the right hand side becomes non-zero. A system with a non-zero right hand side representing a source or a sink in the system is referred to as a conservation law with a source term, or a balance law. The formulation for

a system of m balance laws takes the form

$$\partial_t \boldsymbol{U} + \boldsymbol{F}(\boldsymbol{U})_x = \boldsymbol{S}(\boldsymbol{U}, x). \tag{1.6}$$

A source term may arise as a result of a friction component, a change in topography, chemical reactions, atomic excitations, etc.

The flux terms in the formulations (1.1) and (1.6) are convective. It is also possible for the flux of a conservation or balance law to be diffusive [93]. However, for problems modeling compressible flows, one can often ignore diffusion. Conservation and balance laws with purely convective fluxes are known as hyperbolic whereas a conservation law written as an advection-diffusion equation is parabolic [93]. The hyperbolicity of (1.6) can be more clearly realized by rewriting the system in the quasi-linear form

$$\partial_t \boldsymbol{U} + \mathcal{J}(\boldsymbol{U})\partial_x \boldsymbol{U} = \boldsymbol{S}(\boldsymbol{U}). \tag{1.7}$$

In one dimension, $\mathcal{J}(\boldsymbol{U}) = \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{U}}$ is the $m \times m$ Jacobian matrix of the flux function. The system is called *hyperbolic* if $\mathcal{J}$ has all real-valued eigenvalues and the matrix is diagonalizable, implying m linearly independent eigenvectors. This property allows one to decompose the Jacobian

$$\mathcal{J} = R\Lambda R^{-1}, \tag{1.8}$$

where $\Lambda = \text{diag}\,(\lambda_1, \lambda_2, ..., \lambda_m)$ is a matrix with the eigenvalues along the diagonal in increasing order and $R = [r_1|r_2|...|r_m]$ is a matrix of the corresponding right eigenvectors. Furthermore, when the eigenvalues are all distinct, the system is called *strictly hyperbolic*.

Solutions of hyperbolic PDEs have a wave-like behavior. A disturbance in the solution propagates with finite speed along characteristics, as opposed to a disturbance impacting the solution at every point instantaneously. This is a property unique to hyperbolic

PDEs and does not occur with elliptic and parabolic PDEs. Another distinctive property of hyperbolic PDEs is the emergence of discontinuities in the solution even with smooth initial conditions. These unique properties exhibited by hyperbolic PDEs have led to numerous numerical challenges. Such challenges have catalyzed the development of specialized numerical schemes specifically designed for hyperbolic PDEs. These challenges and numerical schemes will be further discussed in Section 1.2.

Balance laws are important to study for many reasons. LeVeque [93] suggests three reasons. First, there are unique difficulties, such as shock formations, for conservation and balance laws, that require careful attention when designing numerical methods. Second, the theory and mathematical 'structure' is well understood. Since few exact solutions are known analytically, the theoretical knowledge can be used to when designing numerical schemes to help address the numerical difficulties. Lastly, many science and engineering problems can be modeled using balance and conservation laws. Some common examples include the Euler equations and shock tube problem of gas dynamics, traffic flow, astrophysical modeling, the Buckley-Leveret equation for modeling saturation of water in rocks and sand, flow of glacier, and separation of chemical species by chromatography. We will more thoroughly study two groups of hyperbolic balance law models in this thesis: the shallow water equations (and related systems) and the arterial blood flow model.

1.1.1 The Shallow Water Equations

The one-dimensional shallow water equations (SWEs) were originally derived in 1871 by Adhémar Jean Claude Barré de Saint-Venant [3], and thus often referred to as the

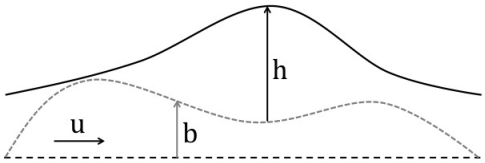


Figure 1.1: Visualization of the water height (h), velocity (u), and bottom topography (b).

Saint-Venant equations. The SWEs in one dimension take the form

$$\begin{cases} \partial_t h + \partial_x(hu) = 0, \\ \partial_t(hu) + \partial_x(hu^2 + \frac{1}{2}gh^2) = -gh\partial_x b, \end{cases} \tag{1.9}$$

where $h(x,t) \geq 0$ represents the height of the water, $u(x,t) \in \mathbb{R}$ describes the velocity, $b(x)$ represents the bottom topography, and $g = 9.812$ is the gravitational constant. The term hu represents water discharge. Figure 1.1 is a representation of the system's state variables. The source term of (1.9), used in this thesis, is due only to the bottom topography function, however it may also include a friction term.

The corresponding state variables, flux terms, and source term are

$$\boldsymbol{U} = \begin{bmatrix} h \\ hu \end{bmatrix}, \qquad \boldsymbol{F}(\boldsymbol{U}) = \begin{bmatrix} hu \\ hu^2 + \frac{1}{2}gh^2 \end{bmatrix}, \qquad \boldsymbol{S}(\boldsymbol{U}, b) = \begin{bmatrix} 0 \\ -gh\partial_x b \end{bmatrix}. \tag{1.10}$$

Thus, the Jacobian matrix can be determined

$$\mathcal{J}(\boldsymbol{U}) = \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{U}} = \begin{bmatrix} 0 & 1 \\ gh - u^2 & 2u \end{bmatrix}, \tag{1.11}$$

which has eigenvalues $\lambda_1 = u - \sqrt{gh}$ and $\lambda_2 = u + \sqrt{gh}$. Since λ_1 and λ_2 are real-valued, we see that (1.9) is hyperbolic

There are at least two schools of thought in regards to motivating the derivation of the SWEs. Adhémar Jean Claude Barré de Saint-Venant [3] first derived the one-dimensional system by considering Newton's first and second laws with mass and momentum conservation. A more common method, recently, involves deriving the system by depth-averaging the Navier-Stokes equations along with the assumption that the horizontal length scale is far greater than the vertical.

The SWEs are used in atmospheric modeling and have been exceptionally beneficial to the tsunami modeling community [6, 5]. Hydro-dynamical issues are well suited for the SWEs, such as simulating storm surges [45], river flooding and dam break problems [112]. Furthermore the SWEs can be used to model avalanches and landslides [67]. This model will be the focus of Chapter 4.

There are numerous extensions of the SWEs. One of them is the shallow water equations with temperature fluctuations. This system was introduced by Pedro Ripa in 1993 [43, 128, 129] and is often referred to as the Ripa system. The system was introduced for the purpose of modeling ocean currents. The introduction of temperature is advantageous because the movement and behavior of ocean currents are impacted by forces such as temperature acting upon the water.

The one-dimensional Ripa equations take the form

$$\begin{cases} \partial_t h + \partial_x(hu) = 0, \\ \partial_t(hu) + \partial_x(hu^2 + \frac{1}{2}gh^2\theta) = -gh\theta\partial_x b, \\ \partial_t(h\theta) + \partial_x(h\theta u) = 0, \end{cases} \tag{1.12}$$

where $\theta(x,t) > 0$ is a potential temperature field and $gh^2\theta/2$ is the pressure depending

on the water temperature. The potential temperature field θ is defined to be the reduced gravity $g\Delta\Theta/\Theta_{\text{ref}}$ [32], where $\Delta\Theta$ is set to be the difference in potential temperature from a reference value Θ_{ref}. Notice that, when the potential temperature field $\theta = 1$ in the Ripa model (1.12), the SWEs (2.2) are recovered. This model will be further explored in Chapter 2.

Other common systems from the shallow water family are the two-layer shallow water systems and the shallow water equations with varying channel widths. Additionally, the model can be modified to include a time-dependent bottom topography function. Many other extensions and modifications to the original SWEs exist that serve varying purposes.

1.1.2 Blood Flow in Arteries Model

Blood flow models have been extensively used to mathematically understand and numerically simulate the human cardiovascular system. In 1775, Euler [48] derived a one-dimensional model of the human arterial system from the conservation of the mass and momentum of the flow. Without the understanding of the wave-like nature of the flow, he noticed that the problem was too difficult to solve. Young [162] was the first to identify blood flow with wave-like behavior by finding analogous behavior between arterial blood flow wave speed and Newton's sound speed in air theories. Blood movement in arteries have flow with periodic variations known as pulsatile flow, which has been understood and explained by Lighthill [98] and Pedley [114]. Nowadays, three-dimensional mathematical models for the blood flow in arteries already exist, but the simpler one-dimensional models with averaged quantities are still of great importance [54, 136, 137]. The low computational cost of a one-dimensional model as compared to higher dimensional models allows for one

to study the wave effects within isolated segments of an artery or within the entire vascular system (i.e in the aorta and systemic arteries) [111, 119, 118, 135]. Another usage of one-dimensional models is the ability to study the effects of arterial modifications, such as placements of stents and prostheses, on pulse waves [22, 54]. Lastly, one-dimensional models can also be easily coupled with lumped parameter models [122] and three-dimensional fluid-structure models [52, 53]. A systematic comparison of computational hemodynamics in arteries between one-dimensional and three-dimensional models with deformable vessel walls was carried out in [149], where they observed good agreement between the two models, especially during the diastolic phase of the cycle.

The one-dimensional partial differential equation (PDE) model for the blood flow through arteries [135, 54, 105] takes the form,

$$\begin{cases} \partial_t A + \partial_x Q = 0, \\ \partial_t Q + \partial_x \left(\alpha \dfrac{Q^2}{A} \right) + \dfrac{A}{\rho} \partial_x p = 0, \end{cases} \tag{1.13}$$

where $A(x,t) = \pi R^2(x,t)$ is the cross-sectional area with $R(x,t) > 0$ being the radius. The variable $Q(x,t) = A(x,t)u(x,t)$ is the discharge, $u(x,t)$ denotes the flow velocity, and the constant ρ is the blood density. The parameter α is the momentum-flux correction coefficient that depends on the assumed velocity profile, and in this chapter, we take $\alpha = 1$, which means a blunt velocity profile. The source terms representing the viscous resistance of the flow and gravitational effect could be added to the system. We refer to [55] for the full description of this model.

To close the system, one needs an additional equation to link the pressure with the displacement of the vessel. A simple law describing the elastic behavior of the arterial

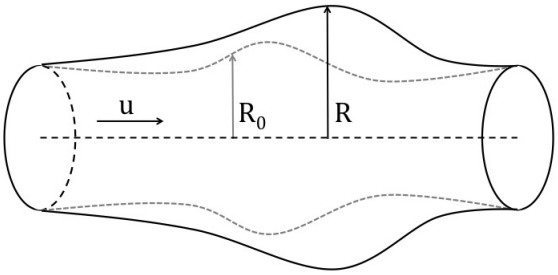


Figure 1.2: Diagram of the one-dimensional blood flow model with the cross-sectional radius at rest (R_0), cross-sectional radius (R), and velocity (u).

wall is given by

$$p = p_{ext} + K(R - R_0), \qquad \text{or equivalently,} \quad p = p_{ext} + \frac{K}{\sqrt{\pi}}\left(\sqrt{A} - \sqrt{A_0}\right), \tag{1.14}$$

where p_{ext} stands for the external pressure (assumed to be constant), the constant K represents arterial stiffness, and $A_0(x) = \pi R_0^2(x)$ is the cross-section at rest (when $u = 0$) with $R_0(x)$ being its radius. Other complex nonlinear relationship could be introduced as well. With the simple elastic law (1.14), the one-dimensional model (1.13) for the blood flow through arteries can be rewritten in the form of hyperbolic balance laws

$$\begin{cases} \partial_t A + \partial_x Q = 0, \\ \partial_t Q + \partial_x \left(\dfrac{Q^2}{A} + \dfrac{K}{3\rho\sqrt{\pi}} A^{\frac{3}{2}} \right) = \dfrac{KA}{2\rho\sqrt{\pi}\sqrt{A_0}} \partial_x (A_0), \end{cases} \tag{1.15}$$

which will be studied in this chapter. A stent or other physical variations may cause a non-constant cross-sectional at rest $A_0(x)$, introducing a non-zero source term. A diagram of the one-dimensional blood flow model with the cross-sectional radius at rest (R_0), cross-sectional radius (R), and velocity (u) is presented in Figure 1.2. For simplicity, we denote $\beta = \frac{K}{\rho\sqrt{\pi}}$ in the rest of the chapter.

The PDE model (1.15) for the blood flow through arteries can be written in the convenient hyperbolic balance laws notation

$$\partial_t \boldsymbol{U} + \partial_x \boldsymbol{F}(\boldsymbol{U}) = \boldsymbol{S}(\boldsymbol{U}, A_0),$$

where

$$\boldsymbol{U} = \begin{bmatrix} A \\ Q \end{bmatrix}, \quad \boldsymbol{F}(\boldsymbol{U}) = \begin{bmatrix} Q \\ \frac{Q^2}{A} + \frac{\beta}{3} A^{\frac{3}{2}} \end{bmatrix}, \quad \boldsymbol{S}(\boldsymbol{U}, A_0) = \begin{bmatrix} 0 \\ \frac{\beta A}{2\sqrt{A_0}} \partial_x (A_0) \end{bmatrix},$$

are the state variables, the flux, and the source term, respectively. The Jacobian matrix $\mathcal{J}(\boldsymbol{U})$ is given by

$$\mathcal{J}(\boldsymbol{U}) = \begin{bmatrix} 0 & 1 \\ c^2 - \left(\frac{Q}{A}\right)^2 & \frac{2Q}{A} \end{bmatrix},$$

with $c = \sqrt{\frac{\beta\sqrt{A}}{2}}$. The eigenvalues of the Jacobian matrix are $\frac{Q}{A} \pm c$, which are real-valued [104]. This implies that the system is hyperbolic. This system with be the focus of Chapter 3.

1.2 Mathematical & Numerical Challenges of Hyperbolic Balance Laws

In this section, we summarize the mathematical and numerical challenges associated with solving balance laws. LeVeque discusses such topics in great detail, see [93] for further information. The first mathematical problem that arises with balance laws is the possibility of non-unique solutions. Since such systems often represent physical phenomena, only one solution can be physically accurate. One method for addressing this concern is to introduce a diffusive term and let the coefficient of the diffusive term go to zero. However,

imposing extra terms makes the system more complicated and is thus not the optimal approach. Instead, appealing to entropy conditions and weak solutions is typically a favored method for determining the unique and physically relevant solution.

A second distinctive mathematical challenge of hyperbolic balance laws is the emergence of discontinuous solutions. Even with smooth initial data and flat bottom topography functions (that is, no source term due to topography), systems may develop discontinuities. In the classical sense, discontinuous solutions cannot satisfy the system of PDEs. However, when a weak or integral form of the PDEs are used, discontinuous solutions are valid. One downside of the use of integral forms is that weak formulations may be more challenging to work with when it comes to the discretization in the numerical schemes.

Numerical challenges arise in the treatment of discontinuities. Smearing of discontinuities due to numerical viscosity of some schemes, such as the first order Godunov method, is one issue. On the other hand, dispersive effects of second order methods, like the Mac-Cormack scheme, may lead to oscillations near the discontinuity. One approach for addressing the numerical challenges of keeping discontinuities sharp is known as shock tracking where jump conditions are used to supplement the numerical scheme across the discontinuities. This method can become too complicated in higher dimensions. For second and higher order accurate methods, an alternative that does not involve explicit use of jump conditions is known as shock capturing. Additionally, such a method does not result in smearing or spurious oscillations. Slope limiters are also used to eliminate oscillations near discontinuities.

The preservation of steady states or flows in near equilibrium is a task that can

be surprisingly difficult to achieve numerically. A method that can preserve a steady state must perfectly balance the flux and source terms and is referred to as a well-balanced method. Well-balanced schemes are advantageous because they are capable or resolving small perturbations to steady state solutions, even on a coarse mesh. Often, the error that arises in a non-well-balanced scheme can be larger than the magnitude of the perturbation to the steady state. However a scheme that satisfies the well-balanced property can aptly handle such perturbations and achieve high resolution solutions without computationally expensive mesh refinement [85]. Numerous well-balanced methods have been developed for balance laws in [91, 82, 86, 57, 131, 96, 109, 26, 101, 110, 133, 51, 9, 31]. Well-balanced methods will be further discussed in Chapters 2 and 3.

The need to accurately model bounded or non-negative variables can be a numerical challenge in regards to conservation laws. Examples include modeling dry fronts with the SWEs or preserving the positivity of density for traffic flow models. Numerical schemes may not be able to maintain the positivity of water height or car density, thus positivity-preserving limiters have been developed to address such challenges [115, 116, 1, 142, 56, 87, 10, 127, 164, 165, 160, 13, 140, 12, 126, 7, 100].

Lastly, one must take extra care when implementing various limiters and trying to preserve the well-balanced property simultaneously. In some cases, a slope limiter and positivity-preserving limiter do not work well together because the time step sizes become so small that the scheme comes to a halt [20, 47]. On the other hand, in some cases the total variation bounded (TVB) [138, 36] slope limiter and positivity-preserving limiter work well together, but do not preserve the well-balanced property [160]. Thus, one has to make

adjustments [160] so that the numerical limiters are not in conflict.

1.3 Review of Numerical Methods for Hyperbolic Equations

Solutions of hyperbolic balance laws are often understood in a weak or integral sense so that discontinuous solutions are well-defined. Schemes based on the weak formulation, such as finite-volume and discontinuous Galerkin (DG) methods, are appropriate frameworks for complicated and discontinuous solutions [85].

1.3.1 Finite Volume Methods

There are three main methods for computing finite volume solutions: upwind methods, central methods, and central-upwind methods. The first finite volume upwind scheme is due to Godunov in 1959 [63]. The main idea of the method involves discretizing the domain into cells, approximating the solution in each cell by a constant, and evaluating the flux integrals which requires one to solve the Riemann problem. While the method is advantageous for stabilizing computation of flux integral, the requirement of solving the Riemann problem is complicated and computationally expensive [85] for some models. Central schemes, in contrast to upwind schemes, do not require a Riemann solver and are often more simple. The method is achieved by carefully selecting the control volume so that the discontinuities of the piecewise constant approximations of the solution do not line up with the cell boundaries. The first order Lax-Friedrichs scheme [92] and second order scheme due to Nessyahu and Tadmor [107] are common choices. A shortcoming of the central schemes is the high amount of numerical dissipation. Such diffusion is particularly

noticeable when the time step size is small or when a steady state for a long-term temporal period is computed [85]. The third choice is the central-upwind method. This method uses the local speeds of propagation to determine non-uniform control volumes. This method is at least second order accurate and does not suffer from dissipation or require one to solve the Riemann problem.

1.3.2 Discontinuous Galerkin Methods

The DG method is another solver that has gained much attention in the last few decades. This is the numerical solver used in all numerical examples for this thesis. The DG methods can be viewed as a generalized version of the finite volume methods. The DG method takes key concepts from the finite volume framework (numerical fluxes, slope limiters, etc.) and inserts them into the finite element framework with the assumption that solutions are possibly discontinuous across the discretized cell interfaces [33]. There are many advantages that the finite element framework in the DG method provide in comparison to finite volume or even finite difference methods. Some advantages include the high order accuracy of the DG method (accuracy depends on the polynomial degree of the approximated solution), the parallelizable nature of the DG methods, DG's ability to handle complex geometries and adaptivity [33].

The DG method was first presented for the neutron transport equation in 1973 by Reed and Hill [124]. Early applications of this numerical method besides the neutron transport equation included ODEs [90], wave propagation by Oden and Wellford [147], and optimal control by Delfour and Trochu [42]. Later, the method was popularized when applied to nonlinear hyperbolic systems. However, finding an appropriate time discretiza-

tion proved to be a challenge. Cockburn and Shu were the first to introduce the Runge Kutta Discontinuous Galerkin (RKDG) method, which were stable, second-order accurate in smooth regions, kept discontinuities sharp without oscillations, and converged to the entropy solution [34]. The DG method has now been expanded to non-hyperbolic systems and many variations on the method are heavily used. A more in depth description of the development of the DG method can be found in [33].

There are many variations of the DG method available. We employ the modal DG methods in this thesis, as opposed to the nodal DG scheme. The derivation of the scheme in one dimension will now be described.

First, one determines the semi-discrete form. We must discretize the computational domain I into cells $I_j = [x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$, and denote the size of the j^{th} cell by Δx_j. Furthermore, we let $\tau = \max_j \Delta x_j$. We seek an approximation $\boldsymbol{U}_\tau$ which belongs to the finite dimensional DG space:

$$\mathbb{V}_\tau^k = \{v : w|_{I_j} \in P^k(I_j), j = 1, ..., J\}, \tag{1.16}$$

where $P^k(I)$ is the space of polynomials of degree up to k in the domain I, and J is the total number of cells. In addition, s_τ denotes the projection of the source parameter function s into $\mathbb{V}_\tau^k$. For example, in the case of the SWEs $s = b$, the bottom topography function, and in the case of the blood flow model $s = A_0$, the cross-sectional area at rest. We denote $\boldsymbol{U}^+_{\tau,j+\frac{1}{2}}$ and $\boldsymbol{U}^-_{\tau,j+\frac{1}{2}}$ as the limit values of $\boldsymbol{U}_\tau$ at the element interface $x_{j+\frac{1}{2}}$ from the right cell I_{j+1} and from the left cell I_j, respectively. The conventional DG scheme in each cell can be formulated as

$$\int_{I_j} \partial_t \boldsymbol{U}_\tau v \, dx - \int_{I_j} \boldsymbol{F}(\boldsymbol{U}_\tau) \partial_x v \, dx + \hat{\boldsymbol{F}}_{j+\frac{1}{2}} v^-_{j+\frac{1}{2}} - \hat{\boldsymbol{F}}_{j-\frac{1}{2}} v^+_{j-\frac{1}{2}} = \int_{I_j} \boldsymbol{S}(\boldsymbol{U}_\tau, s_\tau) v \, dx, \tag{1.17}$$

where $v(x)$ is a test function from the test space $\mathbb{V}_\tau^k$ and

$$\hat{\boldsymbol{F}}_{j+\frac{1}{2}} = \boldsymbol{f}\left(\boldsymbol{U}_\tau\left(x^-_{j+\frac{1}{2}}, t\right), \boldsymbol{U}_\tau\left(x^+_{j-\frac{1}{2}}, t\right)\right),$$

with $\boldsymbol{f}(a, b)$ being the numerical flux. It has been shown that for high order DG methods, the effect of different fluxes on the accuracy of methods is relatively small [120]. Therefore, in this chapter, we will employ the use of the simple Lax-Friedrichs flux:

$$\boldsymbol{f}(a, b) = \frac{1}{2}\left(\boldsymbol{F}(a) + \boldsymbol{F}(b) - \alpha(b - a)\right), \tag{1.18}$$

where α is the maximum of the largest eigenvalue and the maximum in the calculation of α is taken either globally (Lax-Friedrichs flux) or locally (local Lax-Friedrichs flux).

In an effort to convert the semi-discrete method (1.17) into a fully discrete form, we must discretize in time. The total variation diminishing (TVD) Runge-Kutta (also known as Strong-Stability Preserving Runge-Kutta) time discretization is used to increase temporal accuracy and stability [66]. The third order TVD Runge-Kutta method described as

$$\begin{aligned}
\boldsymbol{U}_\tau^{(1)} &= \boldsymbol{U}_\tau^n + \Delta t \mathcal{F}(\boldsymbol{U}_\tau^n), \\
\boldsymbol{U}_\tau^{(2)} &= \frac{3}{4}\boldsymbol{U}_\tau^n + \frac{1}{4}\left(\boldsymbol{U}_\tau^{(1)} + \Delta t \mathcal{F}\left(\boldsymbol{U}_\tau^{(1)}\right)\right), \\
\boldsymbol{U}_\tau^{n+1} &= \frac{1}{3}\boldsymbol{U}_\tau^n + \frac{2}{3}\left(\boldsymbol{U}_\tau^{(2)} + \Delta t \mathcal{F}\left(\boldsymbol{U}_\tau^{(2)}\right)\right),
\end{aligned} \tag{1.19}$$

is used throughout this thesis, where $\mathcal{F}$ is the spatial operator.

1.4 Organization of Thesis

This thesis is organized as follows. Well-balanced methods for the Ripa model are developed in Chapter 2. Sections 2.2 and 2.3 are dedicated to still-water steady states

while Sections 2.4 and 2.5 are dedicated to moving-water steady states. In Chapter 3, well-balanced methods for the one-dimensional blood flow through arteries model is discussed. The man-at-eternal-rest steady states both with zero and non-zero pressure are the focus of Sections 3.2 and 3.3 will the preservation of the living-man steady state is the focus of Sections 3.4 and 3.5. A numerical algorithm targeting an inverse problem for the SWEs is presented in Chapter 4. The aim of the chapter is to recover the bottom topography function from the boundary data in one measurement event. The inverse problem is achieved via an adjoint approach and a three-operator splitting scheme to iteratively update the solution. Lastly, concluding remarks and future work is discussed in Chapter 5.

Chapter 2

High Order Still-Water and Moving-Water Equilibria Preserving Discontinuous Galerkin Methods for the Ripa Model

2.1 Introduction

The one-dimensional Ripa equations, introduced in Section 1.1.1, take the form:

$$\begin{cases} \partial_t h + \partial_x(hu) = 0, \\ \partial_t(hu) + \partial_x(hu^2 + \frac{1}{2}gh^2\theta) = -gh\theta\partial_x b, \\ \partial_t(h\theta) + \partial_x(h\theta u) = 0, \end{cases} \tag{2.1}$$

where $h(x,t) \geq 0$ represents the height of the water, $u(x,t) \in \mathbb{R}$ describes the velocity, $\theta(x,t) > 0$ is a potential temperature field, $b(x)$ represents the bottom topography, and g is the gravitational constant. The term hu represents water discharge and $gh^2\theta/2$ is the pressure depending on the water temperature. The potential temperature field θ is defined to be the reduced gravity $g\Delta\Theta/\Theta_{\text{ref}}$ [32], where $\Delta\Theta$ is set to be the difference in potential temperature from a reference value Θ_{ref}. Additional source terms that model the friction along the bottom and surface or variations in the width of the channel could be included. In this thesis, we only consider the source term that accounts for the bottom topography.

The Ripa system is a generalized model of the SWEs. The one dimensional SWEs take the form:

$$\begin{cases} \partial_t h + \partial_x(hu) = 0, \\ \partial_t(hu) + \partial_x(hu^2 + \frac{1}{2}gh^2) = -gh\partial_x b. \end{cases} \tag{2.2}$$

Notice that, when the potential temperature field $\theta = 1$ in the Ripa model (2.1), the SWEs (2.2) are recovered. The SWEs consist of the conservation of mass and momentum, with the assumption that the density is constant. Two-layer and multi-layer shallow water equations [15] have been studied to model the flows in the shallow water regime where several layers with different densities appear. Such a model assumes a piecewise constant density inside each layer, and allows for different densities across the layers. Challenges in studying such models arise from the complicated eigenstructure, non-conservative terms, and conditional hyperbolicity, etc. Despite these challenges, there have been many studies on various numerical methods for multi-layer shallow water equations. The Ripa model can be obtained by vertically averaging each variable (including the density) over all layers from the bottom to the top, therefore, we lose the information of the interface between

layers, but the resulting model has a simpler eigenstructure and is always hyperbolic in the conservative form. The horizontal temperature gradients are introduced in the Ripa model to represent the variations in the fluid density.

Both the Ripa system (2.1) and the SWEs (2.2) belong to the family of hyperbolic balance laws, which have gained growing attention in the last few decades. In the one dimensional setting, such models usually take the form of

$$\partial_t \boldsymbol{U} + \partial_x \boldsymbol{F}(\boldsymbol{U}) = \boldsymbol{S}(\boldsymbol{U}).$$

Due to the existence of the source term $\boldsymbol{S}(\boldsymbol{U})$, hyperbolic balance laws introduce new computational challenges beyond the existing challenges of hyperbolic conservation laws. Balance laws often admit non-trivial steady state solutions in which the source term balances the effect of the flux gradients. The balance of fluxes and the source term as well as small perturbations of steady state solutions cannot be captured well by standard numerical methods with a straightforward implementation of the source term, unless a much refined mesh is used. Therefore, the well-balanced methods, which can exactly preserve steady state solution at the discrete level, are introduced to provide an accurate solution on a relatively coarse mesh and resolve small perturbations to steady state solutions accurately.

The steady state solutions of the Ripa model (2.1) occur when $\partial_t \boldsymbol{U} = 0$, that is

$$\begin{cases} \partial_x(hu) = 0, \\ \partial_x\left(hu^2 + \frac{1}{2}gh^2\theta\right) = -gh\theta\partial_x b, \\ \partial_x(h\theta u) = 0. \end{cases} \tag{2.3}$$

In the case of still-water, when the velocity u is zero, the steady state system (2.3) reduces

to

$$\begin{cases} u = 0 \\ \partial_x\left(\frac{1}{2}h^2\theta\right) = -h\theta\partial_x b, \end{cases} \tag{2.4}$$

which is an underdetermined PDE system. In order to reach a solution for (2.4), additional assumptions for h, θ or b must be enforced. There are three cases to consider if we require one of these variables to be constant. This raises one discrepancy between the Ripa model and the SWEs. The first case is the *still-water* steady state, which corresponds to a flat water surface under constant temperature:

$$\big(u, \theta, h + b\big) = \big(0, C_1, C_2\big), \tag{2.5}$$

where C_1, C_2 are constants. This is the same lake-at-rest steady state solution of the SWEs. The second case is the *isobaric* steady state, which corresponds to a wave in which the height and temperature jump but velocity and pressure remain constant:

$$\big(u, b, h^2\theta\big) = \big(0, C_1, C_2\big). \tag{2.6}$$

The last case is the *constant water height* steady state:

$$\Big(u, h, b + \frac{1}{2}h\ln\theta\Big) = \big(0, C_1, C_2\big). \tag{2.7}$$

The more general case occurs when the velocity u does not vanish. The *moving-water* equilibrium is given by:

$$\begin{cases} hu = constant, \\ \theta = constant, \\ \frac{u^2}{2} + g\theta(h + b) = constant, \end{cases} \tag{2.8}$$

where the momentum hu and potential temperature field θ are constant. It is easy to observe that the lake-at-rest still-water steady state (2.5) is simply a special case of the moving-water steady state (2.8). Well-balanced methods for the moving-water equilibrium can automatically preserve the lake-at-rest steady state, but not vice versa.

Well-balanced numerical methods for SWEs are far more studied in the literature, and can serve as a foundation of well-balanced methods for Ripa models. A vast amount of well-balanced methods for the still-water steady state [8, 91, 86, 1, 152, 127] have been studied, and we refer to the survey papers [158, 85] for a complete list of existing literature on this topic. Well-balanced methods for the moving-water equilibrium are more complicated and it is much more difficult to design such methods. Comparison of well-balanced methods for the still-water and moving-water steady state solutions has been provided in [159], where some numerical examples are shown to demonstrate the advantage of moving-water well-balanced methods, especially for solutions near a moving-water equilibrium. Some moving-water well-balanced methods been proposed in [131, 14, 31, 30], and high order accurate well-balanced weighted essentially non-oscillatory (WENO) methods can be found in [110, 132, 25]. In [150], well-balanced and positivity-preserving DG methods were developed for the SWEs with moving-water equilibrium.

Designing well-balanced methods for the Ripa model (2.1) can be a challenging task, because its steady states are more complicated than those of the SWEs [32]. In the last few years, there have been some studies on well-balanced methods for the Ripa models, mostly focusing on the zero-velocity steady-state solutions (2.5) and (2.6). The first well-balanced scheme for the Ripa system is developed in [32]. The proposed scheme

is well-balanced, positivity preserving and does not develop spurious pressure oscillations in the neighborhood of temperature jumps. A second-order well-balanced finite volume scheme for the Ripa system in one and two dimensions is designed in [143]. High-order well-balanced WENO schemes that possess sharp shock transition were designed for the Ripa system in [69], by extending the well-balanced technique developed in [152] for the SWEs. Other related works can be found in [44, 73, 134].

High order accurate numerical schemes such as finite difference and finite volume WENO schemes, spectral methods, and DG methods have been developed to reduce the number of computational cells and thus reduce the computational time, while still achieving high order accuracy. Specifically, the DG method is a class of finite element methods in which the numerical solutions and test functions live in a discontinuous piecewise polynomial space. A review of the method can be found in [33]. The DG method combines the flexibility of the finite element method and the stability of the finite volume method. Additionally, DG methods enjoy advantages including high order accuracy, high parallel efficiency, flexibility for hp-adaptivity and arbitrary geometry and meshes, etc.

The main objective of this chapter is to develop high order well-balanced DG methods for the Ripa system (2.1), which can preserve the still-water equilibrium solution (2.5) and moving-water steady state solution (2.8) exactly at the discrete level. This will be the first moving-water well-balanced method for the Ripa model, to our best knowledge. To achieve this goal, we start with a transformation between the conservative variables and the equilibrium variables (to be defined in Section 2.4). For the finite element methods, the initial conditions are projected into a polynomial solution spaces to provide a numerical

initial condition. Even though the exact initial conditions are in moving-water equilibrium, the numerical initial conditions may no longer be. One challenge in designing well-balanced methods is the recovery of the well-balanced states from the numerical initial condition, which is achieved by a new choice of projection operator. Then, we can decompose the numerical solution into an equilibrium part and the fluctuation part, and show that the fluctuation part is zero at the steady state. Following the idea of hydrostatic reconstruction, one can carefully design well-balanced numerical fluxes. A well-balanced source term approximation is achieved by treating the equilibrium and fluctuation parts in different ways.

The methods presented here are extensions of the ones in [150] for the moving-water equilibrium of the SWEs. In this work, several improvements over the well-balanced methods in [150] have been presented to improve the algorithm. First, the recovery of the well-balanced states in [150] are obtained by solving nonlinear equations, which could be complicated. A special projection of the exact initial condition is proposed in this chapter in order to provide a much easier way to recover the well-balanced states. This also leads to a more simple way to evaluate the well-balanced components of the solutions at each time step. Second, due to the existence of the potential temperature field, extra attention is provided to accommodate more components in the conservative and the equilibrium variables. Lastly, we simplify the procedure to evaluate the updated cell boundary values resulted from the hydrostatic reconstruction. This leads to a more efficient and simple way to compute the well-balanced numerical fluxes than those computed in [150]. Furthermore, the same idea can be generalized to preserve other steady state solutions of the Ripa system, including

the isobaric and constant water height equilibria which do not appear in the SWEs.

This chapter is organized as follows. In Section 2.2, we introduce some notation and discuss the well-balanced DG method for still-water lake-at-rest equilibrium. Numerical examples using the methods outlined in Section 2.2 are found in Section 2.3 to demonstrate the accuracy and well-balanced property of the scheme as well as show that it can aptly handle perturbations of the still-water steady state and provide good resolution for discontinuous solutions. Although our main focus is on moving-water well-balanced methods, it is useful to present the still-water preserving DG methods in Sections 2.2 and 2.3. The purpose is twofold. First, the well-balanced DG method for still-water equilibrium is not available in the literature and a simple method achieving such a goal is interesting by itself. Second, this would serve as a basis for the design of the moving-water well-balanced methods for the Ripa system, which is presented in Section 2.4. We will also show that when applied to still-water equilibrium, the proposed moving-water well-balanced methods reduce to the still-water well-balanced method in Section 2.2. Additionally, we show how the method can be modified to preserve the constant water height and isobaric steady states. In Section 2.5, numerical examples of our methods for the one-dimensional Ripa system are provided, to demonstrate the high order accuracy, well-balanced property, and good resolution for smooth and discontinuous solutions.

2.2 Still-Water Well-Balanced DG Methods

A variety of well-balanced DG methods for the SWEs with still-water steady state solutions

$$u = 0, \qquad h + b = \text{constant} \tag{2.9}$$

have been developed. In this section, we extend the method introduced in [155, 160] to provide still-water well-balanced methods for the Ripa system (2.1), with the still-water steady state solution (2.5). The same structure will be generalized in Section 2.4 for the moving-water case. The still-water well-balanced DG methods are much simpler than the moving-water well-balanced methods, and would be useful if one's target is to simulate a small perturbation of the still-water equilibrium state. The still-water well-balanced method presented can be extended to two-dimensional Ripa models easily. However, there is no general form of the moving-water equilibrium in two dimensions, hence no two-dimensional moving-water well-balanced methods are available.

2.2.1 Notation and DG Numerical Scheme

We seek an approximation U_τ which belongs to the finite dimensional DG space $\mathbb{V}_\tau^k$ where k is the polynomial degree. We refer the reader to Section 1.3.2 for details on the function and mesh discretization as well as the formulation for the traditional DG scheme. However, we note that $\alpha = \max(|u| + \sqrt{gh\theta})$ in the definition of the Lax-Friedrichs flux where the maximum can either be taken locally in each cell or globally over the entire domain.

We aim to preserve the lake-at-rest still-water solution (2.5). The well-balanced

numerical scheme, as described in [160] for the SWEs, has the form:

$$\int_{I_j} \partial_t \boldsymbol{U}_\tau^n v \, dx - \int_{I_j} \boldsymbol{F}(\boldsymbol{U}_\tau^n) \partial_x v \, dx + \hat{\boldsymbol{F}}^l_{j+\frac{1}{2}} v^-_{j+\frac{1}{2}} - \hat{\boldsymbol{F}}^r_{j-\frac{1}{2}} v^+_{j-\frac{1}{2}} = \int_{I_j} \boldsymbol{S}(\boldsymbol{U}_\tau^n, b_\tau) v \, dx. \quad (2.10)$$

The design of $\hat{\boldsymbol{F}}^l_{j+\frac{1}{2}}$ and $\hat{\boldsymbol{F}}^r_{j-\frac{1}{2}}$, known as the well-balanced numerical fluxes, are defined below in Section 2.2.2. The source term approximation will be discussed in Section 2.2.3. This method is equivalent to

$$\begin{aligned} &\int_{I_j} \partial_t \boldsymbol{U}_\tau^n v \, dx - \int_{I_j} \boldsymbol{F}(\boldsymbol{U}_\tau^n) \partial_x v \, dx + \hat{\boldsymbol{F}}_{j+\frac{1}{2}} v^-_{j+\frac{1}{2}} - \hat{\boldsymbol{F}}_{j-\frac{1}{2}} v^+_{j-\frac{1}{2}} \\ &\qquad = \int_{I_j} \boldsymbol{S}(\boldsymbol{U}_\tau^n, b_\tau) v \, dx + (\hat{\boldsymbol{F}}_{j+\frac{1}{2}} - \hat{\boldsymbol{F}}^l_{j+\frac{1}{2}}) v^-_{j+\frac{1}{2}} - (\hat{\boldsymbol{F}}_{j-\frac{1}{2}} - \hat{\boldsymbol{F}}^r_{j-\frac{1}{2}}) v^+_{j-\frac{1}{2}}, \end{aligned} \quad (2.11)$$

and the terms $\hat{\boldsymbol{F}}_{j+\frac{1}{2}} - \hat{\boldsymbol{F}}^l_{j+\frac{1}{2}}$ and $\hat{\boldsymbol{F}}_{j-\frac{1}{2}} - \hat{\boldsymbol{F}}^r_{j-\frac{1}{2}}$ are at the level of $O(\tau^{k+1})$ (independent of the smoothness of the solution U) when the bottom topography b is smooth, and can be viewed as high order correction terms to the source term approximation.

2.2.2 Well-Balanced Numerical Fluxes

The conservative variable can be decomposed into a reference equilibrium state $\boldsymbol{U}_\tau^e$ and a fluctuation part $\boldsymbol{U}_\tau^f$. In each computational cell I_j, the equilibrium state $\boldsymbol{U}_\tau^e(x)$ can be computed from the constant equilibrium variables defined as (setting $H = h + b$)

$$\hat{\boldsymbol{V}}_j = \begin{pmatrix} \hat{H}_j \\ \hat{m}_j \\ \hat{\theta}_j \end{pmatrix} = \begin{pmatrix} (h_\tau + b_\tau)\left(x^-_{j+\frac{1}{2}}\right) \\ (hu)_\tau\left(x^-_{j+\frac{1}{2}}\right) \\ \theta_\tau\left(x^-_{j+\frac{1}{2}}\right) \end{pmatrix}, \quad (2.12)$$

and the bottom function b_τ in the form of

$$\boldsymbol{U}^e_{\tau,j}(x) = \begin{pmatrix} h^e_{\tau,j}(x) \\ (hu)^e_{\tau,j}(x) \\ (h\theta)^e_{\tau,j}(x) \end{pmatrix} = \begin{pmatrix} \hat{H}_j - b_\tau(x) \\ \hat{m}_j \\ \left(\hat{H}_j - b_\tau(x)\right)\hat{\theta}_j \end{pmatrix}, \quad (2.13)$$

which belongs to the DG space $\mathbb{V}_\tau^k$. The fluctuation part $\boldsymbol{U}_\tau^f$ is then determined by the decomposition

$$\boldsymbol{U}_\tau = \boldsymbol{U}_\tau^e + \boldsymbol{U}_\tau^f. \tag{2.14}$$

It is easy to observe that at the still-water steady state (2.5), the reference equilibrium state $\boldsymbol{U}_\tau^e$ is equal to $\boldsymbol{U}_\tau$, and $\boldsymbol{U}_\tau^f$ reduces to 0.

The numerical fluxes are constructed following the approaches in [155, 160]. After computing the boundary values $\boldsymbol{U}^{\pm}_{\tau,j+\frac{1}{2}}$ at the time step t^n, we set

$$b^*_{\tau,j+\frac{1}{2}} = \max\left(b^+_{\tau,j+\frac{1}{2}}, b^-_{\tau,j+\frac{1}{2}}\right), \tag{2.15}$$

by utilizing the idea of hydrostatic reconstruction in [1]. A new hydrostatic reconstruction method has been presented in [29], which performs better in some cases, for example, when water runs down a hill. Next, the height function at the cell interface can be redefined as:

$$h^{*,\pm}_{\tau,j+\frac{1}{2}} = \max\left(0, h^{\pm}_{\tau,j+\frac{1}{2}} + b^{\pm}_{\tau,j+\frac{1}{2}} - b^*_{j+\frac{1}{2}}\right), \tag{2.16}$$

or equivalently,

$$h^{*,-}_{\tau,j+\frac{1}{2}} = \max\left(0, \hat{H}_j - b^*_{\tau,j+\frac{1}{2}} + (h^f)^-_{\tau,j+\frac{1}{2}}\right), \quad h^{*,+}_{\tau,j+\frac{1}{2}} = \max\left(0, \hat{H}_{j+1} - b^*_{\tau,j+\frac{1}{2}} + (h^f)^+_{\tau,j+\frac{1}{2}}\right), \tag{2.17}$$

by using the new defined $\hat{H}_j$ and h^f. This results in following updated boundary values of U:

$$\boldsymbol{U}^{*,\pm}_{\tau,j+\frac{1}{2}} = \begin{pmatrix} h^{*,\pm}_{\tau,j+\frac{1}{2}} \\ h^{*,\pm}_{\tau,j+\frac{1}{2}} u^{\pm}_{\tau,j+\frac{1}{2}} \\ h^{*,\pm}_{\tau,j+\frac{1}{2}} \theta^{\pm}_{\tau,j+\frac{1}{2}} \end{pmatrix}. \tag{2.18}$$

Finally, the left and right fluxes are given as:

$$\begin{aligned}\hat{\boldsymbol{F}}^{l}_{j+\frac{1}{2}} &= \boldsymbol{f}\Big(\boldsymbol{U}^{*,-}_{\tau,j+\frac{1}{2}}, \boldsymbol{U}^{*,+}_{\tau,j+\frac{1}{2}}\Big) + \boldsymbol{F}\Big(U^{-}_{\tau,j+\frac{1}{2}}\Big) - \boldsymbol{F}\Big(U^{*,-}_{\tau,j+\frac{1}{2}}\Big),\\ \hat{\boldsymbol{F}}^{r}_{j-\frac{1}{2}} &= \boldsymbol{f}\Big(\boldsymbol{U}^{*,-}_{\tau,j-\frac{1}{2}}, \boldsymbol{U}^{*,+}_{\tau,j-\frac{1}{2}}\Big) + \boldsymbol{F}\Big(U^{+}_{\tau,j-\frac{1}{2}}\Big) - \boldsymbol{F}\Big(U^{*,+}_{\tau,j-\frac{1}{2}}\Big).\end{aligned} \tag{2.19}$$

At the steady state (2.5), although the original cell boundary values $h^{+}_{\tau,j+\frac{1}{2}}$, $h^{-}_{\tau,j+\frac{1}{2}}$ may not be the same, we have $h^{\pm}_{\tau,j+\frac{1}{2}} + b^{\pm}_{\tau,j+\frac{1}{2}} = constant$. Then following the definition in (2.16), this implies $h^{*,+}_{\tau,j+\frac{1}{2}} = h^{*,-}_{\tau,j+\frac{1}{2}}$. Since $u^{\pm}_{\tau,j+\frac{1}{2}} = 0$ and $\theta^{\pm}_{\tau,j+\frac{1}{2}} = constant$ at the steady state (2.5), we can conclude that $\boldsymbol{U}^{*}_{\tau}$ is continuous at cell interfaces, i.e. $\boldsymbol{U}^{*,+}_{\tau,j+\frac{1}{2}} = \boldsymbol{U}^{*,-}_{\tau,j+\frac{1}{2}}$. Furthermore, due to the consistency of the numerical flux $\boldsymbol{f}$, it can be shown that $\hat{\boldsymbol{F}}^{l}_{j+\frac{1}{2}} = \boldsymbol{F}\Big(\boldsymbol{U}^{-}_{\tau,j+\frac{1}{2}}\Big)$ and $\hat{\boldsymbol{F}}^{r}_{j-\frac{1}{2}} = \boldsymbol{F}\Big(\boldsymbol{U}^{+}_{\tau,j-\frac{1}{2}}\Big)$, which is a desirable quality for achieving the well-balanced property.

2.2.3 The Source Term Decomposition

The source term can be decomposed similarly as in (2.14), because $\boldsymbol{S}(\boldsymbol{U}, b) = -gh\theta b_x$ is linear with respect to the conservative variable $h\theta$:

$$\int_{I_j} \boldsymbol{S}\big(\boldsymbol{U}_\tau, b_\tau\big) v \; dx = \int_{I_j} \boldsymbol{S}\big(\boldsymbol{U}^{e}_{\tau}, b_\tau\big) v \; dx + \int_{I_j} \boldsymbol{S}\big(\boldsymbol{U}^{f}_{\tau}, b_\tau\big) v \; dx. \tag{2.20}$$

Notice that the second term on the right hand side can be directly computed by any quadrature rule with sufficient accuracy. On the other hand, since $\boldsymbol{U}^{e}_{\tau,j}(x)$ is the equilibrium state, the first term on the right hand side can be expanded as

$$\int_{I_j} \boldsymbol{S}\Big(\boldsymbol{U}^{e}_{\tau}, b_\tau\Big) v \; dx = -\int_{I_j} \boldsymbol{F}\Big(\boldsymbol{U}^{e}_{\tau}\Big) v_x \; dx + \boldsymbol{F}\Big(\boldsymbol{U}^{e,-}_{\tau,j+\frac{1}{2}}\Big) v^{-}_{j+\frac{1}{2}} - \boldsymbol{F}\Big(\boldsymbol{U}^{e,+}_{\tau,j-\frac{1}{2}}\Big) v^{+}_{j-\frac{1}{2}}, \tag{2.21}$$

where $\boldsymbol{U}^{e}_{\tau} \in \mathbb{V}^{k}_{\tau}$ is a polynomial. On the numerical level, when all these integrals are replaced by numerical integrations, this equality holds exactly if one uses a quadrature rule which is accurate for polynomial of degree $3k-1$. If a less accurate quadrature rule is used, the

equality holds approximately, up to the accuracy of the quadrature rule. Thus the source term can be evaluated using the following form:

$$\begin{aligned}&\int_{I_j} \boldsymbol{S}\Big(\boldsymbol{U}_\tau, b_\tau\Big) v \; dx \\ &= -\int_{I_j} \boldsymbol{F}\Big(U_\tau^e\Big) v_x \; dx + \boldsymbol{F}\Big(\boldsymbol{U}^{e,-}_{\tau,j+\frac{1}{2}}\Big) v^-_{j+\frac{1}{2}} - \boldsymbol{F}\Big(\boldsymbol{U}^{e,+}_{\tau,j-\frac{1}{2}}\Big) v^+_{j-\frac{1}{2}} + \int_{I_j} \boldsymbol{S}\Big(\boldsymbol{U}^f_\tau, b_\tau\Big) v \; dx.\end{aligned} \tag{2.22}$$

Since the only non-zero source term is in the momentum equation, we can plug in the definition of the flux and $\boldsymbol{U}^e_\tau$ to obtain the following equivalent source term approximation

$$\begin{aligned}-\int_{I_j} g h_\tau \theta_\tau (b_\tau)_x v \; dx &= \frac{1}{2} g \hat{\theta}_j \Big(\big(\hat{H}_j - b_\tau\big)^2 v \Big)^-_{j+\frac{1}{2}} - \frac{1}{2} g \hat{\theta}_j \Big(\big(\hat{H}_j - b_\tau\big)^2 v \Big)^+_{j-\frac{1}{2}} \\ &\quad - \int_{I_j} \frac{1}{2} g \hat{\theta}_j \big(\hat{H}_j - b_\tau\big)^2 v_x \; dx - \int_{I_j} g \Big(h_\tau \theta_\tau + (\hat{H}_j - b_\tau)\hat{\theta}_j \Big) (b_\tau)_x v \; dx \\ &= g \hat{\theta}_j \Big(\frac{b_\tau^2}{2} v - \hat{H}_j b_\tau v \Big)^-_{j+\frac{1}{2}} - g \theta \Big(\frac{b_\tau^2}{2} v - \hat{H}_j b_\tau v \Big)^+_{j-\frac{1}{2}} \\ &\quad - \int_{I_j} g \hat{\theta}_j \Big(\frac{b_\tau^2}{2} - \hat{H}_j b_\tau \Big) v_x \; dx - \int_{I_j} g \Big(h_\tau \theta_\tau + (\hat{H}_j - b_\tau)\hat{\theta}_j \Big) (b_\tau)_x v \; dx.\end{aligned} \tag{2.23}$$

This formulation is exactly the extension of the source term approximation introduced in [154] for the SWEs. However, a different well-balanced approach was used in that paper, the source term was decomposed as $-ghb_x = -g(h+b)b_x + g(b^2)_x/2$.

Remark 1. *As we explained, the equality* (2.21) *holds exactly when a quadrature rule accurate for polynomial of degree* $3k-1$ *is used. If this is the case, we can simply replace the source term approximation* (2.22) *by the direct numerical integration of the source term* $-\int_{I_j} g h_\tau \theta_\tau (b_\tau)_x v \; dx$ *with this sufficiently high accurate quadrature rule. In other words, the numerical integration of the source term is automatically well-balanced, without any special treatment. The same conclusion has been observed in [155] for the SWEs.*

Remark 2. *In this chapter, we only consider a source term due to the bottom topography, which is linear with respect to the conservative variable U. When other nonlinear source terms (for instance the Manning friction term) are included, the source term decomposition* (2.20) *will not hold any more, but one can introduce the following decomposition in a similar manner*

$$\int_{I_j} \boldsymbol{S}(\boldsymbol{U}, b) v \, dx = \int_{I_j} \boldsymbol{S}(\boldsymbol{U}^e, b) v \, dx + \int_{I_j} \left(\boldsymbol{S}(\boldsymbol{U}, b) - \boldsymbol{S}(\boldsymbol{U}^e, b)\right) v \, dx. \tag{2.24}$$

Note that when $\boldsymbol{S}$ is linear, i.e., $\boldsymbol{S}(\boldsymbol{U}, b) - \boldsymbol{S}(\boldsymbol{U}^e, b) = \boldsymbol{S}(\boldsymbol{U} - \boldsymbol{U}^e, b) = \boldsymbol{S}(\boldsymbol{U}^f, b)$, this decomposition reduces to (2.20). *The first term on the right hand side of* (2.24) *can be approximated as in* (2.21), *and the second term on the right hand side is computed by a straightforward numerical integral. This will provide well-balanced source term approximation.*

Remark 3. *When the bottom topography is flat (i.e., $b(x) =$ constant C), the traditional DG methods are recovered from our well-balanced DG scheme, that is, the source term approximation reduces to 0 exactly and the left and right numerical fluxes reduce to the original fluxes. First, when $b(x) = C$ is a constant, one has $b_\tau(x) = C$, therefore the source term approximation presented in* (2.23) *simply reduces to zero. Second, when $b(x) = C$, $b^*_{\tau, j\pm\frac{1}{2}}$ defined in* (2.15), *is also the same constant, which leads to*

$$h^{*,\pm}_{\tau,j+\frac{1}{2}} = \max\left(0, h^{\pm}_{\tau,j+\frac{1}{2}} + b^{\pm}_{\tau,j+\frac{1}{2}} - b^*_{j+\frac{1}{2}}\right) = h^{\pm}_{\tau,j+\frac{1}{2}}.$$

More generally we have $\boldsymbol{U}^{*,\pm}_{\tau,j+\frac{1}{2}} = \boldsymbol{U}^{\pm}_{\tau,j+\frac{1}{2}}$. Therefore the left and right numerical fluxes defined in (2.19) *reduce to the original DG fluxes: $\hat{\boldsymbol{F}}^l_{j+\frac{1}{2}} = \hat{\boldsymbol{F}}_{j+\frac{1}{2}}$, $\hat{\boldsymbol{F}}^r_{j-\frac{1}{2}} = \hat{\boldsymbol{F}}_{j-\frac{1}{2}}$.*

At the end of this section, we show that the proposed numerical methods are well-balanced for the still-water steady state solutions.

Proposition 4. *The DG scheme (2.10) for the Ripa system* (2.1)*, paired with the numerical fluxes (2.19) and source term approximation (2.22), is well-balanced for the still-water steady state (2.5).*

One can easily verify this holds by observing that $\boldsymbol{U}_\tau^f = 0$ at the lake-at-rest still-water steady state, and thus the well-balanced numerical fluxes reduce to $\hat{\boldsymbol{F}}^l_{j+\frac{1}{2}} = \boldsymbol{F}\Big(\boldsymbol{U}^-_{\tau,j+\frac{1}{2}}\Big)$ and $\hat{\boldsymbol{F}}^r_{j-\frac{1}{2}} = \boldsymbol{F}\Big(\boldsymbol{U}^+_{\tau,j-\frac{1}{2}}\Big)$.

Remark 5. *Although the description is for one dimensional problem only, the proposed well-balanced methods for the still-water steady state (2.5) can be extended to the two-dimensional problem in a straightforward way.*

2.3 Numerical Tests for the Still-Water Well-Balanced Methods

In this section, we present numerical results of our still-water well-balanced DG methods, described in Section 2.2, for the Ripa system (2.1). Piecewise quadratic polynomials ($k = 2$) in space, paired with the third order TVD Runge-Kutta time discretization (1.19), are used in the tests. The CFL number is taken to be 0.1. The constant M in the TVB limiter is taken to be 0, except for the accuracy test, in which no slope limiter was implemented. The gravitational constant g is fixed to be $9.812 m/s^2$. We compute multiple types of tests: an accuracy test, verification of the well-balanced property, small

perturbations of steady states, and tests for discontinuous solutions.

2.3.1 Test for Accuracy

In this subsection, we test the accuracy of our still-water well-balanced scheme for smooth solutions. The initial conditions in the domain $x \in [0, 1]$ are given by

$$\begin{cases} h(x,0) = 5 + e^{\sin(2\pi x)}, \\ (hu)(x,0) = \sin(\cos(2\pi x)), \\ \theta(x,0) = \sin(2\pi x) + 2, \end{cases} \tag{2.25}$$

with the bottom function $b(x) = \sin^2(\pi x)$ and periodic boundary conditions. We run the simulation until time 0.02, while the solution is still smooth. Since there is no explicitly known solution in this case, the errors are computed by comparing numerical results of uniform meshes with size J and $2J$. Figure 2.1 displays the numerical solutions at time 0.02 with 200 uniform cells. Table 2.1 contains the L^1 errors and numerical orders or accuracy. We can observe that the third order convergence rate is achieved which matches our expectation of order $k + 1$ accuracy.

2.3.2 Tests for the Well-Balanced Property

The following tests are chosen to verify that the DG methods preserve the still-water steady state (2.3) with a non-flat bottom. For these examples, the errors are calculated by comparing the numerical results to the initial conditions.

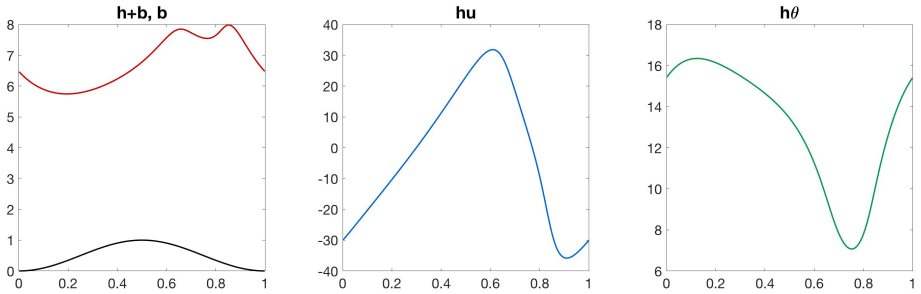


Figure 2.1: Solutions of well-balanced methods for the accuracy test in Section 2.3.1 at time $t = 0.02$.

	h		hu		$h\theta$	
No. of Cells	L^1 Error	Order	L^1 Error	Order	L^1 Error	Order
25	0.001347		0.012963		0.001480	
50	0.000205	2.7137	0.001757	2.8829	0.000206	2.8442
100	2.9801e-05	2.7842	0.000226	2.9580	3.2526e-05	2.6638
200	4.0093e-06	2.8939	2.9190e-05	2.9539	5.1699e-06	2.6534
400	5.0280e-07	2.9953	3.6862e-06	2.9852	7.3483e-07	2.8146

Table 2.1: L^1 errors and orders of accuracy for the test in Section 2.3.1, using the still-water well-balanced method.

First, we will consider the following still-water steady state for $x \in [0, 1]$

$$h + b = 2, \qquad u = 0, \qquad \theta = 10. \tag{2.26}$$

The bottom function is discontinuous and defined as

$$b(x) = \begin{cases} 1, & \text{if } 0.3 < x < 0.7, \\ 0, & \text{otherwise,} \end{cases}$$

and transmissive boundary conditions are employed. We plot the numerical results at time $t = 1$ with 200 uniform cells in Figure 2.2, with the L^1 and L^∞ errors shown in Table 2.2. From the error table, it can be concluded that the well-balanced property is achieved.

Test	Error Type	h	hu	$h\theta$
(2.26)	L^1 Error	9.9959e−16	7.6925e−15	3.9599e−15
	L^∞ Error	2.6401e−13	4.2333e−12	1.9380e−12
(2.28)	L^1 Error	6.9148e−17	1.6519e−15	2.7659e−16
	L^∞ Error	2.8422e−14	4.2056e−12	1.1369e−13

Table 2.2: L^1 and L^∞ errors for the well-balanced test to preserve the still-water equilibria.

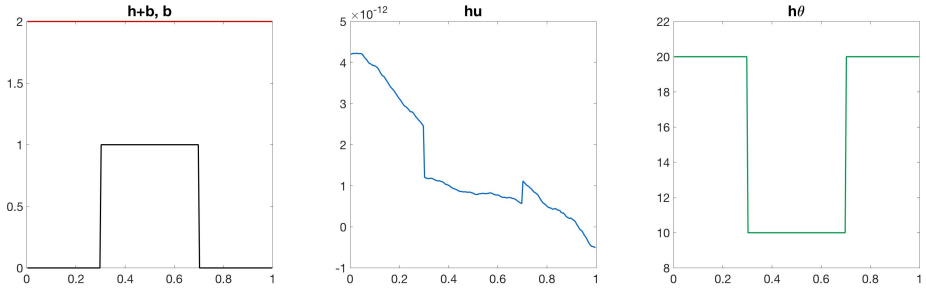


Figure 2.2: Solution of well-balanced methods for the still-water steady state problem (2.26) at $t = 1$. The steady state is preserved as $h + b$, hu, and θ are constant.

Second, we consider another lake-at-rest still-water steady state problem in which the bottom function defined on the interval domain $[-2, 2]$ consists of two humps and is defined as

$$b(x) = \begin{cases} 0.85(\cos(10\pi(x + 0.9)) + 1), & \text{if } -1 \le x \le -0.8, \\ 1.25(\cos(10\pi(x - 0.4)) + 1), & \text{if } 0.3 \le x \le 0.5, \\ 0, & \text{otherwise.} \end{cases} \tag{2.27}$$

The initial conditions are set as:

$$h(x, 0) = 6 - b(x), \qquad u(x, 0) = 0, \qquad \theta(x, 0) = 4. \tag{2.28}$$

We run the simulation until time $t = 1$. The numerical results are shown in

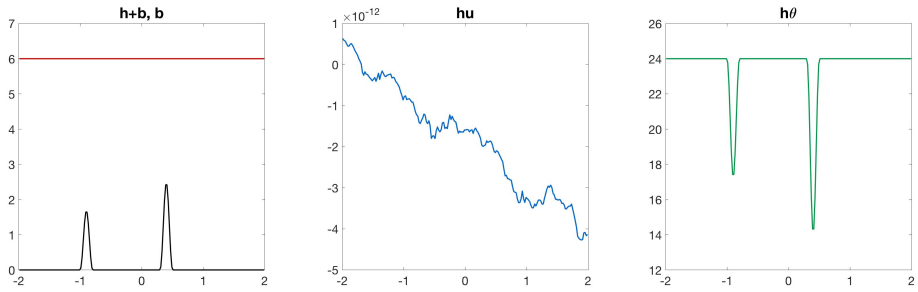


Figure 2.3: Solution of well-balanced methods for the still-water steady state flow over two bumps (2.28) at $t = 1$, with 200 uniform cells.

Figure 2.3, and the L^1, L^∞ errors of the numerical solution with 200 uniform cells are presented in Table 2.2, which shows that the well-balanced property is again maintained. For comparison, we also run the same test with the traditional DG method (i.e., the standard numerical fluxes and the straightforward integration of the source term). The numerical results are presented in Figures 2.4 and 2.5, where we can observe that the steady state is not preserved. It can be seen that, at the region where the bottom function is non-zero, $h + b$ does not preserve the constant steady state, and the solution hu is non-zero.

2.3.3 Tests of Small Perturbations

The tests in this subsection are selected to demonstrate that perturbations to the still-water steady states are aptly captured by the proposed well-balanced scheme. We will also compare the performance of well-balanced and traditional DG schemes. Prior to

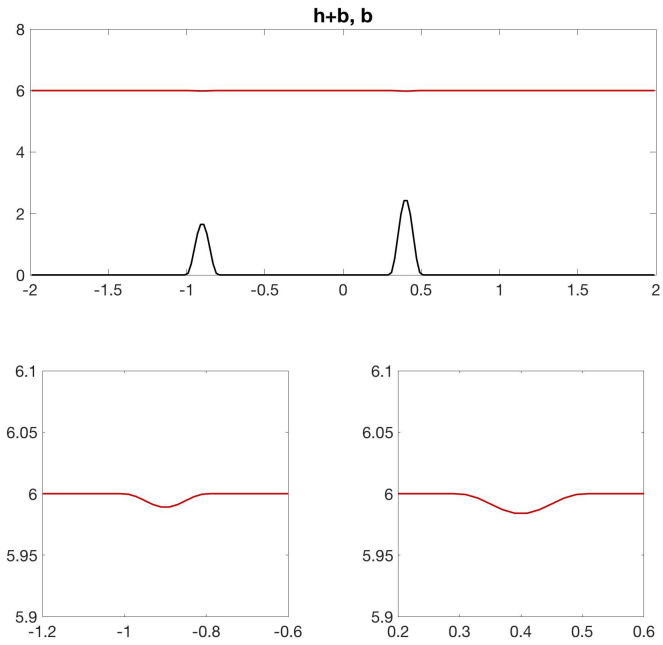


Figure 2.4: Solution of non-well-balanced methods for the still-water steady state flow over two bumps (2.28) with 200 uniform cells. It can be seen that the steady state for $h + b$ is not preserved, as the water height is not flat above the not-constant portions of the bottom function. The bottom figures show zoomed in images of $h + b$ where b is non-constant.

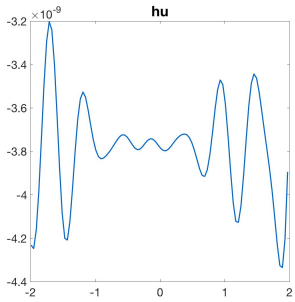


Figure 2.5: Solution for hu of the non-well-balanced methods for the still-water steady state flow over two bumps (2.28) with 200 uniform cells. It can be seen that the steady state solution of hu is not preserved as it is non-zero.

defining the perturbations, let us denote χ as the indicator function on an interval:

$$\chi_{[a,b]} = \begin{cases} 1, & \text{if } x \in [a,b], \\ 0, & \text{otherwise,} \end{cases} \tag{2.29}$$

and denote the still-water steady state initial conditions (2.28) in Section 2.3.2 as

$$\big(h_{eq}, (hu)_{eq}, (h\theta)_{eq}\big)(x,0).$$

We will examine the numerical results with three different perturbations outlined below.

(a) *Small perturbation to both h and $h\theta$:* First, perturbations of sizes $0.01, 0.04$ are applied to the initial conditions of h and $h\theta$, respectively, in the interval $[-1.5, -1.4]$:

$$\big(h, hu, h\theta\big)(x,0) = \big(h_{eq}, (hu)_{eq}, (h\theta)_{eq}\big)(x,0) + [0.01, 0, 0.04]\chi_{[-1.5,-1.4]}. \tag{2.30}$$

(b) *Small perturbation to h:* Second, a small perturbation of size 0.01 is applied to the initial condition of h in the interval $[-1.5, -1.4]$:

$$\big(h, hu, h\theta\big)(x,0) = \big(h_{eq}, (hu)_{eq}, (h\theta)_{eq}\big)(x,0) + [0.01, 0, 0]\chi_{[-1.5,-1.4]}. \tag{2.31}$$

(c) *Large perturbation to h:* Third, a larger perturbation of size 1 is applied to the initial condition of h in the interval $[-1.5, -1.4]$:

$$\big(h, hu, h\theta\big)(x,0) = \big(h_{eq}, (hu)_{eq}, (h\theta)_{eq}\big)(x,0) + [1, 0, 0]\chi_{[-1.5,-1.4]}. \tag{2.32}$$

The numerical test with the initial condition (2.32) is considered in [32, 143] and will be compared to the other tests (2.30), (2.31) in this section. The perturbation of the test (2.30) splits into two waves moving away from the point of origin. On the other hand,

the perturbation of the tests (2.31) and (2.32) split into three waves. The two outer waves move away from the point of origin, as expected, while there is a third wave in the center which remains unmoved in the perturbed region. The amplitude of this wave reduces until it reaches a total water height of approximately 6.005 for the test (2.31) and 6.479 for the test (2.32). In the same interval, the value of $h\theta$ converges to a value of approximately 23.980 for the test (2.31) and 22.215 for the test (2.32). It is easy to verify that, in this region, the solution converges to an isobaric steady state (2.6) as $h^2\theta$ stays constant inside and outside of the interval.

We run the test until time $t = 0.125$. At this point for all perturbation examples, the downstream moving wave has passed through both bumps of the bottom topography function and the upstream moving wave has exited the domain. Figures 2.6 and 2.7 compares the numerical results of three perturbation tests for $h + b$ using well-balanced DG methods with mesh sizes of 200 and 800 uniform cells. It can be observed that the solutions are well captured by our methods. Figure 2.8 compares the results of three cases for hu and $h\theta$. We would like to point out that the solutions contain the isobaric steady state in the interval $[-1.5, -1.4]$ (for examples (2.31) and (2.32)), and our methods perform well for these test cases.

Next we examine the performance of the traditional DG scheme on the same test cases. Figure 2.9 includes the numerical results for these tests. It can be seen that the traditional DG scheme does not handle the perturbation as well as the well-balanced scheme. It is evident for (2.30) and (2.31) in which the initial perturbations were small.

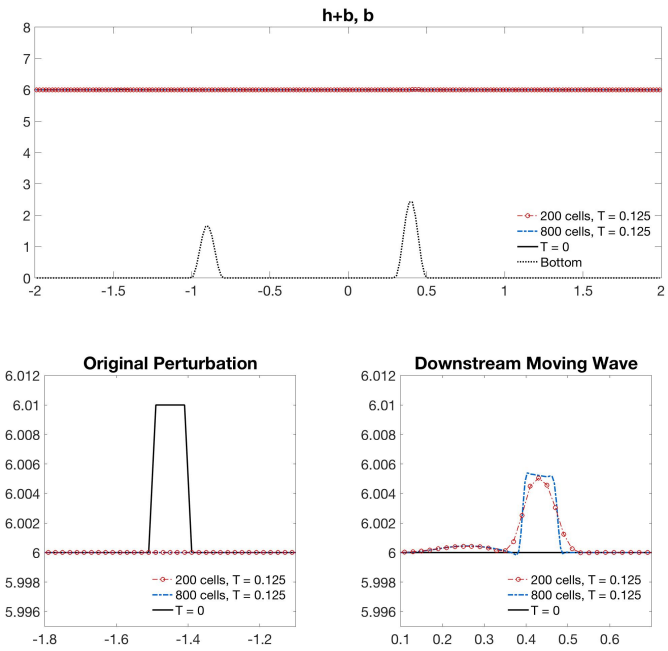


Figure 2.6: Initial conditions and numerical solution of water surface $h + b$ for the perturbation tests (2.30), with 200 and 800 uniform cells. The initial perturbation split into two waves moving away from the point of origin. At time $t = 0.125$, the downstream moving wave has passed over the two bumps in the bottom topography and the upstream wave has exited the domain. The bottom left plot contains the original perturbation and the bottom right plot contains a zoomed in image of the remaining downstream moving wave.

2.3.4 Tests for Riemann Problems

In this subsection, we consider problems that contain discontinuities in the initial conditions, known as Riemann problems or dam break problems for the SWEs.

First, we consider the dam break problem in the computational domain $[-200, 400]$ with the initial conditions

$$(h, u, \theta)(x, 0) = \begin{cases} (5, 0, 20), & \text{when } x < 0, \\ (10, 40, 5), & \text{when } x \geq 0. \end{cases} \tag{2.33}$$

The bottom function is given as $b(x) = 0$ and we employ transmissive boundary conditions.

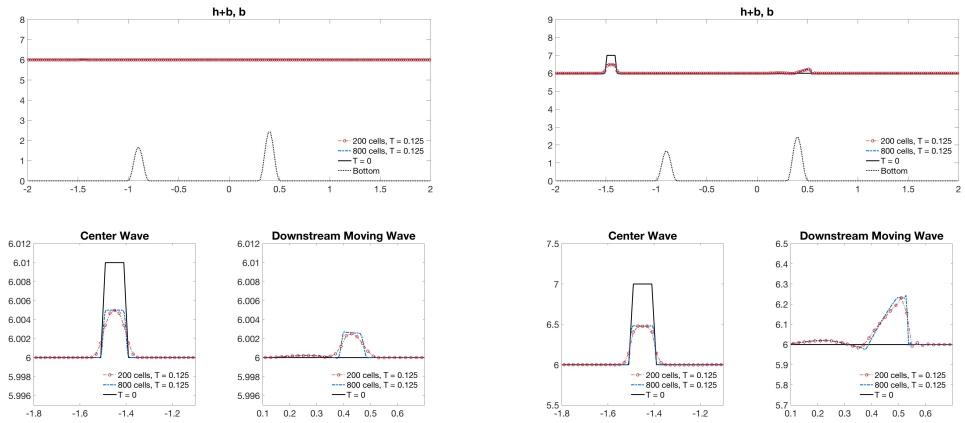


Figure 2.7: Initial conditions and numerical solution of water surface $h + b$ for the perturbation tests (2.31) (left plots) and (2.32) (right plots), with 200 and 800 uniform cells. The initial perturbation split into three waves, one unmoved in the center and the others moving away from the point of origin. At time $t = 0.125$, the downstream moving wave has passed over the two bumps in the bottom topography and the upstream wave has exited the domain. The bottom plots contain zoomed in images of the remaining waves.

This example is computed using the still-water well-balanced DG method with a uniform mesh of 200 cells. Figure 2.10 displays the numerical results at times $t = 1, 2,$ and 3, and the solutions are captured well by our methods.

Next, we look at the dam break problem over a non-flat bottom. We define the bottom topography on the domain $[-1, 1]$ to include two bumps:

$$b(x) = \begin{cases} 0.5\cos\left(10\pi(x + 0.3)\right) + 1, & \text{when } -0.4 \le x \le -0.2, \\ 0.75\cos\left(10\pi(x - 0.3)\right) + 1, & \text{when } 0.2 \le x \le 0.4, \\ 0, & \text{otherwise.} \end{cases} \tag{2.34}$$

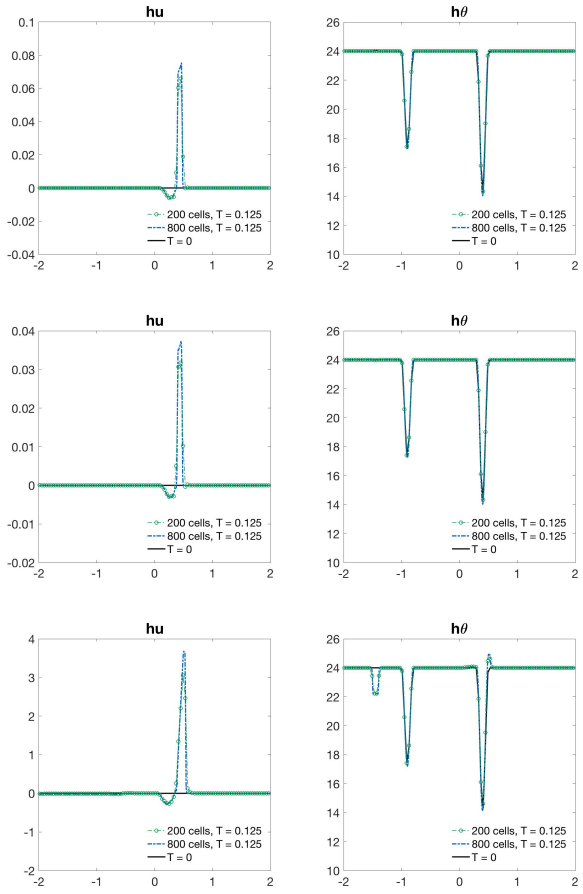


Figure 2.8: Initial conditions and numerical solution of $hu, h\theta$ for the perturbation tests (2.30) (top row), (2.31) (middle row), (2.32) (bottom row), with 200 and 800 uniform cells at time $t = 0.125$. Although the shape of hu is similar for all cases, the amplitude varies.

The initial conditions are defined as:

$$(h, u, \theta)(x, 0) = \begin{cases} (5, 0, 3), & \text{when } x < 0, \\ (2, 0, 5), & \text{when } x \geq 0. \end{cases} \tag{2.35}$$

We test the problem until time 0.05. Figure 2.11 shows the results at the final time, compared to the initial conditions. We see that the potential temperature field at the final time is a horizontal translation of the initial condition. The proposed DG methods can

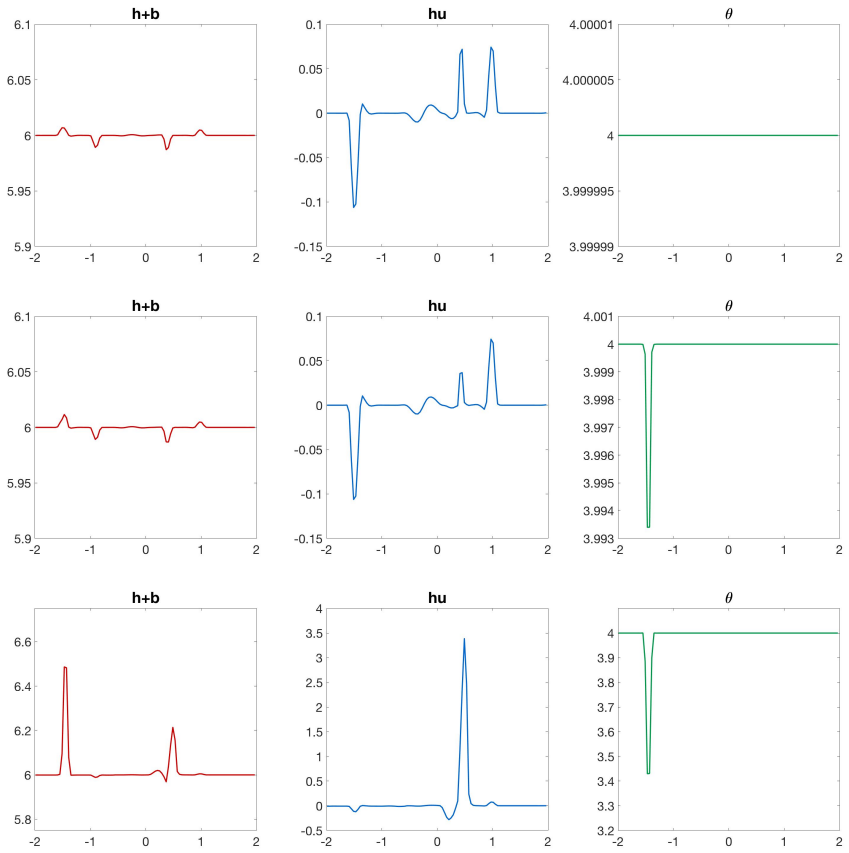


Figure 2.9: Solutions of perturbation tests (2.30) (top row), (2.31) (middle row), and (2.32) (bottom row), using the traditional DG method. In all cases, the traditional DG scheme doesn't handle perturbations as well as the still-water well-balanced scheme. The larger the perturbation, the closer the results of the traditional DG methods are to the well-balanced method.

capture the discontinuity very well even on a coarse mesh of 200 uniform cells.

2.4 Moving-Water Well-Balanced DG Methods

The objective of this section is to present well-balanced methods which maintain the general moving steady state (2.8). Compared with the still-water equilibrium, the

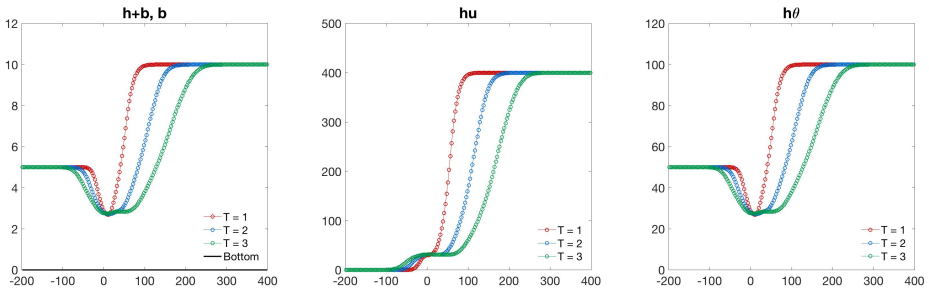


Figure 2.10: Numerical solutions of the well-balanced DG methods for (2.33) at various times with 200 uniform cells.

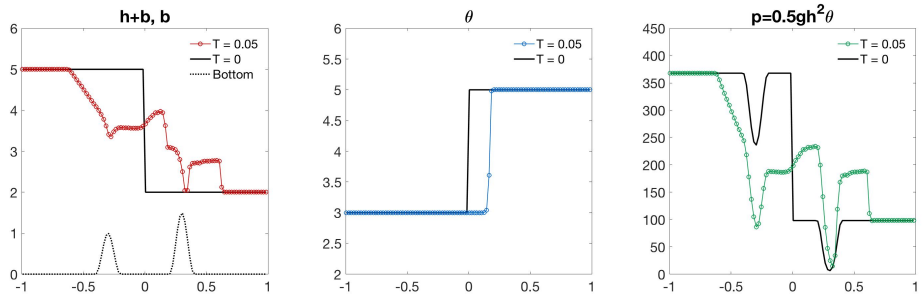


Figure 2.11: Initial conditions and numerical solutions of test (2.35) at time $t = 0.05$ using 200 uniform cells. The total water height $h+b$, bottom topography b, potential temperature field θ, and pressure $p = \frac{1}{2}gh^2\theta$ are plotted. The result for θ is a horizontal translation of the initial condition.

moving-water steady state is more complex, therefore, special care is provided to the recovery of the well-balanced states, the source term approximation, and the construction of well-balanced numerical fluxes. We also show how the well-balanced methods for the general moving steady state can be extended to the isobaric (2.6) and constant water height (2.7) steady states.

2.4.1 Numerical Initial Conditions

Recovery of the well-balanced states from the numerical initial conditions in the piecewise polynomial space $\mathbb{V}_\tau^k$ can be a challenging task. Usually, the initial conditions of the modal DG method are taken to be the L^2 projection of the true initial conditions $\boldsymbol{U}_0$, which works well for the lake-at-rest still-water preserving methods. However, the projected polynomial, denoted as $\boldsymbol{U}_{0,\tau}$, may not be in the equilibrium state when the moving-water equilibrium is considered. As a result, the cell boundary values $\boldsymbol{U}^{\pm}_{\tau,j+\frac{1}{2}}$ may not be in equilibrium. This is problematic because the cell boundary values are used to compute the numerical fluxes and thus increases the difficulty in designing a well-balanced scheme.

This difficulty has been observed in [150] in which well-balanced methods were designed for the SWEs with moving-water equilibrium state. It was resolved there by defining the well-balanced states as the solutions of nonlinear equations and then solving them using Newton's method, which could be complicated. In this work, we propose a different and much simpler method to define the numerical initial conditions and to recover the well-balanced states, thanks to the flexibility of the DG methods.

The initial conditions of the DG method should be polynomials that approximate the exact solutions with enough accuracy. Thus we introduce a new projection (known as the Radau projection) of the initial condition by defining $\mathbb{P}_\tau\omega$ to be a projection of ω into $\mathbb{V}_\tau^k$, such that

$$\int_{I_j} \mathbb{P}_\tau\omega v \, dx = \int_{I_j} \omega v \, dx \tag{2.36}$$

for any $v \in P^{k-1}$ on I_j, and

$$(\mathbb{P}_\tau\omega)(x^-_{j+\frac{1}{2}}) = \omega(x_{j+\frac{1}{2}}), \tag{2.37}$$

at the right boundary value $x_{j+\frac{1}{2}}$ of the cell I_j. The polynomial $\mathbb{P}_\tau\omega$ for each cell I_j can be determined by solving a local linear algebra problem of the size $(k+1)\times(k+1)$ derived from the discretized versions of (2.36) and (2.37).

We define the projection of the initial condition $\boldsymbol{U}_{0,\tau}$ and the projection of the bottom function $b_\tau(x)$ to be

$$\boldsymbol{U}_\tau^0(x) = \mathbb{P}_\tau \boldsymbol{U}_0(x), \qquad b_\tau(x) = \mathbb{P}_\tau b(x), \tag{2.38}$$

and it can be shown that

$$\boldsymbol{U}_\tau^0(x_{j+\frac{1}{2}}^-) = \boldsymbol{U}_0(x_{j+\frac{1}{2}}), \qquad b_\tau(x_{j+\frac{1}{2}}^-) = b(x_{j+\frac{1}{2}}), \qquad \text{for all } j. \tag{2.39}$$

Thus, at the right boundary point of each cell I_j, the equilibrium states (2.8) are recovered using the piecewise polynomial projections of the exact solutions, i.e.,

$$\left(\frac{u_\tau^2}{2} + g\theta_\tau(h_\tau + b_\tau)\right)\left(x_{j+\frac{1}{2}}^-\right) = constant, \qquad \text{for all } j.$$

Notice that the numerical initial condition is only in perfect equilibrium at the right boundary point of each cell.

Remark 6. *The choice of the projection $\mathbb{P}_\tau$ is not unique. Instead of requiring the projected function match the original function at the right end point $x_{j+\frac{1}{2}}$ as in* (2.37)*, we could choose any other point in the cell I_j, except the center x_j. Such projection with the choice of x_j has been presented in [95]. However, we noticed that projection is only optimal when the polynomial degree k is even, and it is suboptimal when k is odd.*

2.4.2 Conservative and Equilibrium Variables

We denote the moving-water equilibrium variables from (2.8) as:

$$\boldsymbol{V} = \begin{pmatrix} E \\ m \\ \theta \end{pmatrix} = \begin{pmatrix} \frac{u^2}{2} + g\theta(h+b) \\ hu \\ \theta \end{pmatrix}, \tag{2.40}$$

which become constants at the steady state (2.8). It is necessary to transform the conservative variables $\boldsymbol{U}$ into equilibrium variables $\boldsymbol{V}$, and vice versa in the process of maintaining the well-balanced property. Given $\boldsymbol{U}$ and the bottom function b, the equilibrium variables can be easily computed, and we denote it by $\boldsymbol{V} = \boldsymbol{V}(\boldsymbol{U}, b)$. On the other hand, suppose $\boldsymbol{V}$ and the bottom function b are given, we can evaluate $\boldsymbol{U} = \boldsymbol{U}(\boldsymbol{V}, b)$ in the following way. Let us rewrite E as

$$E = \frac{m^2}{2h^2} + g\theta(h+b), \tag{2.41}$$

which leads to the cubic polynomial

$$\frac{m^2}{2} + (g\theta b - E)h^2 + (g\theta)h^3 = 0. \tag{2.42}$$

To recover the water height h, we find the correct root of this cubic polynomial. The three roots of a cubic polynomial can be found analytically. If the polynomial returns three real-valued roots, one of them is negative, and the other two correspond to the subsonic and supersonic cases. We choose $h(\boldsymbol{V}, b)$ as the root that is closest to $h_\tau(\hat{x}_i)$, where $\hat{x}_i$ is either a quadrature point or a cell-boundary value. In the other case when the polynomial has one real-valued root and two complex-valued roots, the only real-valued root is negative and we choose $h(V, b)$ to be the real part of the complex roots. Once h is obtained, m and $h\theta$ can be easily recovered from V.

Next, we propose to decompose the solution $\boldsymbol{U}_\tau$, into the reference equilibrium state $\boldsymbol{U}_\tau^e$ and the fluctuation part $\boldsymbol{U}_\tau^f$, with the expectation that $\boldsymbol{U}_\tau^f$ reduces to 0 at the moving-water equilibrium (2.8). Note that this decomposition will be computed not only for the initial conditions, but also at each time step. In each computational cell I_j, the equilibrium state $\boldsymbol{U}_\tau^e(x)$ can be computed from the constant equilibrium variables

$$\hat{\boldsymbol{V}}_j = \begin{pmatrix} \hat{E}_j \\ \hat{m}_j \\ \hat{\theta}_j \end{pmatrix} = \begin{pmatrix} E_\tau(x^-_{j+\frac{1}{2}}) \\ m_\tau(x^-_{j+\frac{1}{2}}) \\ \theta_\tau(x^-_{j+\frac{1}{2}}) \end{pmatrix}, \tag{2.43}$$

and the bottom function b_τ in the form of

$$\boldsymbol{U}_\tau^e(x) = \begin{pmatrix} h_\tau^e(x) \\ (hu)_\tau^e(x) \\ (h\theta)_\tau^e(x) \end{pmatrix} := \mathbb{P}_\tau \boldsymbol{U}(\hat{\boldsymbol{V}}_j, b_\tau). \tag{2.44}$$

Unlike the still-water case in (2.13), the functions $\boldsymbol{U}(\hat{V}_j, b_\tau)$ may not be piecewise polynomials as a result of the nonlinear mapping, even though $\hat{\boldsymbol{V}}_j$ is constant and $b_\tau \in \mathbb{V}_\tau^k$. Therefore, the same projection $\mathbb{P}_\tau$ is used to project them into the DG space $\mathbb{V}_\tau^k$. Lastly, we can decompose the numerical solution $\boldsymbol{U}_\tau$ as follows:

$$\boldsymbol{U}_\tau = \boldsymbol{U}_\tau^e + \boldsymbol{U}_\tau^f, \tag{2.45}$$

where $\boldsymbol{U}_\tau^f = \boldsymbol{U}_\tau - \boldsymbol{U}_\tau^e \in \mathbb{V}_\tau^k$. If the initial condition $\boldsymbol{U}_0(x)$ is in the moving-water equilibrium (2.8), the reference equilibrium state $\boldsymbol{U}_\tau^e$ (constructed in the way above) is identical to the solution $\boldsymbol{U}_\tau$, and therefore $\boldsymbol{U}_\tau^f = 0$.

2.4.3 Well-Balanced Numerical Fluxes

The semi-discrete moving-water well-balanced DG method for (2.1) is defined as follows: we seek the DG solution $\boldsymbol{U}_\tau$ satisfying

$$\int_{I_j} \partial_t(\boldsymbol{U}_\tau)v\ dx - \int_{I_j} \boldsymbol{F}(\boldsymbol{U}_\tau)\partial_x v\ dx + \hat{\boldsymbol{F}}^l_{j+\frac{1}{2}} v^-_{j+\frac{1}{2}} - \hat{\boldsymbol{F}}^r_{j-\frac{1}{2}} v^+_{j\frac{1}{2}} = \int_{I_j} \boldsymbol{S}(\boldsymbol{U}_\tau, b_\tau)v\ dx, \quad (2.46)$$

for any test function $v(x) \in \mathbb{V}^k_\tau$. In order to determine the well-balanced numerical fluxes, we extend the hydrostatic reconstruction approach presented in Section 2.2.2 for the still-water equilibrium. After computing the boundary values $\boldsymbol{U}^\pm_{\tau,j+\frac{1}{2}}$ at the time step t^n, we again set

$$b^*_{\tau,j\pm\frac{1}{2}} = \max\left(b^+_{\tau,j\pm\frac{1}{2}}, b^-_{\tau,j\pm\frac{1}{2}}\right). \quad (2.47)$$

However, other generalizations of the hydrostatic reconstruction could also be used. Next, the height function at the cell interface can be redefined as:

$$\begin{aligned} h^{*,-}_{\tau,j+\frac{1}{2}} &= \max\left(0, h\left(\hat{V}_j, b^*_{\tau,j+\frac{1}{2}}\right) + h^{f,-}_{\tau,j+\frac{1}{2}}\right), \\ h^{*,+}_{\tau,j+\frac{1}{2}} &= \max\left(0, h\left(\hat{V}_{j+1}, b^*_{\tau,j+\frac{1}{2}}\right) + h^{f,+}_{\tau,j+\frac{1}{2}}\right), \end{aligned} \quad (2.48)$$

where the values $\hat{V}_j$ are given in (2.43) and h^f_τ is given in (2.45). This results in following updated boundary values of $\boldsymbol{U}$:

$$\boldsymbol{U}^{*,\pm}_{\tau,j+\frac{1}{2}} = \begin{pmatrix} h^{*,\pm}_{\tau,j+\frac{1}{2}} \\ m^\pm_{\tau,j+\frac{1}{2}} \\ h^{*,\pm}_{\tau,j+\frac{1}{2}}\theta^\pm_{\tau,j+\frac{1}{2}} \end{pmatrix}. \quad (2.49)$$

Finally, the well-balanced numerical fluxes can be computed:

$$\begin{aligned} \hat{\boldsymbol{F}}^l_{j+\frac{1}{2}} &= \boldsymbol{f}\left(\boldsymbol{U}^{*,-}_{\tau,j+\frac{1}{2}}, \boldsymbol{U}^{*,+}_{\tau,j+\frac{1}{2}}\right) + \boldsymbol{F}\left(\boldsymbol{U}^-_{\tau,j+\frac{1}{2}}\right) - \boldsymbol{F}\left(\boldsymbol{U}^{*,-}_{\tau,j+\frac{1}{2}}\right), \\ \hat{\boldsymbol{F}}^r_{j-\frac{1}{2}} &= \boldsymbol{f}\left(\boldsymbol{U}^{*,-}_{\tau,j-\frac{1}{2}}, \boldsymbol{U}^{*,+}_{\tau,j-\frac{1}{2}}\right) + \boldsymbol{F}\left(\boldsymbol{U}^+_{\tau,j-\frac{1}{2}}\right) - \boldsymbol{F}\left(\boldsymbol{U}^{*,+}_{\tau,j-\frac{1}{2}}\right), \end{aligned} \quad (2.50)$$

where $\boldsymbol{f}(a,b)$ is a consistent numerical flux, such as the Lax-Friedrichs flux defined in (1.18). Again, at the moving-water equilibrium (2.8), we have $\hat{\boldsymbol{V}}_j = constant$ and $h^f_\tau = 0$, therefore, $h^{*,+}_{\tau,j+\frac{1}{2}} = h^{*,-}_{\tau,j+\frac{1}{2}}$ following the definition in (2.48). This leads to the continuity of $\boldsymbol{U}^*_\tau$ at cell interfaces, i.e. $\boldsymbol{U}^{*,+}_{\tau,j+\frac{1}{2}} = \boldsymbol{U}^{*,-}_{\tau,j+\frac{1}{2}}$. As a result, we can show that

$$\hat{\boldsymbol{F}}^l_{j+\frac{1}{2}} = \boldsymbol{F}(\boldsymbol{U}^-_{j+\frac{1}{2}}), \qquad \hat{\boldsymbol{F}}^r_{j-\frac{1}{2}} = \boldsymbol{F}(\boldsymbol{U}^+_{j-\frac{1}{2}}), \tag{2.51}$$

due to the consistency of the numerical flux $\boldsymbol{f}(a,b)$.

Remark 7. *The recovery of h from the equilibrium variable V via solving a cubic polynomial is needed twice at each time step, namely in* (2.44) *and* (2.48) *described above. Numerically, the cubic polynomial computations allot for approximately $10-12\%$ of total computational time. One could further reduce the computational cost if redefining the variable b^* in* (2.47) *to*

$$b^*_{\tau,j\pm\frac{1}{2}} = b^-_{\tau,j\pm\frac{1}{2}}, \tag{2.52}$$

which matches the definition of the Radau projection in (2.37)*. Therefore, $h(\hat{\boldsymbol{V}}_j, b^*_{\tau,j+\frac{1}{2}})$ reduces to $h^-_{\tau,j+\frac{1}{2}}$, and the updated cell boundary value $h^{*,\pm}$ in* (2.48) *becomes*

$$h^{*,-}_{\tau,j+\frac{1}{2}} = \max\left(0, h^-_{\tau,j+\frac{1}{2}} + h^{f,-}_{\tau,j+\frac{1}{2}}\right), \qquad h^{*,+}_{\tau,j+\frac{1}{2}} = \max\left(0, h^-_{\tau,j+\frac{1}{2}} + h^{f,+}_{\tau,j+\frac{1}{2}}\right), \tag{2.53}$$

which now does not involve the step to solve the cubic polynomial.

2.4.4 The Source Term Approximation

To maintain the well-balanced property, we introduce the following way to decompose the source term, by directly extending the idea in Section 2.2.3 for the still-water

equilibrium. We first decompose the source term into the equilibrium and fluctuation parts

$$\int \boldsymbol{S}(\boldsymbol{U}_\tau, b_\tau)v \; dx = \int \boldsymbol{S}(\boldsymbol{U}^e_\tau, b_\tau)v \; dx + \int \boldsymbol{S}(\boldsymbol{U}^f_\tau, b_\tau)v \; dx. \tag{2.54}$$

As before, we use a quadrature rule to compute the second term on the right hand side. For the first term, we notice that the equality

$$\begin{aligned} &\int_{I_j} \boldsymbol{S}\Big(\boldsymbol{U}(\hat{V}_j, b_\tau), b_\tau\Big)v \; dx \\ &\quad = -\int_{I_j} \boldsymbol{F}\Big(\boldsymbol{U}(\hat{V}_j, b_\tau)\Big)v_x \; dx + \boldsymbol{F}\Big(\boldsymbol{U}(\hat{V}_j, b_\tau)(x^-_{j+\frac{1}{2}})\Big)v^-_{j+\frac{1}{2}} - \boldsymbol{F}\Big(\boldsymbol{U}(\hat{V}_j, b_\tau)(x^+_{j-\frac{1}{2}})\Big)v^+_{j-\frac{1}{2}}, \end{aligned} \tag{2.55}$$

holds when the exact integration is used, since $\boldsymbol{U}(\hat{\boldsymbol{V}}_j, b_\tau)$ is the equilibrium solution in the cell I_j. In (2.44), $\boldsymbol{U}^e_\tau$ is defined as the projection of $\boldsymbol{U}(\hat{\boldsymbol{V}}_j, b_\tau)$ with approximation error $O(\tau^{k+1})$. Taking in consideration of this error, the equation (2.55) becomes

$$\int_{I_j} \boldsymbol{S}(\boldsymbol{U}^e_\tau, b_\tau)v \; dx + O(\tau^{k+1}) = -\int_{I_j} \boldsymbol{F}(\boldsymbol{U}^e_\tau)v_x \; dx + \boldsymbol{F}(\boldsymbol{U}^{e,-}_{\tau,j+\frac{1}{2}})v^-_{j+\frac{1}{2}} - \boldsymbol{F}(\boldsymbol{U}^{e,+}_{\tau,j-\frac{1}{2}})v^+_{j-\frac{1}{2}}, \tag{2.56}$$

in which the integrals can also be computed via quadrature rule with the same error $O(\tau^{k+1})$. Unlike the still-water case in Section 2.2.3, the term $\boldsymbol{F}(\boldsymbol{U}^e_\tau)$ is no longer a polynomial, even though $\boldsymbol{U}^e_\tau$ is, and we cannot compute these integrals exactly with a sufficiently accuracy quadrature rule. To this end, the source term can be evaluated using the following form:

$$\int_{I_j} \boldsymbol{S}(\boldsymbol{U}_\tau, b_\tau)v \; dx = -\int_{I_j} \boldsymbol{F}(\boldsymbol{U}^e_\tau)v_x \; dx + \boldsymbol{F}(\boldsymbol{U}^{e,-}_{\tau,j+\frac{1}{2}})v^-_{j+\frac{1}{2}} - \boldsymbol{F}(\boldsymbol{U}^{e,+}_{\tau,j-\frac{1}{2}})v^+_{j-\frac{1}{2}} + \int_{I_j} \boldsymbol{S}(\boldsymbol{U}^f_\tau, b_\tau)v \; dx. \tag{2.57}$$

2.4.5 Slope Limiter

When the solution contains discontinuities, a slope limiter procedure may be necessary after each inner stage of the Runge-Kutta time stepping procedure. We use the

standard total variation bounded (TVB) slope limiter [138] presented below. The TVB corrected minmod function is defined by

$$\tilde{m}(a_1, ..., a_l) = \begin{cases} a_1, & \text{if } |a_1| \leq M\tau^2, \\ m(a_1, ..., a_l), & \text{otherwise,} \end{cases} \tag{2.58}$$

where M is the TVB parameter, and the minmod function $m(a_1, ..., a_l)$ is defined as:

$$m(a_1, ..., a_l) = \begin{cases} s \min(|a_1|, ..., |a_l|), & \text{if } s = \text{sign}(a_1) = ... = \text{sign}(a_l), \\ 0, & \text{otherwise .} \end{cases} \tag{2.59}$$

The slope limiter procedure involves two steps. First, one must check if any limiting in needed in a specific cell. If the corrected minmod function returns the first argument, i.e.,

$$\begin{aligned} \tilde{m}\Big(\boldsymbol{U}_\tau(x^-_{j+\frac{1}{2}}) - \bar{\boldsymbol{U}}_{\tau,j}, \bar{\boldsymbol{U}}_{\tau,j} - \bar{\boldsymbol{U}}_{\tau,j-1}, \bar{\boldsymbol{U}}_{\tau,j+1} - \bar{\boldsymbol{U}}_{\tau,j}\Big) &= \boldsymbol{U}_\tau(x^-_{j+\frac{1}{2}}) - \bar{\boldsymbol{U}}_{\tau,j}, \\ \tilde{m}\Big(\bar{\boldsymbol{U}}_{\tau,j} - \boldsymbol{U}_\tau(x^+_{j-\frac{1}{2}}), \bar{\boldsymbol{U}}_{\tau,j} - \bar{\boldsymbol{U}}_{\tau,j-1}, \bar{\boldsymbol{U}}_{\tau,j+1} - \bar{\boldsymbol{U}}_{\tau,j}\Big) &= \bar{\boldsymbol{U}}_{\tau,j} - \boldsymbol{U}_\tau(x^+_{j-\frac{1}{2}}), \end{aligned} \tag{2.60}$$

with $\bar{\boldsymbol{U}}_{\tau,j}$ standing for the cell average of $\boldsymbol{U}_\tau$ in the cell I_j, the limiting is not necessary in this cell. Otherwise, we need to apply the limiter to all the variables in that cell, which is the second step of the slope limiter procedure. We define the modified cell boundary values to be

$$\begin{aligned} \boldsymbol{U}^{(\text{mod})}_\tau(x^-_{j+\frac{1}{2}}) &= \bar{\boldsymbol{U}}_{\tau,j} + \tilde{m}\Big(\boldsymbol{U}_\tau(x^-_{j+\frac{1}{2}}) - \bar{\boldsymbol{U}}_{\tau,j}, \bar{\boldsymbol{U}}_{\tau,j} - \bar{\boldsymbol{U}}_{\tau,j-1}, \bar{\boldsymbol{U}}_{\tau,j+1} - \bar{\boldsymbol{U}}_{\tau,j}\Big), \\ \boldsymbol{U}^{(\text{mod})}_\tau(x^+_{j-\frac{1}{2}}) &= \bar{\boldsymbol{U}}_{\tau,j} - \tilde{m}\Big(\bar{\boldsymbol{U}}_{\tau,j} - \boldsymbol{U}_\tau(x^+_{j-\frac{1}{2}}), \bar{\boldsymbol{U}}_{\tau,j} - \bar{\boldsymbol{U}}_{\tau,j-1}, \bar{\boldsymbol{U}}_{\tau,j+1} - \bar{\boldsymbol{U}}_{\tau,j}\Big), \end{aligned} \tag{2.61}$$

A P^2 polynomial that preserves the original cell average in I_j can then be recovered from the updated cell boundary values (2.61) and the cell average $\bar{\boldsymbol{U}}_{\tau,j}$.

This limiting procedure may not preserve the moving-water equilibrium state (2.8). Therefore, to maintain the well-balanced state when using the slope limiter, we perform the

first step (2.60) to determine if limiting is necessary in each cell, based on the fluctuation part of the variables $\boldsymbol{U}_\tau^f$. If limiting is required in a specific cell, the slope limiting procedure is performed on the conservation variables $\boldsymbol{U}_\tau$. If the steady state solution is reached, the fluctuation part satisfies $\boldsymbol{U}_\tau^f = 0$, hence no TVB limiter is needed in any cell. Therefore the well-balanced property will not be affected by the limiter procedure. Unlike in [150], we cannot use the equilibrium variable $\boldsymbol{V}_\tau$ in the first step (2.60), due to the possibility of the cell average $\bar{\boldsymbol{V}}_\tau$ not being in equilibrium even at the moving-water steady state.

2.4.6 Well-Balanced Property

Proposition 8. *The proposed DG scheme (2.46) for the Ripa system* (2.1)*, paired with numerical fluxes (2.50) and source term approximation (2.57), is well-balanced for the smooth moving-water equilibrium (2.8).*

Proof. : Suppose the initial data are moving-water steady state equilibria. The special projection $\mathbb{P}_\tau$ ensures that the equilibrium state is maintained at the cell boundary $x_{j+1/2}^-$ for all j for the numerical initial conditions. Therefore, from the definition of (2.43), we have $\hat{\boldsymbol{V}}_j = constant$ for all j. This implies the equilibrium part U_τ^e, computed from $\hat{\boldsymbol{V}}_j$ and b_τ in (2.44), is equivalent to $\boldsymbol{U}_\tau$ and that $\boldsymbol{U}_\tau^f = 0$. The source term approximation (2.57) becomes

$$\int_{I_j} \boldsymbol{S}(\boldsymbol{U}_\tau, b_\tau) v \, dx = -\int_{I_j} \boldsymbol{F}(\boldsymbol{U}_\tau) v_x \, dx + \boldsymbol{F}(\boldsymbol{U}^-_{\tau,j+\frac{1}{2}}) v^-_{j+\frac{1}{2}} - \boldsymbol{F}(\boldsymbol{U}^+_{\tau,j-\frac{1}{2}}) v^+_{j-\frac{1}{2}} \tag{2.62}$$

At the same time, since $\boldsymbol{U}_\tau^f = 0$, the modified cell boundary values satisfy

$$\boldsymbol{U}^{*,-}_{\tau,j+\frac{1}{2}} = \begin{pmatrix} \max\left(0, h(\hat{\boldsymbol{V}}_j, b^*_{j+\frac{1}{2}})\right) \\ \hat{m}^-_{j+\frac{1}{2}} \\ \max\left(0, h(\hat{\boldsymbol{V}}_j, b^*_{j+\frac{1}{2}})\right)\hat{\theta}^-_{j+\frac{1}{2}} \end{pmatrix} = \begin{pmatrix} \max\left(0, h(\hat{\boldsymbol{V}}_{j+1}, b^*_{j+\frac{1}{2}})\right) \\ \hat{m}^+_{j+\frac{1}{2}} \\ \max\left(0, h(\hat{\boldsymbol{V}}_{j+1}, b^*_{j+\frac{1}{2}})\right)\hat{\theta}^+_{j+\frac{1}{2}} \end{pmatrix} = \boldsymbol{U}^{*,+}_{\tau,j+\frac{1}{2}}. \tag{2.63}$$

Due to the consistency of the Lax-Friedrichs flux and (2.63), it can be shown that

$$\hat{\boldsymbol{F}}^l_{j+\frac{1}{2}} = \boldsymbol{F}(\boldsymbol{U}^-_{\tau,j+\frac{1}{2}}), \qquad \hat{\boldsymbol{F}}^r_{j-\frac{1}{2}} = \boldsymbol{F}(\boldsymbol{U}^+_{\tau,j-\frac{1}{2}}), \tag{2.64}$$

and the flux terms at the left hand side of (2.46) reduce to

$$-\int_{I_j} \boldsymbol{F}(\boldsymbol{U}_\tau)\partial_x v \; dx + \boldsymbol{F}(\boldsymbol{U}^-_{\tau,j+\frac{1}{2}})v^-_{j+\frac{1}{2}} - \boldsymbol{F}(\boldsymbol{U}^+_{\tau,j-\frac{1}{2}})v^+_{j-\frac{1}{2}}.$$

It thus follows that the flux terms exactly balance the source term approximation, and the well-balanced property is proven. ■

We finish this section with some remarks about the moving-water equilibrium preserving well-balanced RKDG method.

Remark 9. *When the bottom topography is flat (i.e., $b(x) =$ constant C), the traditional DG methods are recovered from our well-balanced DG scheme, that is, the source term approximation reduces to 0 exactly and the left and right numerical fluxes reduce to the original fluxes. First of all, in each cell I_j, $\hat{\boldsymbol{V}}_j$ defined in* (2.43) *is a scalar number. When $b(x) = C$ is a constant, the value of $h(\hat{\boldsymbol{V}}_j, C)$ is also a constant which does not depend on x within each cell I_j. The same observation holds for $\boldsymbol{U}^e_\tau(x) \equiv \mathbb{P}_\tau \boldsymbol{U}(\hat{\boldsymbol{V}}_j, C)$. Therefore, we have*

$$\int_{I_j} \boldsymbol{S}\left(\boldsymbol{U}^e_\tau, b_\tau\right) v \; dx = -\int_{I_j} \boldsymbol{F}\left(\boldsymbol{U}^e_\tau\right) v_x \; dx + \boldsymbol{F}\left(\boldsymbol{U}^{e,-}_{\tau,j+\frac{1}{2}}\right) v^-_{j+\frac{1}{2}} - \boldsymbol{F}\left(\boldsymbol{U}^{e,+}_{\tau,j-\frac{1}{2}}\right) v^+_{j-\frac{1}{2}} = 0.$$

Also, the numerical integral $\int_{I_j} \boldsymbol{S}\left(\boldsymbol{U}_\tau^f, b_\tau\right) v \, dx = 0$, *since* $b_\tau(x) = C$ *leads to* $(b_\tau)_x = 0$. *Together, this implies the source term approximation presented in* (2.57) *is zero.*

Second, we show that the well-balanced numerical flux reduces to the standard DG flux. Note that, when $b(x) = C$ *is a constant,* $b^*_{\tau,j\pm\frac{1}{2}}$ *defined by* (2.47) *or* (2.52) *is also the same constant* C. *Therefore,*

$$h^{*,-}_{\tau,j+\frac{1}{2}} = \max\left(0, h(\hat{\boldsymbol{V}}_j, C) + h^{f,-}_{\tau,j+\frac{1}{2}}\right) = \max\left(0, h^{e,-}_{\tau,j+\frac{1}{2}} + h^{f,-}_{\tau,j+\frac{1}{2}}\right) = h^{-}_{\tau,j+\frac{1}{2}}. \tag{2.65}$$

Similarly, it can be shown that $h^{*,+}_{\tau,j+\frac{1}{2}} = h^{+}_{\tau,j+\frac{1}{2}}$ *and more generally that* $\boldsymbol{U}^{*,\pm}_{\tau,j+\frac{1}{2}} = \boldsymbol{U}^{\pm}_{\tau,j+\frac{1}{2}}$. *Therefore the left and right numerical fluxes defined in* (2.50) *reduce to the original DG fluxes:* $\hat{\boldsymbol{F}}^l_{j+\frac{1}{2}} = \hat{\boldsymbol{F}}_{j+\frac{1}{2}}$, $\hat{\boldsymbol{F}}^r_{j-\frac{1}{2}} = \hat{\boldsymbol{F}}_{j-\frac{1}{2}}$.

Remark 10. *Our well-balanced methods are designed to preserve the moving-water equilibrium (2.8). When applied to the still-water steady state* (2.5), *which is a special case of the moving-water equilibrium with* $m = 0$, *they should automatically preserve this simpler equilibrium. In fact, they reduce to the same still-water well-balanced methods presented in Section 2.2 with the exception of the choice of projection.*

2.4.7 Well-Balanced Methods for the Constant Water Height and Isobaric Equilibria

The proposed framework to balance the moving-water equilibrium can also be extended to preserve the constant water height steady state (2.7) and the isobaric steady state (2.6) of the Ripa model. For each given equilibrium, one well-balanced method to balance this equilibrium can be designed, but we cannot combine these into a unified well-balanced method. Below, we will discuss how to design such scheme briefly with some

necessary details provided.

First, let's consider the constant water height steady state solution (2.7). Let the equilibrium variables be denoted

$$\hat{\boldsymbol{V}}_j = \begin{pmatrix} \hat{h}_j \\ \hat{m}_j \\ \hat{L}_j \end{pmatrix} = \begin{pmatrix} h_\tau\left(x^-_{j+\frac{1}{2}}\right) \\ (hu)_\tau\left(x^-_{j+\frac{1}{2}}\right) \\ \left(b_\tau + \frac{1}{2}h_\tau \ln\theta_\tau\right)\left(x^-_{j+\frac{1}{2}}\right) \end{pmatrix}, \tag{2.66}$$

then the equilibrium parts can be determined as follows:

$$\boldsymbol{U}^e_{\tau,j}(x) = \begin{pmatrix} h^e_{\tau,j}(x) \\ (hu)^e_{\tau,j}(x) \\ (h\theta)^e_{\tau,j}(x) \end{pmatrix} = \begin{pmatrix} \hat{h}_j \\ \hat{m}_j \\ \hat{h}_j \mathbb{P}_\tau\Big(\exp\left(\frac{2}{\hat{h}_j}(\hat{L}_j - b_\tau(x))\right)\Big) \end{pmatrix}, \tag{2.67}$$

where $\mathbb{P}_\tau$ is the same Radau projection described in (2.36) and (2.37). The fluctuation part $\boldsymbol{U}^f_\tau$ is again determined by the decomposition $\boldsymbol{U}_\tau = \boldsymbol{U}^e_\tau + \boldsymbol{U}^f_\tau$. Next, we determine the modified cell interface values as

$$\begin{aligned} (h\theta)^{*,-}_{\tau,j+\frac{1}{2}} &= \max\left(0, \hat{h}_j \exp\left(\frac{2}{\hat{h}_j}\left(\hat{L}_j - b^*_{\tau,j+\frac{1}{2}}\right)\right) + (h\theta)^{f,-}_{j+\frac{1}{2}}\right), \\ (h\theta)^{*,+}_{\tau,j+\frac{1}{2}} &= \max\left(0, \hat{h}_{j+1} \exp\left(\frac{2}{\hat{h}_{j+1}}\left(\hat{L}_{j+1} - b^*_{\tau,j+\frac{1}{2}}\right)\right) + (h\theta)^{f,+}_{j+\frac{1}{2}}\right), \end{aligned} \tag{2.68}$$

where $b^*_{\tau,j+\frac{1}{2}} = \max\left(b^+_{\tau,j+\frac{1}{2}}, b^-_{\tau,j+\frac{1}{2}}\right)$. We let $h^{*,\pm}_{\tau,j+\frac{1}{2}} = h^\pm_{\tau,j+\frac{1}{2}}$ and $(hu)^{*,\pm}_{\tau,j+\frac{1}{2}} = (hu)^\pm_{\tau,j+\frac{1}{2}}$. Then at the steady state we have $\boldsymbol{U}^{*,-}_{\tau,j+\frac{1}{2}} = \boldsymbol{U}^{*,+}_{\tau,j+\frac{1}{2}}$ since h, hu are both constant and $U^f_\tau = 0$ at the steady state. Lastly, let the choices of left and right fluxes as well as the source term approximation be the same as for the moving-water schemes. The numerical methods designed in this way can be shown to preserve the constant water height state exactly.

Secondly, the isobaric steady state solution (2.6) is considered. Let us denote the equilibrium variables as

$$\hat{\boldsymbol{V}}_j = \begin{pmatrix} \hat{m}_j \\ \hat{S}_j \end{pmatrix} = \begin{pmatrix} (hu)_\tau \left(x^-_{j+\frac{1}{2}} \right) \\ h_\tau (h\theta)_\tau \left(x^-_{j+\frac{1}{2}} \right) \end{pmatrix}, \tag{2.69}$$

then the equilibrium parts can be determined as

$$\boldsymbol{U}^e_{\tau,j}(x) = \begin{pmatrix} h^e_{\tau,j}(x) \\ (hu)^e_{\tau,j}(x) \\ (h\theta)^e_{\tau,j}(x) \end{pmatrix} = \begin{pmatrix} h_{\tau,j}(x) \\ \hat{m}_j \\ \mathbb{P}_\tau \left(\frac{\hat{S}_j}{h_{\tau,j}(x)} \right) \end{pmatrix}, \tag{2.70}$$

and the fluctuation part is defined by $\boldsymbol{U}^f_\tau = \boldsymbol{U}_\tau - \boldsymbol{U}^e_\tau$. Now, if we let $h^*_{\tau,j+\frac{1}{2}} = h^{*,\pm}_{\tau,j+\frac{1}{2}} = \max\left(h^-_{\tau,j+\frac{1}{2}}, h^+_{\tau,j+\frac{1}{2}} \right)$ and set $(hu)^{*,\pm}_{\tau,j+\frac{1}{2}} = (hu)^{\pm}_{\tau,j+\frac{1}{2}}$, we can determine the last conservative variable as

$$(h\theta)^{*,-}_{\tau,j+\frac{1}{2}} = \max\left(0, \frac{\hat{S}_j}{h^*_{\tau,j+\frac{1}{2}}} + (h\theta)^{f,-}_{\tau,j+\frac{1}{2}} \right), \qquad (h\theta)^{*,+}_{\tau,j+\frac{1}{2}} = \max\left(0, \frac{\hat{S}_{j+1}}{h^*_{\tau,j+\frac{1}{2}}} + (h\theta)^{f,-}_{\tau,j+\frac{1}{2}} \right). \tag{2.71}$$

The numerical scheme is completed by using the formulations for the source term as well as the left and right fluxes found in the moving-water scheme. It then follows that the scheme exactly preserves the isobaric steady state.

2.5 Numerical Tests for Moving-Water Well-Balanced Methods

In this section, we present numerical results of our moving-water well-balanced DG methods for the one-dimensional Ripa system (2.1), using the well-balanced technique

described in Section 2.4. We implement our scheme using piecewise quadratic polynomials ($k = 2$) in space, paired with the third order TVD Runge-Kutta time discretization (1.19). The CFL number is again taken to be 0.1. The constant M in the TVB limiter is taken to be 0, except for the accuracy test in which no slope limiter was used. The gravitational constant g is fixed to be $9.812m/s^2$. Multiple types of numerical tests have been carried out in this section to demonstrate the capacity of our proposed moving-water well-balanced methods.

2.5.1 Test for Accuracy

In this subsection, we will check the accuracy of our moving-water well-balanced scheme for smooth solutions. The same setup of the initial conditions and boundary conditions, as in Section 2.3.1, is used here. We run the numerical simulation until time $t = 0.02$, while the numerical solution is still smooth. Table 2.3 contains the L^1 errors and numerical orders for the moving-water well-balanced DG method. It is easy to observe that the expected high order accuracy is obtained by the proposed methods.

	h		hu		$h\theta$	
No. of Cells	L^1 Error	Order	L^1 Error	Order	L^1 Error	Order
25	7.3659e-04		6.7798e-03		7.8134e-04	
50	1.1235e-04	2.7129	9.0751e-04	2.9013	1.1063e-04	2.8201
100	1.5781e-05	2.8317	1.1708e-04	2.9544	1.8243e-05	2.6004
200	2.0662e-06	2.9331	1.5041e-05	2.9606	2.7879e-06	2.7101
400	2.5592e-07	3.0132	1.8865e-06	2.9951	3.8607e-07	2.8522

Table 2.3: L^1 errors and orders of accuracy for the test in Section 2.5.1, using the moving-water well-balanced method.

2.5.2 Tests for the Well-Balanced Property

The following tests are chosen to verify that the proposed DG methods preserve the moving-water steady states (2.8) when a non-flat bottom topography $b(x)$ exists. For these examples, the errors are calculated by comparing the numerical results to the initial conditions.

We present the test cases of both sub-critical and trans-critical flows. These tests are extensions of widely used numerical experiments for verifying the performance of numerical schemes for the SWEs in [145, 110, 152, 150]. The bottom function is given by:

$$b(x) = \begin{cases} 0.2 - 0.05(x-10)^2, & \text{if } 8 \leq x \leq 12, \\ 0, & \text{otherwise,} \end{cases} \tag{2.72}$$

for a channel of length 25 meters. Two types of moving-water steady states solutions, corresponding to sub-critical and trans-critical flows, will be investigated below.

(a) *Sub-critical flow*: The initial conditions are given by

$$\begin{cases} m = 4.42\sqrt{5} \\ E = 22.06605 \cdot 5 \\ \theta = 5 \end{cases} \tag{2.73}$$

with the boundary conditions of $m = 4.42\sqrt{5}$ at the upstream and $h = 2$ at the downstream.

(b) *Trans-critical flow*: The initial conditions are given by:

$$\begin{cases} m = 1.53\sqrt{5} \\ E = 11.09098731433671 \cdot 5 \\ \theta = 5 \end{cases} \tag{2.74}$$

with the boundary conditions of $m = 1.53\sqrt{5}$ at the upstream and $h = 0.405737258401203$ at the downstream. The flow moves from subsonic to supersonic at $x = 10$, which occurs at the peak of the bottom topography function.

For both test cases, the moving-water steady state solutions should be preserved exactly. We run both cases until time the stopping time of $t = 1$, using a coarse mesh of 200 uniform cells and a finer mesh of 800 cells for comparison. The L^1 and L^∞ errors in Table 2.4 demonstrate the well-balanced property is maintained up to round-off error when using the moving-water preserving method. The numerical results are presented in Figures 2.12 and 2.14. For comparison, we also compute these moving-water steady states using the still-water well-balanced methods presented in Section 2.2. Figures 2.13 and 2.15 display the difference between the numerical solution and the initial conditions. It can be seen from the plots that the moving-water steady states are not preserved by the still-water preserving method.

Test	Error Type	h	hu	$h\theta$
(2.73)	L^1 Error	3.9850e−13	6.0707e−13	4.0459e−13
	L^∞ Error	1.5654e−13	4.2100e−13	1.5965e−13
(2.74)	L^1 Error	7.2879e−14	3.0429e−13	7.2849e−14
	L^∞ Error	8.0269e−14	1.8407e−13	7.6161e−14

Table 2.4: L^1 and L^∞ errors for the well-balanced tests of different moving-water equilibria, at time $t = 1$ with 200 uniform cells.

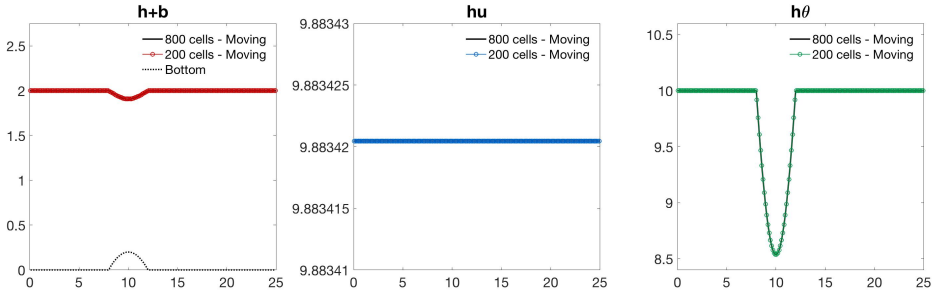


Figure 2.12: Numerical solutions for sub-critical flow (2.73) at time $t = 1$ with 200 and 800 uniform cells. The steady state is preserved for both mesh sizes when the moving-water well-balanced scheme is used.

2.5.3 Tests of Small Perturbations

The tests in this subsection are selected to demonstrate that small perturbations to the moving-water steady states are well captured by the proposed moving-water well-balanced DG scheme. Similar tests have been presented in [159, 150] for the SWEs. We will consider two perturbation sizes to the sub-critical and trans-critical moving-water steady states.

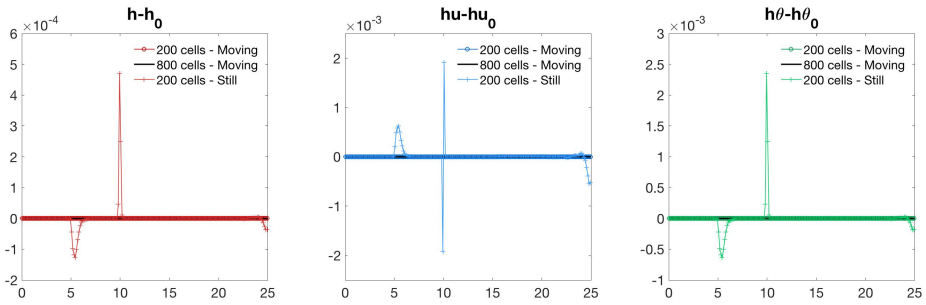


Figure 2.13: The difference between the solutions $h, hu, h\theta$ for the sub-critical flow problem (2.73) at time $t = 1$ and the corresponding steady state solutions. Both the moving-water and still-water scheme results are plotted.

The initial conditions are given by the sub-critical and trans-critical flow (2.73) and (2.74) in Section 2.5.2, with an added perturbation to h and $h\theta$ in the interval $[5.75, 6.25]$. That is, if we denote the moving-water steady state initial conditions in Section 2.5.2 as $\big(h_{eq}, (hu)_{eq}, (h\theta)_{eq}\big)$, the initial conditions of the perturbed tests are given by

(a) *Larger Perturbation:*

$$\big(h, (hu), (h\theta)\big)(x, 0) = \big(h_{eq}, (hu)_{eq}, (h\theta)_{eq}\big)(x, 0) + [0.05, 0, 0.25]_{\chi_{[5.75,6.25]}}. \tag{2.75}$$

(b) *Smaller Perturbation:*

$$\big(h, (hu), (h\theta)\big)(x, 0) = \big(h_{eq}, (hu)_{eq}, (h\theta)_{eq}\big)(x, 0) + [0.0001, 0, 0.0005]_{\chi_{[5.75,6.25]}}. \tag{2.76}$$

The value $\chi_{[5.75,6.25]}$ is defined to be 1 in the region $[5.75, 6.25]$ and 0 everywhere else in the domain.

It is expected that the perturbation will split into two waves traveling in opposite directions away from their point of origin. We run the test until the stopping time of $t = 0.75$. At this time, the downstream moving wave will have passed the bump of the

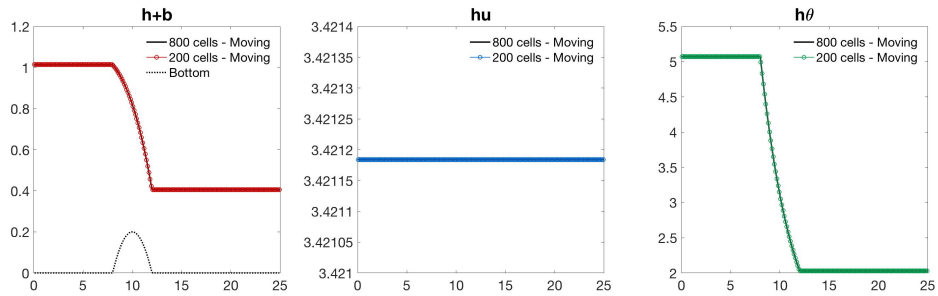


Figure 2.14: Numerical solutions for trans-critical flow (2.74) at time $t = 1$ with 200 and 800 uniform cells. The steady state is preserved for both mesh sizes even with the flow changing from sub-critical to super-critical when the moving-water well-balanced scheme is implemented.

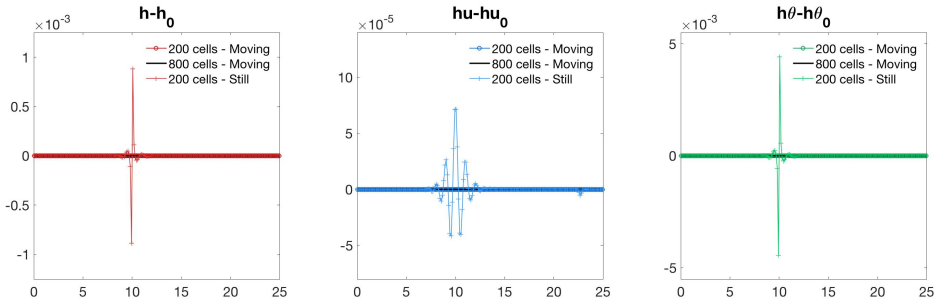


Figure 2.15: Same as Figure 2.13, except for the trans-critical flow problem (2.74).

bottom topography function. The numerical results for the large perturbation of the sub-critical flow problem are shown in Figures 2.16 and 2.17, where we compare the results of moving-water well-balanced DG method on mesh sizes of 200 and 800 uniform cells. One can observe that the propagation of these small perturbations is well captured by our methods, and there are no visible numerical oscillations in the solutions. We demonstrate the numerical results of the larger perturbation for the trans-critical flow problem in Figures

2.18 and 2.19, and similar observations can be found.

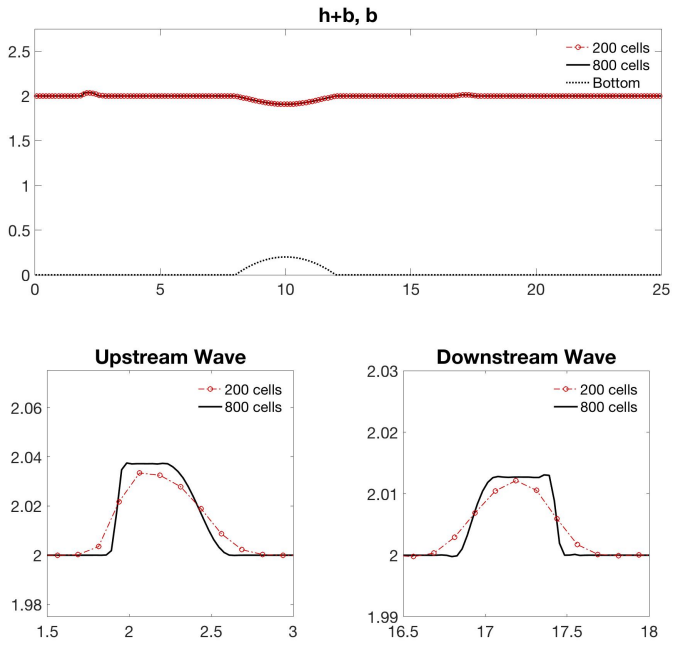


Figure 2.16: Water height h of the well-balanced methods for the larger perturbation test of sub-critical flow (2.73) at time $t = 0.75$ with mesh sizes of 200 and 800 uniform cells. The bottom row contains zoomed in images of the waves of h.

Next, we present the numerical results from the smaller perturbation tests, and compare the performance of the moving-water and still-water preserving schemes on a mesh of 200 uniform cells. Figure 2.20 contains plots of the difference between the solutions $h, hu, h\theta$ at the final time 0.75 and the corresponding steady states for the sub-critical flow problem. It can be seen that the still-water preserving DG scheme does not handle perturbations to the moving-water steady state nearly as well as the moving-water preserving scheme. Similar results have been observed for the trans-critical flow problem, see Fig-

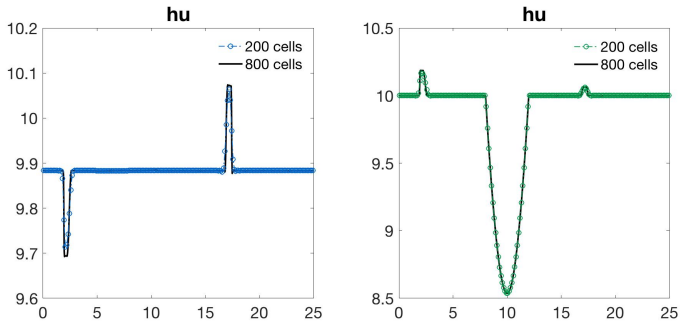


Figure 2.17: Numerical solutions of hu and $h\theta$ for the larger perturbation test of sub-critical flow problem (2.73) at time $t = 0.75$ with mesh sizes of 200 and 800 uniform cells.

ure 2.21. These results demonstrate the importance of moving-water preserving method in capturing the propagation of these small perturbations.

2.5.4 Tests for Riemann Problems

In this subsection, we consider three Riemann problems that contain discontinuities in the initial conditions, to demonstrate the performance of our proposed methods when the solutions contain discontinuity.

First, we consider a problem with no initial velocity, a flat bottom ($b(x) = 0$), and the following initial conditions for $x \in [-1, 1]$:

$$h(x,0) = \begin{cases} 5, & \text{when } x < 0, \\ 1, & \text{when } x \geq 0, \end{cases} \qquad u(x,0) = 0, \qquad \theta(x,0) = \begin{cases} 3, & \text{when } x < 0, \\ 5, & \text{when } x \geq 0. \end{cases} \tag{2.77}$$

Transmissive boundary conditions are employed for all variables. We run the simulation until time $t = 0.04$, and both shock and rarefaction waves appear in this test.

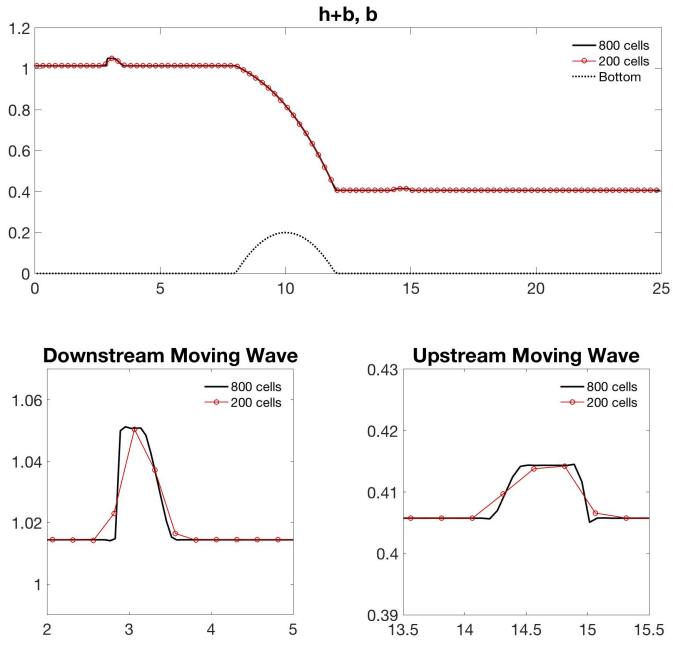


Figure 2.18: Water height h of the well-balanced methods for the larger perturbation test of trans-critical flow (2.74) at time $t = 0.75$ with mesh sizes of 200 and 800 uniform cells. The bottom row contains zoomed in images of the waves of h.

Figure 2.22 displays the numerical results of our methods with 200 uniform cells, and we also include the numerical results of 1600 cells as the "reference" solutions for comparison. One can observe that the numerical solutions agree well with the "reference" solutions, and both the shock and rarefaction wave are captured well by the moving-water well-balanced DG methods.

Second, we consider a dam break problem over a flat bottom, in which all initial conditions are non-zero. The initial conditions in the computational domain $[-1, 1]$ are set

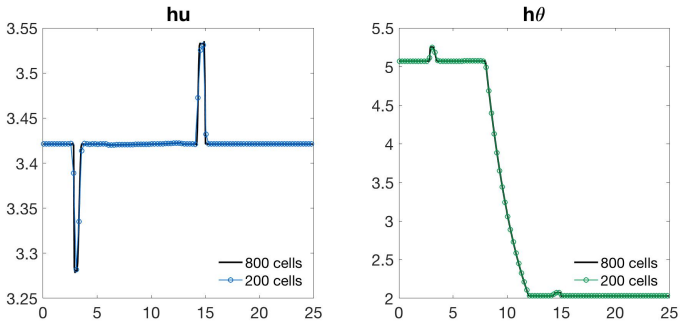


Figure 2.19: Numerical solutions of hu and $h\theta$ for the larger perturbation test of transcritical flow problem (2.74) at time $t = 0.75$ with mesh sizes of 200 and 800 uniform cells.

as

$$\big(h(x,0), u(x,0), \theta(x,0)\big) = \begin{cases} (2, 0.5, 1), & \text{when } |x| < 0.5, \\ (1, 0.75, 1.55), & \text{otherwise,} \end{cases} \tag{2.78}$$

with $b(x) = 0$. Transmissive boundary conditions are employed for all variables. We run the simulation until time $t = 0.075$, and six waves including both shock and rarefaction waves appear. Figure 2.23 displays the numerical results of our methods with 200 uniform cells, and the "reference" solutions obtained with refined 1600 uniform cells. Again, one can observe that the numerical solutions agree well with the "reference" solutions, and all these waves are captured well by the moving-water well-balanced DG methods.

Lastly, we consider a dam break problem over a non-flat and discontinuous bottom topography. We set our computational domain to be $[0, 600]$ in which the bottom function

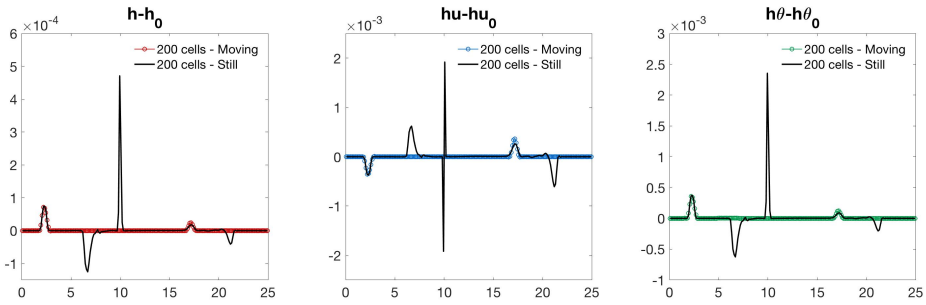


Figure 2.20: The difference between the solutions $h, hu, h\theta$ for the smaller perturbation of the sub-critical flow problem (2.73) at time 0.75 and the corresponding steady state solutions. A uniform mesh of 200 cells has been used for both the still-water and moving-water schemes.

and initial conditions are defined as:

$$b(x) = \begin{cases} 8, & \text{when } |x - 300| < 75, \\ 0, & \text{otherwise}, \end{cases} \tag{2.79}$$

and

$$\big(h, u, \theta\big)(x, 0) = \begin{cases} \big(20 - b(x), 1, 10\big), & \text{when } x < 300, \\ \big(15 - b(x), 5, 5\big), & \text{when } x \geq 300. \end{cases} \tag{2.80}$$

Transmissive boundary conditions are employed for all variables. The simulation is ran until the stopping time $t = 3$, and the solution demonstrate a complicated structure, due to the interaction of the waves with the discontinuous bottom function. The conservative variables $(h, hu, h\theta)$ as well as velocity u, temperature θ, and pressure $p = \frac{1}{2}g\theta h^2$ are displayed in Figure 2.24. The numerical results of both 200 and 1600 uniform cells are presented in these figures. We can observe that the numerical solutions agree well with the "reference" solutions.

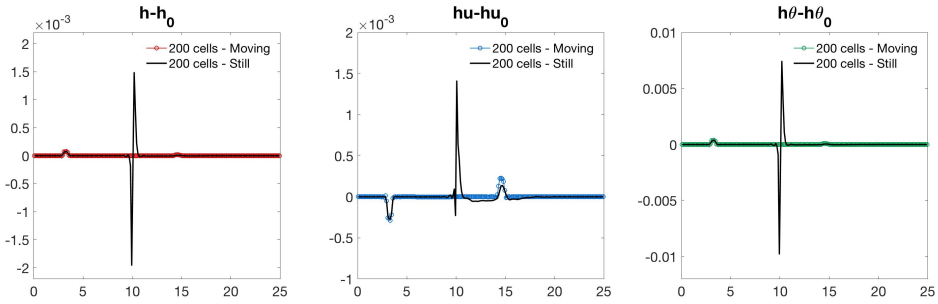


Figure 2.21: The difference between the solutions $h, hu, h\theta$ for the smaller perturbation of the trans-critical flow problem (2.74) at time 0.75 and the corresponding steady state solutions. A uniform mesh of 200 cells has been used for both the still-water and moving-water schemes.

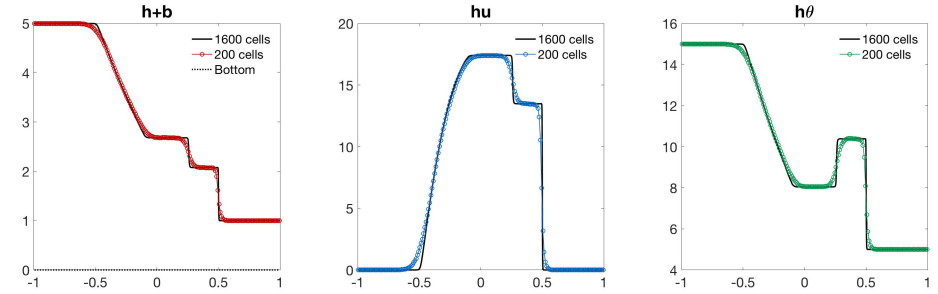


Figure 2.22: Numerical solution for the Riemann problem (2.77) at time $t = 0.04$ with 200 and 1600 cells using the moving-water well-balanced method.

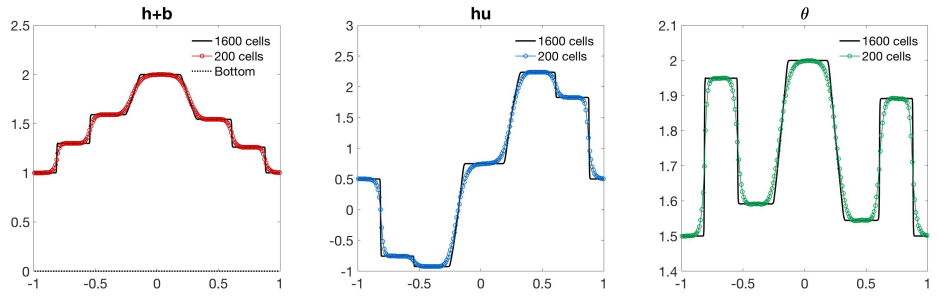


Figure 2.23: Numerical solution for the Riemann problem (2.78) at time $t = 0.075$ with 200 and 1600 cells using the moving-water well-balanced method.

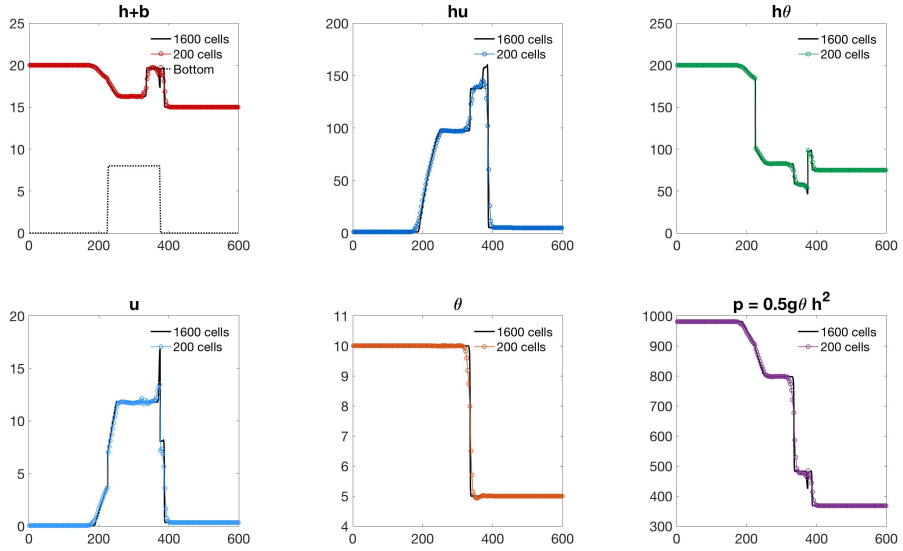


Figure 2.24: Numerical solution for the Riemann problem (2.80) at time $t = 3$ with 200 and 1600 cells using the moving-water well-balanced method. Both the water surface $h + b$, momentum hu, $h\theta$, and the velocity u, θ, pressure p are shown.

Chapter 3

High Order Man-at-Eternal-Rest and Living-Man Equilibria Preserving Discontinuous Galerkin Methods for the Blood Flow Model

3.1 Introduction

The one-dimensional blood flow through arteries can be described by the system of equations

$$\begin{cases} \partial_t A + \partial_x Q = 0, \\ \partial_t Q + \partial_x \left(\dfrac{Q^2}{A} + \dfrac{\beta}{3} A^{\frac{3}{2}} \right) = \dfrac{\beta A}{2\sqrt{A_0}} \partial_x (A_0), \end{cases} \tag{3.1}$$

where $A(x,t) = \pi R^2(x,t)$ is the cross-sectional area of the artery with radius $R(x,t)$ and $Q(x,t) = A(x,t)u(x,t)$ is the discharge with $u(x,t)$ representing the flow velocity. The parameter $\beta = \frac{K}{\rho\sqrt{\pi}}$ is a constant with K denoting the wall stiffness and ρ the blood density. In the source term, $A_0(x) = \pi R_0^2(x)$ is the cross-sectional area at rest. More details of the model are provided in Section 1.1.2 where the model is introduced.

The system (3.1) representing the blood flow through arteries is similar to the SWEs model. This model is widely used for modeling rivers, river networks, lake flows, tides, and tsunamis. The SWEs model and the blood flow model both have two equations representing mass and momentum conservation. They belong to the family of hyperbolic balance laws, and such equations often admit non-trivial steady state solutions. These equilibria involve the perfect cancellation of the source term and the flux gradients in the PDE level, which may not be satisfied numerically due to different numerical approximations to these two terms. As a result, standard numerical methods may not be able to maintain the steady state nor capture the nearly equilibrium flow (small perturbation of the equilibrium state) well, unless a much refined mesh is used in the simulation. To resolve this issue, well-balanced methods [8] are introduced to exactly preserve the steady state solutions at the discrete level. They are often found to be efficient in capturing nearly equilibrium flow on a coarse mesh. There have been extensive studies on designing well-balanced methods for the SWEs over non-flat bottom topography [1, 158, 85] and Euler equations under gravitational fields [156, 28].

The one-dimensional blood model through arteries (3.1) admits non-trivial steady state solutions. By definition, the steady state solutions appear when the conservative

variables (A, Q) do not change over time, or equivalently, $\partial_t A = 0$ and $\partial_t Q = 0$ which leads to

$$\begin{cases} \partial_x Q = 0, \\ \partial_x \Big(\dfrac{Q^2}{A} + \dfrac{\beta}{3} A^{\frac{3}{2}} \Big) = \dfrac{\beta A}{2\sqrt{A_0}} \partial_x (A_0). \end{cases} \tag{3.2}$$

A simple steady state occurs when the velocity becomes 0, known as the *(non-zero pressure) man-at-eternal-rest* steady state or *dead-man* equilibrium in the literature

$$\Big(u, \sqrt{A} - \sqrt{A_0} \Big) = (0, constant). \tag{3.3}$$

A special case of this steady state appears when the pressure in (1.14) is zero. This implies that A reduces to A_0, that is,

$$(u, A) = (0, A_0). \tag{3.4}$$

We will refer to this steady state as the *zero pressure man-at-eternal-rest* steady state. The more general case occurs when the velocity u does not vanish. By some simple algebra, the general equilibrium state, denoted as the *living-man* equilibrium, can be derived as

$$\left(Q, \frac{Q^2}{2A^2} + \beta \left(\sqrt{A} - \sqrt{A_0} \right) \right) = constant. \tag{3.5}$$

One case in which this steady state might occur is in small arteries that are extremely constricted by stenosis. In this case, there is so much flow resistance that the flow loses pulsatility and approaches a steady state with non-zero velocity [60].

Various numerical methods have been designed for the one-dimensional blood flow model. A recent study [11] provides a systematic comparison of six commonly used numerical schemes for one-dimensional blood flow modelling. The numerical results are compared with theoretical results, as well as three-dimensional numerical data in compatible domains,

and good agreement was observed. Recently, well-balanced methods for the blood flow through arteries which are efficient in capturing nearly equilibrium flows have gained more attention. In [41], Delestre and Lagree developed well-balanced first-order and second-order finite volume schemes for the blood flow system in elastic tubes with the man-at-eternal-rest equilibrium. They also showed the appearance of spurious flows when a simple, non-well-balanced, numerical method is used. Müller et al. [106] constructed high order well-balanced Weighted Essentially Non-Oscillatory (WENO) scheme for blood flow in elastic vessels with varying mechanical and geometrical properties. A modified version of the Dumbser-Osher-Toro Riemann solver was introduced to treat the nonconservative term, and they show the resulting methods preserves the man-at-eternal-rest equilibrium exactly. The numerical methods are then extended to networks of elastic vessels with satisfying performance. An upwind discretization for the source term to create a energy-balanced numerical solver was introduced by Murillo et al. [104]. Wang et al. [146] derived high order well-balanced finite difference WENO schemes that possess sharp shock transition. The main idea was to split the source term into two parts and approximate them with compatible WENO operators. In [94], the authors extended the hydrostatic construction idea (commonly used in the design of well-balanced methods for the SWEs) to develop high order DG and finite volume WENO scheme for the blood flow. Most of these well-balanced methods are designed to preserve the stationary man-at-eternal-rest steady state. As explained in [60], such steady states may not be that relevant for blood flow as they only occur in "dead men". For the more general living-man equilibrium state (3.5), well-balanced methods are first studied in [105], where a generalized hydrostatic reconstruction technique was used to

construct the well-balanced numerical flux. In [60], Ghigo et al. presented a simple second order well-balanced method for the one-dimensional blood flow in large arteries, with two well-balanced hydrostatic reconstruction techniques designed to preserve the general steady state solutions. Numerically, the proposed methods outperform the well-balanced methods for the man-at-eternal-rest steady states based on the original hydrostatic reconstruction technique.

This work in this chapter aims to develop high order well-balanced DG methods for both the man-at-eternal-rest (3.3), (3.4) and living-man (3.5) steady states of the one-dimensional blood flow model. High order accurate numerical methods are developed to provide accurate simulation on a relatively coarse mesh. DG methods, which combine the flexibility of the finite element method and stability of the finite volume method, have gained increased attention recently. Another reason we choose DG methods in this study is due to their flexibility to treat the junctions for network problems, as pointed out in [23, 19], which would be important in the simulation of the human cardiovascular system. Specific advantages of the DG scheme include straightforward implementation of junction coupling conditions due to compactness and preservation of high order accuracy. Other high order methods may require stencils (wide or one-sided) that negatively impact the accuracy and stability of the scheme [23]. We start by presenting two simple approaches to design well-balanced methods for the man-at-eternal-rest steady state solution. The first approach is based on the decomposition of the numerical solution and the hydrostatic reconstruction technique, while the second one is based on the well-balanced technique in [152] to split the source term (as done in [146]). We will show the link of these two approaches, although

they are derived based on different motivations. The main component of this chapter is on how to design efficient well-balanced DG methods for the general living-man steady state. Special attention is paid to the projection operator to define the numerical initial condition (piecewise polynomials) of finite element methods. With a carefully chosen projection, one can recover the nonlinear living-man equilibrium states from these numerical initial conditions. Next, the numerical solutions are decomposed into two parts, one corresponding to the equilibrium component, and the other corresponding to the fluctuation. We can show that, if the living-man equilibrium is reached, this decomposition is exact in the sense that the fluctuation part becomes zero. Note that the equilibrium component is computed from the numerical solution at the current time step, and is not given a priori. With this decomposition, the modified solution values at the cell interface can be defined, which can recover the exact equilibrium solutions when the equilibrium state is reached. The general hydrostatic reconstruction idea is then adopted to provide the well-balanced numerical flux. Together with a careful choice of the source term approximation, well-balanced DG methods for the general living-man equilibrium can be designed.

The chapter is organized as follows. In Section 3.2, the necessary notations are introduced and well-balanced DG methods to preserve the man-at-eternal-rest steady state are discussed. Section 3.3 presents the numerical performance of the DG methods in Section 3.2. The well-balanced DG methods that maintain the more general living-man equilibrium state of the blood flow model are discussed in Section 3.4. In Section 3.5, numerical examples are given to demonstrate the high-order accuracy, well-balanced property, and good resolution for smooth and discontinuous solution of the methods described in Section 3.4.

3.2 Man-at-Eternal-Rest Well-Balanced DG Methods

We start by presenting well-balanced DG scheme for the simpler man-at-eternal-rest steady states (3.3), (3.4). The proposed methods will be extended to the general living-man equilibrium (3.5) in Section 3.4.

3.2.1 Notations and Discontinuous Galerkin Methods

Similarly to Chapter 2, we seek to find approximate solutions $\boldsymbol{U}_\tau$ to the state variables $\boldsymbol{U}$ by use of the DG numerical method. We note that the cross-sectional area at rest, A_0, will also be approximated and denoted by $(A_0)_\tau$. We denote $\boldsymbol{U}^+_{\tau,j+\frac{1}{2}} = \begin{pmatrix} A^+_{\tau,j+\frac{1}{2}} \\ Q^+_{\tau,j+\frac{1}{2}} \end{pmatrix}$ as the limit from the right cell I_{j+1} and $\boldsymbol{U}^-_{\tau,j+\frac{1}{2}} = \begin{pmatrix} A^-_{\tau,j+\frac{1}{2}} \\ Q^-_{\tau,j+\frac{1}{2}} \end{pmatrix}$ as the limit from the left cell I_j. Details on the traditional DG scheme are found in Section 1.3.2. In this context the Lax-Friedrich numerical flux term α is derived from one of the eigenvalues of the Jacobian matrix of $\boldsymbol{F}(\boldsymbol{U})$, $\alpha = \max\limits_{U_\tau} \left(\frac{Q_\tau}{A_\tau} + \sqrt{\frac{\beta\sqrt{A_\tau}}{2}} \right)$. The maximum in the calculation of α can be taken either over the entire computational domain or locally.

Our goal is to design well-balanced DG methods which can preserve the man-at-eternal-rest steady states (3.3) and (3.4). Many different approaches to design well-balanced methods have been studied in the literature, mostly for the SWEs with non-flat bottom topography. The key idea of well-balanced methods in this chapter is to decompose the numerical solution at each time step into the equilibrium part and the fluctuation part, which has also been studied in [150]. We carefully choose the decomposition so that, if the steady state is reached, the equilibrium part recovers the steady state perfectly. Then

by approximating the contribution of the equilibrium part and the fluctuation part in the source term in a different way, one can achieve well-balanced property. The details are given below. We would like to comment that this approach is somehow similar to that of solving the new PDE with the unknown being the perturbation to the equilibrium state. The main difference is that our approach does not assume the explicit knowledge of the equilibrium state a priori, and we recover that equilibrium part (more specifically, the constant values in (3.5)) numerically.

In the framework of DG methods, all of the numerical solutions (including U_τ and $(A_0)_\tau$) are discontinuous at the cell interfaces, even at the steady state. To address this, we follow the idea of hydrostatic reconstruction, and present our well-balanced numerical scheme in the form that was introduced in Section 2.2.1 as

$$\int_{I_j} \partial_t \boldsymbol{U}_\tau^n v \, dx - \int_{I_j} \boldsymbol{F}(\boldsymbol{U}_\tau^n) \partial_x v \, dx + \hat{\boldsymbol{F}}^l_{j+\frac{1}{2}} v^-_{j+\frac{1}{2}} - \hat{\boldsymbol{F}}^r_{j-\frac{1}{2}} v^+_{j-\frac{1}{2}} = \int_{I_j} \boldsymbol{S}(\boldsymbol{U}_\tau^n, b_\tau) v \, dx. \quad (3.6)$$

A similar form has been described to obtain well-balanced methods for the SWEs [160] and for the blood flow model [94].

The focus of the following subsections will be on defining the left and right fluxes as well as presenting how the source term is evaluated. To illustrate the approaches, we will start with the simpler zero pressure man-at-eternal-rest case (3.4) in Section 3.2.2. The non-zero pressure man-at-eternal-rest case (3.3) will be discussed in Section 3.2.3.

3.2.2 The Zero Pressure Man-at-Eternal-Rest Well-Balanced Scheme

Well-Balanced Numerical Fluxes

First, at each time step, we decompose the conservative unknown variables $\boldsymbol{U}_\tau$ into the sum of a reference equilibrium state $\boldsymbol{U}_\tau^e$ and a fluctuation part $\boldsymbol{U}_\tau^f$. Taking the zero pressure man-at-eternal-rest steady state (3.4) in consideration, we can define the equilibrium part of the conservative variables in I_j by

$$\boldsymbol{U}_{\tau,j}^e(x) = \begin{pmatrix} A_{\tau,j}^e(x) \\ \\ Q_{\tau,j}^e(x) \end{pmatrix} = \begin{pmatrix} (A_0)_{\tau,j}(x) \\ \\ 0 \end{pmatrix}, \tag{3.7}$$

since the equilibrium state is explicitly given. The fluctuation part $\boldsymbol{U}_\tau^f$ can be determined by the decomposition of the summation

$$\boldsymbol{U}_\tau = \boldsymbol{U}_\tau^e + \boldsymbol{U}_\tau^f, \tag{3.8}$$

which leads to

$$\boldsymbol{U}_{\tau,j}^f(x) = \begin{pmatrix} A_{\tau,j}^f(x) \\ \\ Q_{\tau,j}^f(x) \end{pmatrix} = \begin{pmatrix} A_{\tau,j}(x) - (A_0)_{\tau,j}(x) \\ \\ Q_{\tau,j}(x) \end{pmatrix}. \tag{3.9}$$

When the solution is at a steady state, one can observe that the equilibrium parts $\boldsymbol{U}_\tau^e$ are equivalent to $\boldsymbol{U}_\tau$, hence $\boldsymbol{U}_\tau^f = 0$. The notations of $\boldsymbol{U}_\tau^e$ and $\boldsymbol{U}_\tau^f$ are introduced here to be consistent with those in the living-man well-balanced methods in Section 3.4, and are not necessary for this simpler man-at-eternal-rest steady state problem.

The idea of hydrostatic reconstruction is used for computing the numerical fluxes. It was first introduced by Audusse in [1]. At time step t^n, the cell interface values $\boldsymbol{U}_{\tau,j+\frac{1}{2}}^{\pm}$ are computed first. We construct the cell interface value of A_0 as

$$(A_0)_{\tau,j+\frac{1}{2}}^* = \max\left((A_0)_{\tau,j+\frac{1}{2}}^+, (A_0)_{\tau,j+\frac{1}{2}}^-\right), \tag{3.10}$$

and use it to evaluate the modified cell interface values of A_τ

$$
\begin{aligned}
A^{*,-}_{\tau,j+\frac{1}{2}} &= \max\left((A_0)^*_{\tau,j+\frac{1}{2}} + A^{f,-}_{\tau,j+\frac{1}{2}}, 0\right) = \max\left(A^-_{\tau,j+\frac{1}{2}} - (A_0)^-_{\tau,j+\frac{1}{2}} + (A_0)^*_{\tau,j+\frac{1}{2}}, 0\right), \\
A^{*,+}_{\tau,j+\frac{1}{2}} &= \max\left((A_0)^*_{\tau,j+\frac{1}{2}} + A^{f,+}_{\tau,j+\frac{1}{2}}, 0\right) = \max\left(A^+_{\tau,j+\frac{1}{2}} - (A_0)^+_{\tau,j+\frac{1}{2}} + (A_0)^*_{\tau,j+\frac{1}{2}}, 0\right).
\end{aligned}
\tag{3.11}
$$

The new cell boundary values for $\boldsymbol{U}$ are then defined as

$$
\boldsymbol{U}^{*,\pm}_{\tau,j+\frac{1}{2}} = \begin{pmatrix} A^{*,\pm}_{\tau,j+\frac{1}{2}} \\ Q^{\pm}_{\tau,j+\frac{1}{2}} \end{pmatrix}. \tag{3.12}
$$

Lastly, the left and right fluxes are determined in the following manner

$$
\begin{aligned}
\hat{\boldsymbol{F}}^l_{j+\frac{1}{2}} &= \boldsymbol{f}\left(\boldsymbol{U}^{*,-}_{\tau,j+\frac{1}{2}}, \boldsymbol{U}^{*,+}_{\tau,j+\frac{1}{2}}\right) + \begin{pmatrix} 0 \\ \frac{\beta}{3}(A^-_{\tau,j+\frac{1}{2}})^{\frac{3}{2}} - \frac{\beta}{3}(A^{*,-}_{\tau,j+\frac{1}{2}})^{\frac{3}{2}} \end{pmatrix}, \\
\hat{\boldsymbol{F}}^r_{j-\frac{1}{2}} &= \boldsymbol{f}\left(\boldsymbol{U}^{*,-}_{\tau,j-\frac{1}{2}}, \boldsymbol{U}^{*,+}_{\tau,j-\frac{1}{2}}\right) + \begin{pmatrix} 0 \\ \frac{\beta}{3}(A^+_{\tau,j-\frac{1}{2}})^{\frac{3}{2}} - \frac{\beta}{3}(A^{*,+}_{\tau,j-\frac{1}{2}})^{\frac{3}{2}} \end{pmatrix}.
\end{aligned}
\tag{3.13}
$$

The choice of $\boldsymbol{U}^{*,\pm}_{\tau,j+\frac{1}{2}}$ was defined in the way such that they are the same at the cell interfaces when steady state is reached, which is desirable for achieving the well-balanced property. Notice that at the steady state, the left and right fluxes simplify to $\hat{\boldsymbol{F}}^l_{j+\frac{1}{2}} = \boldsymbol{F}\left(\boldsymbol{U}^-_{\tau,j+\frac{1}{2}}\right)$ and $\hat{\boldsymbol{F}}^r_{j-\frac{1}{2}} = \boldsymbol{F}\left(\boldsymbol{U}^+_{\tau,j-\frac{1}{2}}\right)$ as a result of the numerical flux $\boldsymbol{f}$ being consistent.

The Source Term Approximation

In an effort to balance the source term with the numerical fluxes, an approximation of the source term will be discussed in this section. The source term $\boldsymbol{S}(\boldsymbol{U}, A_0) = \frac{\beta A}{2\sqrt{A_0}}(A_0)_x = \beta A(\sqrt{A_0})_x$ is linear with respect to the variable A. As a result of this linearity

and the decomposition of $\boldsymbol{U}_\tau$ in (3.8), the source term can be decomposed as

$$\int \boldsymbol{S}\left(\boldsymbol{U}_\tau, (A_0)_\tau\right) v \, dx = \int \boldsymbol{S}\left(\boldsymbol{U}^e_\tau, (A_0)_\tau\right) v \, dx + \int \boldsymbol{S}\left(\boldsymbol{U}^f_\tau, (A_0)_\tau\right) v \, dx. \tag{3.14}$$

The second term on the right hand side can be directly computed by a quadrature rule. On the other hand, since the function $\boldsymbol{U}^e_\tau(x)$ is the equilibrium state, we recall the following relation

$$\int_{I_j} \boldsymbol{S}\left(\boldsymbol{U}^e_\tau, (A_0)_\tau\right) v \, dx = -\int_{I_j} \boldsymbol{F}\left(\boldsymbol{U}^e_\tau\right) v_x \, dx + \boldsymbol{F}\left(\boldsymbol{U}^{e,-}_{\tau,j+\frac{1}{2}}\right) v^-_{j+\frac{1}{2}} - \boldsymbol{F}\left(\boldsymbol{U}^{e,+}_{\tau,j-\frac{1}{2}}\right) v^+_{j-\frac{1}{2}}, \tag{3.15}$$

holds. When quadrature rule is used for numerical integration, this equality holds approximately, up to the accuracy of the quadrature rule. For the purpose of well-balancedness, we cannot use (3.14) with quadrature rule to approximate the source term, instead, the approximation for the source term will be evaluated by

$$\begin{aligned}
&\int_{I_j} \boldsymbol{S}\left(\boldsymbol{U}_\tau, (A_0)_\tau\right) v \, dx \\
&= -\int_{I_j} \boldsymbol{F}\left(\boldsymbol{U}^e_\tau\right) v_x \, dx + \boldsymbol{F}\left(\boldsymbol{U}^{e,-}_{\tau,j+\frac{1}{2}}\right) v^-_{j+\frac{1}{2}} - \boldsymbol{F}\left(\boldsymbol{U}^{e,+}_{\tau,j-\frac{1}{2}}\right) v^+_{j-\frac{1}{2}} + \int_{I_j} \boldsymbol{S}\left(\boldsymbol{U}^f_\tau, (A_0)_\tau\right) v \, dx.
\end{aligned} \tag{3.16}$$

Taking the choice of $\boldsymbol{U}^e_\tau$ in (3.7), the decomposition of the source term (3.14) becomes

$$\int \beta A_\tau (\sqrt{(A_0)_\tau})_x v \, dx = \int \beta A_0 (\sqrt{(A_0)_\tau})_x v \, dx + \int \beta (A_\tau - (A_0)_\tau)(\sqrt{(A_0)_\tau})_x v \, dx, \tag{3.17}$$

and the approximation for the source term in (3.16) reduces to

$$\begin{aligned}
\int_{I_j} \beta A_\tau (\sqrt{A_0})_x v \, dx = &-\int_{I_j} \frac{\beta}{3} \left((A_0)_\tau\right)^{\frac{3}{2}} v_x \, dx + \frac{\beta}{3} \left((A_0)^-_{\tau,j+\frac{1}{2}}\right)^{\frac{3}{2}} v^-_{j+\frac{1}{2}} \\
&- \frac{\beta}{3} \left((A_0)^+_{\tau,j-\frac{1}{2}}\right)^{\frac{3}{2}} v^+_{j-\frac{1}{2}} + \int_{I_j} (A_\tau - (A_0)_\tau)(\sqrt{(A_0)_\tau})_x v \, dx.
\end{aligned} \tag{3.18}$$

Remark 11. *The source term used in this chapter is due to only the cross-sectional area at rest. The decomposition* (3.14) *will not hold if the source term also included a friction term or another term that is not linear with respect to the conservative variables. However, one could consider a similar decomposition for the source term as follows*

$$\int \boldsymbol{S}\left(\boldsymbol{U}_\tau,(A_0)_\tau\right) v \; dx = \int \boldsymbol{S}\left(\boldsymbol{U}^e_\tau,(A_0)_\tau\right) v \; dx + \int \left(\boldsymbol{S}\left(\boldsymbol{U}_\tau,(A_0)_\tau\right) - \boldsymbol{S}\left(\boldsymbol{U}^e_\tau,(A_0)_\tau\right)\right) v \; dx. \tag{3.19}$$

The first term on the right hand side, $\int \boldsymbol{S}\left(\boldsymbol{U}^e_\tau,(A_0)_\tau\right) v \; dx$*, can be approximated by* (3.15)*. The second term on the right hand side can be computed using numerical integration with an appropriate quadrature scheme. It is interesting to note that* $\boldsymbol{S}\left(\boldsymbol{U}_\tau,(A_0)_\tau\right) - \boldsymbol{S}\left(\boldsymbol{U}^e_\tau,(A_0)_\tau\right) = \boldsymbol{S}\left(\boldsymbol{U}_\tau - \boldsymbol{U}^e_\tau,(A_0)_\tau\right) = \boldsymbol{S}\left(\boldsymbol{U}^f_\tau,(A_0)_\tau\right)$ *when the source is linear, this leads to the original source decomposition formulation.*

Remark 12. *In the well-balanced methods designed for the SWEs in [153], a straightforward numerical integration of the source term by a quadrature rule accurate for polynomial of degree* $3k-1$ *is sufficient. This is due to the fact that the equation* (3.15) *holds exactly with sufficiently accurate quadrature, therefore, equations* (3.14) *and* (3.16) *are equivalent. For the blood flow with a source term of the form* $\boldsymbol{S}(\boldsymbol{U}_\tau,(A_0)_\tau) = \frac{\beta A_\tau}{2\sqrt{(A_0)_\tau}}\left((A_0)_\tau\right)_x$*, a direct numerical integral with quadrature rules does not yield a well-balanced method, because* $\sqrt{(A_0)_\tau}$ *is no longer a polynomial, hence any numerical integration may not be exact. For the steady state problem, the numerical error would be dominated by the integration error, which becomes non-negligible, especially on a coarse mesh.*

We conclude this subsection by showing the scheme indeed satisfies the well-balanced property.

Proposition 13. *The DG scheme* (2.10) *for the blood flow system* (1.15) *with the zero pressure man-at-eternal-rest steady state* (3.4) *is well-balanced when paired with the numerical fluxes* (3.13) *and the source term decomposition* (3.16).

Proof. At the steady state, we have $\boldsymbol{U}_\tau^f = 0$ and $\boldsymbol{U}_\tau^e = \boldsymbol{U}_\tau$. The approximation (3.16) to the source term becomes

$$\begin{aligned}\int_{I_j} \boldsymbol{S}\left(\boldsymbol{U}_\tau, (A_0)_\tau\right) v \, dx &= -\int_{I_j} \boldsymbol{F}\left(\boldsymbol{U}_\tau^e\right) v_x \, dx + \boldsymbol{F}\left(\boldsymbol{U}^{e,-}_{\tau,j+\frac{1}{2}}\right) v^-_{j+\frac{1}{2}} - \boldsymbol{F}\left(\boldsymbol{U}^{e,+}_{\tau,j-\frac{1}{2}}\right) v^+_{j-\frac{1}{2}} \\ &= -\int_{I_j} \boldsymbol{F}\left(\boldsymbol{U}_\tau\right) v_x \, dx + \boldsymbol{F}\left(\boldsymbol{U}^{-}_{\tau,j+\frac{1}{2}}\right) v^-_{j+\frac{1}{2}} - \boldsymbol{F}\left(\boldsymbol{U}^{+}_{\tau,j-\frac{1}{2}}\right) v^+_{j-\frac{1}{2}}.\end{aligned} \tag{3.20}$$

Additionally, the left and right fluxes (3.13) simplify to $\hat{\boldsymbol{F}}^l_{j+\frac{1}{2}} = \boldsymbol{F}\left(\boldsymbol{U}^-_{\tau,j+\frac{1}{2}}\right)$ and $\hat{\boldsymbol{F}}^r_{j-\frac{1}{2}} = \boldsymbol{F}\left(\boldsymbol{U}^+_{\tau,j-\frac{1}{2}}\right)$ at the steady state. Therefore, we have shown the fluxes and source term balance, which implies the scheme is indeed well-balanced. ■

An Alternative Zero Pressure Man-at-Eternal-Rest Well-Balanced DG Scheme

In this subsection, we present an alternative well-balanced DG method for the zero pressure man-at-eternal-rest steady state (3.4). This follows the idea of decomposing the source term, proposed first in [152] for the SWEs, and later in [95, 154] for a general class of hyperbolic balance laws. The same idea has been studied in [146] to develop well-balanced finite difference WENO scheme for the blood flow model.

The key idea is to introduce the following source term decomposition,

$$\frac{\beta A}{2\sqrt{A_0}}\partial_x(A_0) = \beta(A - A_0)\partial_x\sqrt{A_0} + \partial_x\left(\frac{\beta}{3}A_0^{\frac{3}{2}}\right), \tag{3.21}$$

and we refer to [154] for the motivation of such decomposition. Note that this coincides with the source term approximation (3.18), although they arise from different approaches.

We can then move the term $\partial_x\left(\frac{\beta}{3}A_0^{\frac{3}{2}}\right)$ to the left side and combine it with the flux to achieve the updated equation of the form

$$\begin{cases} \partial_t A + \partial_x Q = 0, \\ \partial_t Q + \partial_x\Big(\dfrac{Q^2}{A} + \dfrac{\beta}{3}A^{\frac{3}{2}} - \dfrac{\beta}{3}A_0^{\frac{3}{2}}\Big) = \beta(A - A_0)\partial_x\left(\sqrt{A_0}\right), \end{cases} \tag{3.22}$$

It is clear that at the zero pressure man-at-eternal-rest steady state $(Q, A) = (0, A_0)$, the system of PDEs (3.22) has both zero flux and source term, hence the traditional DG scheme (1.17) is automatically well-balanced when the Lax-Friedrichs fluxes defined in (1.18) is updated to be

$$\boldsymbol{f}(\boldsymbol{U}_\tau^-, \boldsymbol{U}_\tau^+) = \frac{1}{2}\left(\boldsymbol{F}(\boldsymbol{U}_\tau^-) + \boldsymbol{F}(\boldsymbol{U}_\tau^+) - \alpha\left(\begin{pmatrix} A_\tau^+ - (A_0)_\tau^+ \\ Q_\tau^+ \end{pmatrix} - \begin{pmatrix} A_\tau^- - (A_0)_\tau^- \\ Q_\tau^- \end{pmatrix}\right)\right), \tag{3.23}$$

so that the added numerical diffusion term disappears at the steady state. Note that $\boldsymbol{F}$ in (3.23) now corresponds to the flux in the updated form (3.22). The proof of the well-balanced property is rather straightforward and is not included here.

This is a simple approach to achieve well-balanced property, and there is no need to introduce the hydrostatic reconstruction idea when constructing the numerical fluxes. However, this cannot be extended to the more complicated living-man equilibrium case.

3.2.3 The Non-Zero Pressure Man-at-Eternal-Rest Well-Balanced Scheme

In this section, we present well-balanced DG methods for the man-at-eternal-rest steady state with non-zero pressure (3.3).

We start with the description of well-balanced numerical fluxes. For the non-zero pressure man-at-eternal-rest steady state (3.3), the choice of the decomposition into $\boldsymbol{U}_\tau^e$

and $\boldsymbol{U}_\tau^f$, as well as the definition of $A^{*,\pm}_{\tau,j+\frac{1}{2}}$ must be modified from those presented in the previous subsection. The decomposed variables will be computed based on the steady state solution (3.3), therefore we denote $\boldsymbol{V} = \begin{pmatrix} Q \\ E \end{pmatrix} = \begin{pmatrix} Q \\ \sqrt{A} - \sqrt{A_0} \end{pmatrix}$ to be the equilibrium variables. The reference equilibrium values $\hat{\boldsymbol{V}}$ in each cell I_j are defined by the following,

$$\hat{\boldsymbol{V}}_j = \begin{pmatrix} \hat{Q}_j \\ \hat{E}_j \end{pmatrix} = \begin{pmatrix} Q_\tau\left(x^-_{j+\frac{1}{2}}\right) \\ \left(\sqrt{A_{\tau,j}} - \sqrt{(A_0)_{\tau,j}}\right)\left(x^-_{j+\frac{1}{2}}\right) \end{pmatrix}, \tag{3.24}$$

which will be constant for all j if the system is at a steady state. It follows that the equilibrium part $\boldsymbol{U}^e_{\tau,j}$ can be defined using $\hat{\boldsymbol{V}}_j$ and the true value of $A_0(x)$,

$$\boldsymbol{U}^e_{\tau,j}(x) = \begin{pmatrix} A^e_{\tau,j}(x) \\ Q^e_{\tau,j}(x) \end{pmatrix} = \begin{pmatrix} \mathbb{P}\left(\left(\hat{E}_j + \sqrt{A_0(x)}\right)^2\right) \\ \hat{Q}_j \end{pmatrix}, \tag{3.25}$$

where the operator $\mathbb{P}$ can be taken as any projection into the piecewise polynomial space $\mathbb{V}^k_\tau$ as long as it is the same projection that was used to evaluate the numerical initial condition. The fluctuation part, $\boldsymbol{U}^f_{\tau,j}$, is again defined as in (3.8). With this, the cell interface value of A_0 takes the form of

$$(A_0)^*_{\tau,j+\frac{1}{2}} = \max\left((A_0)^+_{\tau,j+\frac{1}{2}}, (A_0)^-_{\tau,j+\frac{1}{2}}\right), \tag{3.26}$$

and the modified cell interface values of A_τ become

$$\begin{aligned} A^{*,-}_{\tau,j+\frac{1}{2}} &= \max\left(\left(\hat{E}_j + \sqrt{(A_0)^*_{\tau,j+\frac{1}{2}}}\right)^2 + A^{f,-}_{\tau,j+\frac{1}{2}}, 0\right), \\ A^{*,+}_{\tau,j+\frac{1}{2}} &= \max\left(\left(\hat{E}_{j+1} + \sqrt{(A_0)^*_{\tau,j+\frac{1}{2}}}\right)^2 + A^{f,+}_{\tau,j+\frac{1}{2}}, 0\right). \end{aligned} \tag{3.27}$$

The definition of $\boldsymbol{U}^{*,\pm}_{\tau,j+\frac{1}{2}}$, as well as the left and right fluxes $\hat{\boldsymbol{F}}^l_{j+\frac{1}{2}}$, $\hat{\boldsymbol{F}}^r_{j-\frac{1}{2}}$, are defined in the same way, as in (3.13). These well-balanced numerical fluxes are consistent with those for the living-man equilibrium which will be presented in Section 3.4.

Alternatively, one may also follow the approach in [94] by letting

$$\sqrt{(A_0)^*_{\tau,j+\frac{1}{2}}} = \max\left(\sqrt{(A_0)^+_{\tau,j+\frac{1}{2}}}, \sqrt{(A_0)^-_{\tau,j+\frac{1}{2}}}\right), \tag{3.28}$$

and then redefining A_τ at the cell interfaces as

$$\begin{aligned}
\sqrt{A^{*,-}_{\tau,j+\frac{1}{2}}} &= \max\left(\sqrt{A^-_{\tau,j+\frac{1}{2}}} - \sqrt{(A_0)^-_{\tau,j+\frac{1}{2}}} + \sqrt{(A_0)^*_{\tau,j+\frac{1}{2}}}, 0\right), \\
\sqrt{A^{*,+}_{\tau,j+\frac{1}{2}}} &= \max\left(\sqrt{A^+_{\tau,j+\frac{1}{2}}} - \sqrt{(A_0)^+_{\tau,j+\frac{1}{2}}} + \sqrt{(A_0)^*_{\tau,j+\frac{1}{2}}}, 0\right),
\end{aligned} \tag{3.29}$$

without involving $\boldsymbol{U}^e_\tau$ and $\boldsymbol{U}^f_\tau$ in the definition of the numerical fluxes.

As for the source term approximation, we note that the source term decomposition (3.16) still holds. While $\boldsymbol{U}^e$ and $\boldsymbol{U}^f$ are defined differently in this subsection, the decomposition approach presented in Subsection 3.2.2 can still be applied. Note that the direct numerical integration may not give well-balanced methods, as explained in Remark 12. This completes our description of well-balanced methods for the non-zero pressure man-at-eternal-rest steady state (3.3). One can show that the living-man well-balanced method in Section 3.4 can reduce to this man-at-eternal-rest well-balanced method. Furthermore, one can show (3.29) and (3.27) each simplify to (3.11) when $A = A_0$.

Proposition 14. *The DG scheme* (2.10) *for the blood flow system* (1.15) *with the non-zero pressure man-at-eternal-rest steady state* (3.3) *is well-balanced when paired with* (3.29) *or* (3.27)*, the numerical fluxes* (3.13)*, and the source term decomposition* (3.16)*.*

The proof is similar to the the zero-pressure case and is thus omitted here.

Remark 15. *When the cross-sectional area at rest, A_0, is constant, the traditional DG scheme is recovered, i.e., the numerical fluxes reduce to standard flux and the source term*

approximation is simply zero. First, it is easy to observe that the source term approximation (3.18) *reduces to exactly 0, since* $(A_0)_\tau$ *is constant. Second, we will show that the left and right numerical fluxes reduce to the original DG fluxes. When* A_0 *is constant, then* $(A_0)^*_{\tau,j+\frac{1}{2}} = (A_0)^+_{\tau,j+\frac{1}{2}} = (A_0)^-_{\tau,j+\frac{1}{2}}$*, so the calculation of* $A^{*,\pm}_\tau$ *by definitions* (3.11), (3.29) *or* (3.27) *reduces to*

$$A^{*,\pm}_{\tau,j+\frac{1}{2}} = \max\left(A^{\pm}_{\tau,j+\frac{1}{2}}, 0\right) = A^{\pm}_{\tau,j+\frac{1}{2}}. \tag{3.30}$$

More generally, we have that $\boldsymbol{U}^{*,\pm}_{\tau,j+\frac{1}{2}} = \boldsymbol{U}^{\pm}_{\tau,j+\frac{1}{2}}$*. Therefore, by definition of the left and right fluxes, we obtain* $\hat{\boldsymbol{F}}^l_{j+\frac{1}{2}} = \hat{\boldsymbol{F}}_{j+\frac{1}{2}}$ *and* $\hat{\boldsymbol{F}}^r_{j-\frac{1}{2}} = \hat{\boldsymbol{F}}_{j-\frac{1}{2}}$.

3.3 Numerical Tests for the Man-at-Eternal-Rest Well-Balanced Methods

In this section, we present some numerical examples by testing the well-balanced DG scheme designed for the man-at-eternal-rest steady state in Section 3.2. The third order TVD Runge-Kutta time discretization (1.19) is used in conjunction with piecewise quadratic polynomials ($k = 2$) in space, unless otherwise stated. The CFL number is taken to be 0.15.

3.3.1 Test for Accuracy

Our first numerical example tests the accuracy of our man-at-eternal-rest well-balanced scheme on a problem with smooth solutions. The initial conditions for $x \in [0, 10]$ are

$$A(x,0) = \sin\left(\frac{\pi}{5}x\right) + 10, \qquad Q(x,0) = e^{\cos(\frac{\pi}{5}x)}, \tag{3.31}$$

with the cross-sectional area at rest

$$A_0(x) = \frac{1}{2}\cos^2\left(\frac{\pi}{5}x\right) + 5,$$

and $K = 10^8\ {}^{Pa}/_{m}$, $\rho = 1060\ {}^{kg}/_{m^3}$. Periodic boundary conditions are employed. We compute until time $t = 0.01$ when the solution is still smooth. Since there is no explicitly known solution in this case, the errors are computed by iteratively comparing results from meshes of uniform cell widths h and $h/2$. Table 3.1 contains the L^1 errors and orders of accuracy for P^0, P^1 and P^2 polynomials. For each polynomial degree k, we see that $(k+1)^{\text{th}}$ order accuracy is achieved.

		$k=0$		$k=1$		$k=2$	
Variable	J	L^1 Error	Order	L^1 Error	Order	L^1 Error	Order
	25	6.1718e-01		4.0986e-02		1.7291e-03	
	50	4.0692e-01	0.6009	1.0077e-02	2.0241	2.2503e-04	2.9418
A	100	2.4564e-01	0.7282	2.5017e-03	2.0093	2.8740e-05	2.9690
	200	1.3642e-01	0.8485	6.2420e-04	2.0028	3.6297e-06	2.9851
	400	7.2062e-02	0.9207	1.5614e-04	1.9992	4.5768e-07	2.9874
	25	2.5275e02		6.2128e00		3.4189e-01	
	50	1.4833e02	0.7689	1.5283e00	2.0233	4.1757e-02	3.0334
Q	100	8.0719e01	0.8779	3.8083e-01	2.0047	5.1194e-03	3.0280
	200	4.2203e01	0.9356	9.5287e-02	1.9989	6.3175e-04	3.0185
	400	2.1582e01	0.9676	2.3935e-02	1.9931	7.8696e-05	3.0050

Table 3.1: L^1 errors and convergence orders of the accuracy test in Section 3.3.1, using P^0, P^1 and P^2 piecewise polynomials and the man-at-eternal-rest well-balanced method. In each case, $k+1$ order of accuracy is achieved.

3.3.2 Tests for the Well-Balanced Property

A Zero Pressure Man-at-Eternal-Rest Steady State

In this section, we demonstrate that the proposed DG scheme aptly preserves the zero pressure man-at-eternal-rest steady state (3.4) with a non-constant cross-sectional area at rest. We consider the case of a dead man with an aneurysm, which implies there is no blood flow occurring in a section of an artery that has a non-constant radius. An aneurysm occurs when the arterial wall weakens and balloons outwards. The initial conditions for the radius, R, are given by

$$R(x,0) = R_0(x) = \begin{cases} \tilde{R}, & \text{if } x \in [0, x_1] \cup [x_4, L], \\ \tilde{R} + \frac{\Delta R}{2}\left[\sin\left(\frac{x-x_1}{x_2-x_1}\pi - \frac{\pi}{2}\right) + 1\right], & \text{if } x \in [x_1, x_2], \\ \tilde{R} + \Delta R, & \text{if } x \in [x_2, x_3], \\ \tilde{R} + \frac{\Delta R}{2}\left[\cos\left(\frac{x-x_3}{x_4-x_3}\pi\right) + 1\right], & \text{if } x \in [x_3, x_4], \end{cases} \tag{3.32}$$

for an artery of length L, with all the remaining parameters found in Table 3.2. The initial conditions for the cross-sectional area and cross-sectional area at rest are thus given by

$$A(x,0) = \pi R(x,0)^2, \qquad A_0(x) = \pi R_0(x)^2. \tag{3.33}$$

The initial velocity is assumed to be zero, thus $Q(x,0) = 0$. We impose transmissive boundary conditions at both endpoints of the domain and compute this example until time $t = 5$. Since the initial condition is the man-at-eternal-rest steady state, the solution should stay unchanged. The L^1 and L^∞ errors of the numerical solutions are shown in Table 3.3 and demonstrate that the well-balanced property was maintained when using the

$\tilde{R}$	ΔR	K	ρ	
$4 \times 10^{-3} m$	$10^{-3} m$	10^8 Pa/m	1060 kg/m^3	
x_1	x_2	x_3	x_4	L
$10^{-2} m$	$3.05 \times 10^{-2} m$	$4.95 \times 10^{-2} m$	$7 \times 10^{-2} m$	$0.14m$

Table 3.2: Parameters in the initial condition (3.32) for the zero pressure man-at-eternal-rest well-balanced tests.

man-at-eternal-rest well-balanced scheme. The errors were computed by comparing the numerical solution to the numerical initial conditions. Figure 3.1 shows the area of the artery and the velocity at $t = 5$ with a mesh of 200 cells.

		Traditional DG Scheme		MAER WB DG Scheme	
Variable	Error Type	L^1 Error	L^∞ Error	L^1 Error	L^∞ Error
A	Absolute	1.8404e-13	1.7712e-08	2.042e-19	2.4335e-15
	Relative	3.3148e-09	3.1990e-04	3.5744e-15	3.8730e-11
Q	Absolute	1.8404e-13	1.7712e-08	2.042e-19	2.4335e-15

Table 3.3: Table of absolute and relative L^1 and L^∞ errors for the zero pressure man-at-eternal-rest (MAER) well-balanced test representing an aneurysm in Section 3.3.2. Errors are given for both the traditional DG scheme and the well-balanced scheme. The well-balanced scheme demonstrates the well-balanced property, while the traditional DG scheme does not have the well-balanced property. Relative errors are included for A since the scale of the problem is so small. However, relative errors are not included for Q since the exact value is identically 0.

For comparison, we also compute the same test using the traditional DG method in which the standard numerical fluxes are used and the source term is computed with a straightforward numerical integration. Figure 3.2 and the errors in Table 3.3 demonstrate that the traditional DG scheme does not preserve the steady state exactly in the discrete

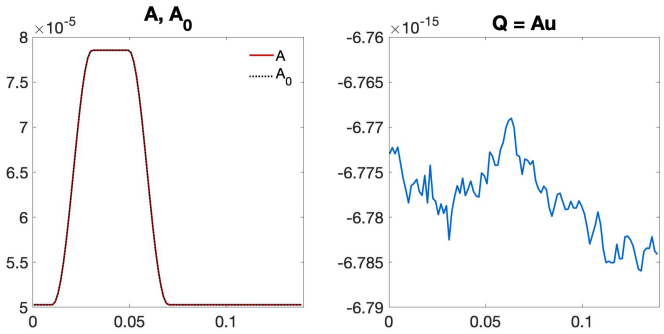


Figure 3.1: Numerical solutions of the man-at-eternal-rest problem in Section 3.3.2, at time $t = 5$ with quadratic basis functions and mesh of size 200 uniform cells.

level.

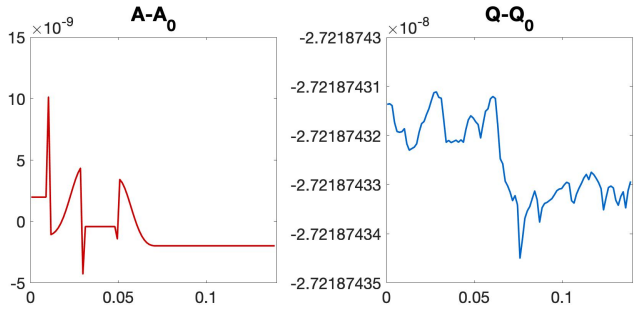


Figure 3.2: The difference between A, Q at the final time $t = 5$ and the numerical initial conditions, when the traditional DG scheme is used for the man-at-eternal-rest problem in Section 3.3.2.

A Non-Zero Pressure Man-at-Eternal-Rest Steady State

In this subsection, we consider the case of a dead man with stenosis. Stenosis occurs when the artery narrows and it leads to reduced blood flow from the heart to the rest of the body. Stenosis can be caused by a congenital heart defect, calcium buildup,or

rheumatic fever which is a result of a strep throat infection. The radius at rest, R_0, for an artery of length L is given by

$$R_0(x) = \begin{cases} \tilde{R} + \Delta R, & \text{if } x \in [0, x_1] \cup [x_4, L], \\ \tilde{R} - \frac{\Delta R}{2}\left[\sin\left(\frac{x-x_1}{x_2-x_1}\pi - \frac{\pi}{2}\right) - 1\right], & \text{if } x \in [x_1, x_2], \\ \tilde{R}, & \text{if } x \in [x_2, x_3], \\ \tilde{R} - \frac{\Delta R}{2}\left[\cos\left(\frac{x-x_3}{x_4-x_3}\pi\right) - 1\right], & \text{if } x \in [x_3, x_4], \end{cases} \tag{3.34}$$

where all the parameters are found in Table 3.4.

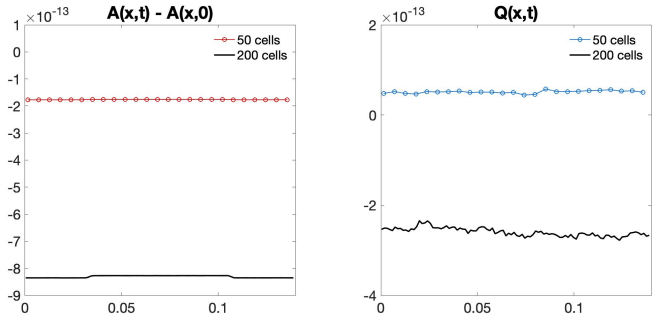
$\tilde{R}$	ΔR	K	ρ	L	x_1	x_2	x_3	x_4
$4 \times 10^{-3}m$	$10^{-3}m$	10^8 Pa/m	1060 kg/m^3	$0.14m$	$\frac{9L}{40}$	$\frac{1L}{4}$	$\frac{3L}{4}$	$\frac{31L}{40}$

Table 3.4: Parameters in (3.34) for the non-zero pressure man-at-eternal-rest well-balanced tests.

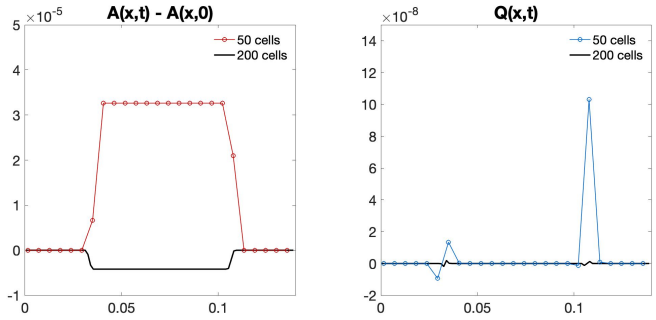
Then the initial conditions are determined by the equilibrium values, that is

$$A(x,0) = \left(C + \sqrt{\pi}R_0(x)\right)^2, \qquad Q(x,0) = 0. \tag{3.35}$$

where we set the constant $C = 10^{-3}$. We impose transmissive boundary conditions and run the scheme until the final time of $t = 1$ on mesh sizes of 50 and 200 uniform cells. The errors are found in Table 3.5 and demonstrate that the well-balanced property is preserved even on the coarse mesh of 50 cells. We also compare the results of the traditional DG scheme and non-zero pressure man-at-eternal-rest scheme well-balanced DG scheme in Figure 3.3. It is clear from the figures that the use of a well-balanced scheme is especially important for preserving the non-constant area A.



(a)Non-zero pressure man-at-eternal-rest well-balanced DG scheme



(b)Traditional DG scheme

Figure 3.3: Plots of the errors at time $t = 1$ for the non-zero pressure man-at-eternal-rest problem from Section 3.3.2. The results using the non-zero pressure man-at-eternal-rest well-balanced scheme (top row) are compared with the results when using the traditional DG scheme (bottom row).

			Traditional DG Scheme		MAER WB DG Scheme	
J	Variable	Error Type	L^1 Error	L^∞ Error	L^1 Error	L^∞ Error
50	A	Absolute	6.5745e-09	3.2637e-05	6.9075e-17	1.7727e-13
		Relative	2.0676e-07	1.0273e-03	2.1538e-15	5.5236e-12
	Q	Absolute	6.9733e-13	7.5473e-08	2.0062e-17	5.8616e-14
200	A	Absolute	2.1580e-10	4.1932e-06	8.1454e-17	8.3574e-13
		Relative	6.7907e-09	1.3198e-04	2.5398e-15	2.6044e-11
	Q	Absolute	2.1929e-15	1.3615e-09	2.5451e-17	2.7981e-13

Table 3.5: Table of absolute and relative L^1 and L^∞ errors for non-zero pressure man-at-eternal-rest (MAER) well-balanced test representing stenosis in Section 3.3.2. Errors are given for the traditional DG scheme and the well-balanced scheme for both $J = 50$ and $J = 200$ uniform spatial cells. The well-balanced scheme demonstrates the well-balanced property, while the traditional DG scheme does not have the well-balanced property.

3.3.3 Tests for Small Perturbations of the Man-at-Eternal-Rest Steady States

In this section, we examine multiple tests in which the initial conditions of a man-at-eternal-rest steady state are perturbed in a small region. The initial perturbation will split into two waves moving away from the source in opposite directions. We compare the man-at-eternal-rest well-balanced and traditional DG schemes to demonstrate the advantage of well-balanced methods in handling the propagation of these small perturbations.

Different wave propagation behaviors arise in arteries with constant and variable cross-sectional areas. We will first discuss the expected behaviors before exploring specific examples. Suppose an incident pulse is introduced at the left arterial end (with cross-sectional area A_1) and then travels towards the right (with cross-sectional area A_2). When the pulse crosses into the region of the vessel with cross-sectional area A_2 it generates a transmission pulse propagating in the same direction, and also a reflected pulse propagating to the left. The speed and wavelength of the reflected pulse is the same as the incident pulse

because both pulses are traveling in the same medium. The amplitude of the reflected pulse is smaller and can be either inverted or non-inverted depending on the shape of the cross-sectional area. The reflection pulse is inverted when $A_1 < A_2$, and non-inverted when $A_1 > A_2$. See Figure 3.4 for a visualization of this process. The ratio of the amplitude of the reflected pulse and the incident pulse, known as the reflection coefficient $\mathcal{R}$, can be computed exactly and is given by

$$\mathcal{R} = \frac{A_1/C_1 - A_2/C_2}{A_1/C_1 + A_2/C_2}. \tag{3.36}$$

where the Moens-Korteweg coefficients C_i corresponding to A_1 and A_2 are defined as $C_i = \sqrt{\frac{K\sqrt{A_i}}{2\rho\sqrt{\pi}}}$ for $i = 1, 2$. The transmission coefficient, $\mathcal{T}$, given by $\mathcal{T} = 1 + \mathcal{R}$ represents the ratio of the amplitude of the transmission pulse to the incident pulse.

The numerical example in Section 3.3.3 portrays an artery with constant cross-sectional area. Therefore, the waves resulting from the perturbation will propagate through the domain and after they exit the domain, the radii will return to the unperturbed state. On the other hand, the examples in Sections 3.3.3 and 3.3.3 represent arteries with non-constant cross-sectional area, resulting in the appearance of reflection and transmission pulses.

Wave Equation

We start with the following wave equation example with constant cross-section at rest, which has been studied in [41] by Delestre et al. It is a small perturbation test, in which an "approximate" solution can be found analytically. The initial conditions are given

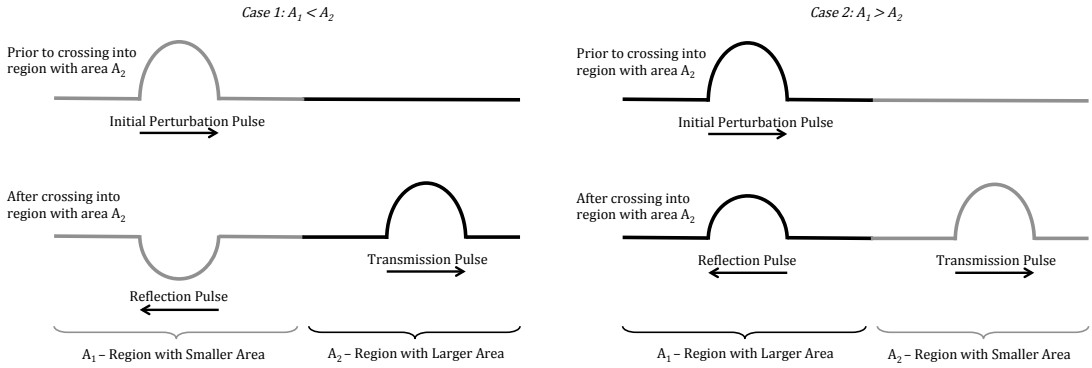


Figure 3.4: An initial perturbation pulse splits into a transmission and reflection pulse when it moves from a region of area A_1 to a region of area A_2. The transmission pulse continues in the same direction of the original pulse while the reflection pulse moves in the opposite direction.

by

$$A(x,0) = \begin{cases} \pi(R_0)^2, & \text{if } x \in [0,x_2] \cup [x_3,L], \\ \pi(R_0)^2\left[1+\epsilon\sin\left(\pi\frac{x-x_2}{x_1}\right)\right]^2, & \text{if } x \in [x_2,x_3], \end{cases} \qquad Q(x,0) = 0, \tag{3.37}$$

on the computational domain $[0, L]$. The cross-section at rest is given by $A_0(x) = \pi R_0(x)^2$.

The parameters used in our simulation are listed in Table 3.6.

R_0	K	ρ	x_1	x_2	x_3	L	ϵ
$4\times10^{-3}m$	$10^8\ Pa/m$	$1060\ kg/m^3$	$\frac{2L}{10}$	$\frac{4L}{10}$	$\frac{6L}{10}$	$0.16m$	5×10^{-3}

Table 3.6: Parameters for the wave equation problem (3.37).

As shown in [41], when neglecting all the high order terms of ϵ, its solution can be

expressed as

$$\begin{cases} R(x,t) &= R_0 + \frac{\epsilon}{2}[\phi(x - C_0 t) + \phi(x + C_0 t)], \\ u(x,t) &= -\epsilon \frac{C_0}{R_0}[-\phi(x - C_0 t) + \phi(x + C_0 t)], \end{cases} \tag{3.38}$$

where $\phi(x) = R_0 \sin\left(\pi \frac{x - x_2}{x_1}\right) \mathbb{1}_{[x_2, x_3]}$ with $\mathbb{1}$ being the indicator function and the constant C_0 is the Moens-Korteweg wave velocity

$$C_0 = \sqrt{\frac{K\sqrt{A_0}}{2\rho\sqrt{\pi}}} = \sqrt{\frac{KR_0}{2\rho}} \approx 13.73. \tag{3.39}$$

We employee transmissive boundary conditions at the endpoints of the domain. Figure 3.5 shows the numerical results at times $t = 0.002, 0.004$, and 0.006 with a mesh of 200 cells. Comparison with the analytical exact solution (3.38) demonstrates that these small perturbations are well captured.

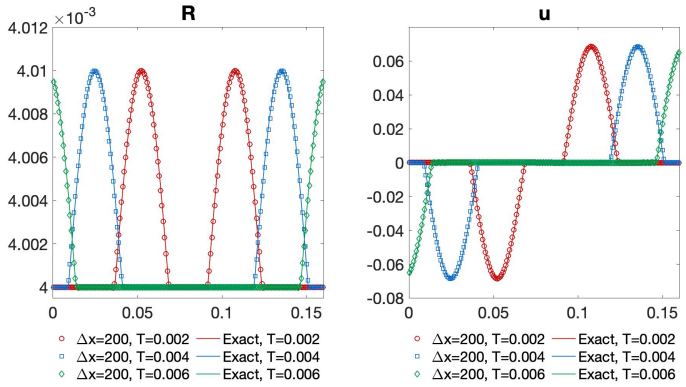


Figure 3.5: Solutions of the radius R and velocity u of the wave equation problem at various times with quadratic basis functions and a uniform mesh of 200 cells and the exact solutions at the same times.

Propagation of a Pulse to and from an Expansion

In the examples below, we consider the reflection and the transmission of a small wave in an aneurysm, when the cross-section at rest is not a constant. Following the setup in [41], we consider the radius of the cross-section at rest given by

$$R_0(x) = \begin{cases} \tilde{R} + \Delta R, & \text{if } x \in [0, x_1], \\ \tilde{R} + \frac{\Delta R}{2}\left[1 + \cos\left(\frac{x - x_1}{x_2 - x_1}\pi\right)\right], & \text{if } x \in [x_1, x_2], \\ \tilde{R}, & \text{otherwise,} \end{cases} \tag{3.40}$$

where the necessary parameters are listed in Table 3.7.

$\tilde{R}$	ΔR	K	ρ	x_1	x_2	L
$4 \times 10^{-3} m$	$10^{-3} m$	10^8 Pa/m	1060 kg/m^3	$\frac{19L}{40}$	$\frac{L}{2}$	$0.16m$

Table 3.7: Parameters for (3.40) in the propagation of a pulse to and from an expansion problems.

First, we consider a pulse propagating towards an expansion. The perturbation is applied to the region of the artery with smaller radii and given by the following

$$R(x,0) = \begin{cases} R_0(x)\left[1 + \epsilon \sin\left(\frac{100}{20L}\pi\left(x - \frac{65L}{100}\right)\right)\right], & \text{if } x \in \left[\frac{65L}{100}, \frac{85L}{100}\right], \\ R_0(x), & \text{otherwise.} \end{cases} \tag{3.41}$$

The parameter $\epsilon = 5.0 \times 10^{-3}$ and the momentum $Q(x,0) = 0$ m^3/s. The boundary conditions are transmissive at the endpoints of the domain. The initial state and numerical solutions at times $t = 0.002$ and $t = 0.006$ are presented in Figure 3.6. Figure 3.7 demonstrates how the wave propagates as a function of time for all time.

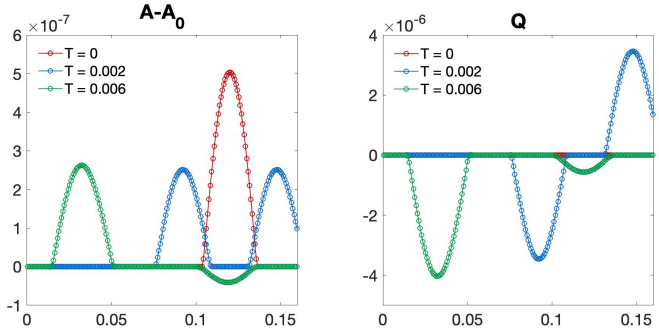


Figure 3.6: Initial conditions and solution of the propagation of a pulse to an expansion problem at various times with a mesh of 200 uniform cells. The reflection pulse, visible at time $t = 0.006$, is inverted.

Second, we consider a pulse propagation from an expansion. In this case, the perturbation is initiated in the region of the vessel with larger area. The perturbed radius now becomes

$$R(x,0) = \begin{cases} R_0(x)\left[1 + \epsilon \sin\left(\dfrac{100}{20L}\pi\left(x - \dfrac{15L}{100}\right)\right)\right], & \text{if } x \in \left[\frac{15L}{100}, \frac{35L}{100}\right], \\ R_0(x), & \text{otherwise,} \end{cases} \tag{3.42}$$

where $\epsilon = 5.0 \times 10^{-3}$. As before, the fluid is at rest, $Q(x,0) = 0$ $^{m^3}/_s$, and transmissive boundary conditions are imposed at the endpoints of the domain. In Figure 3.8, we see the initial state and numerical results at times $t = 0.002$ and $t = 0.006$. Figure 3.9 demonstrates how the wave propagates as a function of time for all time.

Perturbation of a Non-Zero Pressure Man-at-Eternal-Rest Well-Balanced Problem

In this subsection, we impose a small perturbation to a non-zero pressure man-at-eternal-rest steady state problem representing stenosis in a 'dead man'. We show that the

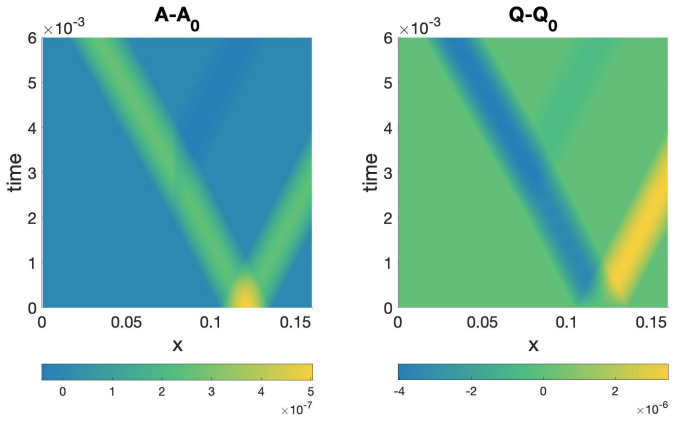


Figure 3.7: Propagation of the pulse to an expansion over all time. The plots show the difference between the numerical solution at time t and the initial conditions. A mesh of 200 uniform cells was used to compute the solution. Around time $t = 0.003$ the left-moving wave meets the expansion and the inverted reflection wave forms.

well-balanced scheme aptly handles the perturbation. We also compute the same test using the traditional DG scheme and compare the results.

The radius at rest is given by (3.34) and the original initial conditions are given by (3.35). We impose a small perturbation at the center of the artery to the cross-sectional radii in the following manner,

$$R_{pert}(x,0) = \begin{cases} R_0(x)\left[1+\epsilon\sin\left(\frac{100}{10L}\pi\left(x-\frac{45L}{100}\right)\right)\right], & \text{if } x \in \left[\frac{45L}{100}, \frac{55L}{100}\right], \\ R_0(x), & \text{otherwise.} \end{cases} \tag{3.43}$$

The initial condition for the perturbed cross-sectional area is then defined as $A_{pert}(x,0) = \pi R_{pert}(x,0)^2$. The scheme is run until time $t = 8 \times 10^{-4}$, before the traveling perturbation waves exit the domain.

We test this problem with $\epsilon = 10^{-3}$ and $\epsilon = 10^{-4}$ for two different sizes of mesh, $J = 50$ and $J = 200$ uniform cells. The results for the well-balanced DG scheme are presented in Figure 3.10 and the results for the traditional DG scheme are found in Figure

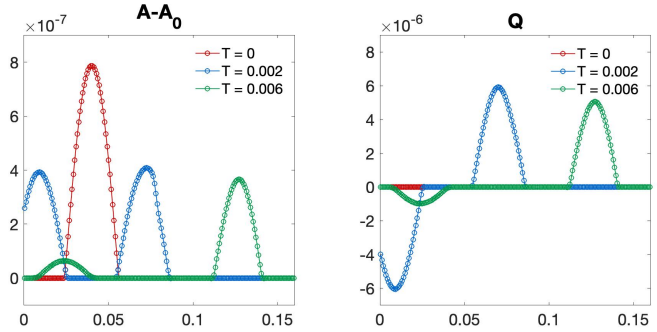


Figure 3.8: Initial conditions and solution of the propagation of a pulse from an expansion problem at various times computed with a mesh of 200 uniform cells. The reflection pulse, visible at time $t = 0.006$, is non-inverted.

3.11. The well-balanced scheme aptly handles the perturbation for either mesh size and for either size perturbation. On the other hand, the traditional DG scheme does not work as well. In the case where $J = 50$ uniform cells, then the undesirable behavior arises for either perturbation size. In the case where $J = 200$ uniform cells, the scheme has similar results for the perturbation with smaller amplitude (i.e. $\epsilon = 10^{-4}$), however the scheme improves when the amplitude of the perturbation is larger (i.e. $\epsilon = 10^{-3}$). However, in either case, the scheme is still out-performed by the well-balanced method.

3.4 Living-Man Well-Balanced DG Methods

In this section, well-balanced methods for maintaining the general living-man steady state (3.5) will be described. Due to the complexity of the steady state, extra attention is given to the projection of the initial conditions, as well as the source term and numerical flux calculations.

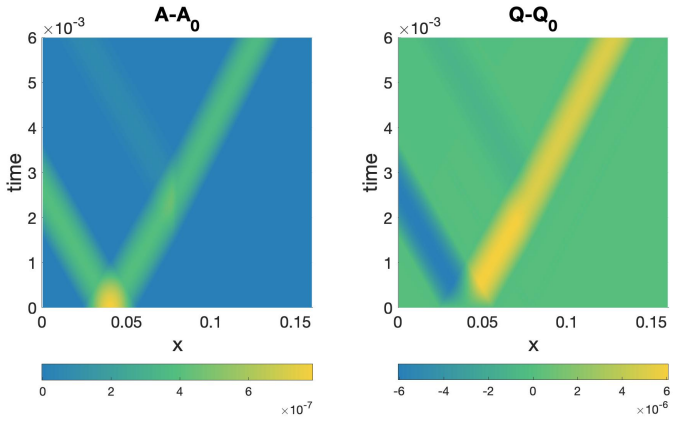
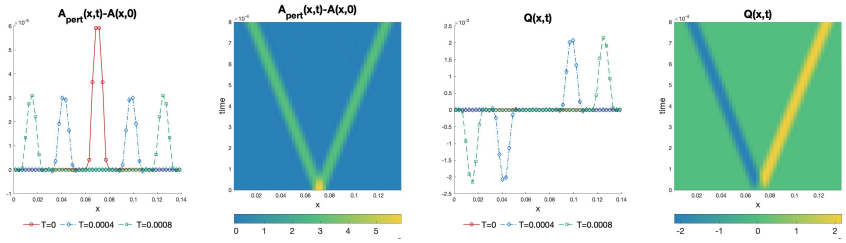


Figure 3.9: Propagation of the pulse from an expansion over all time. The plots show the difference between the numerical solution at time t and the initial conditions. A mesh of 200 uniform cells was used to compute the solution. Around time $t = 0.003$ the right-moving wave meets the expansion and the non-inverted reflection wave forms.

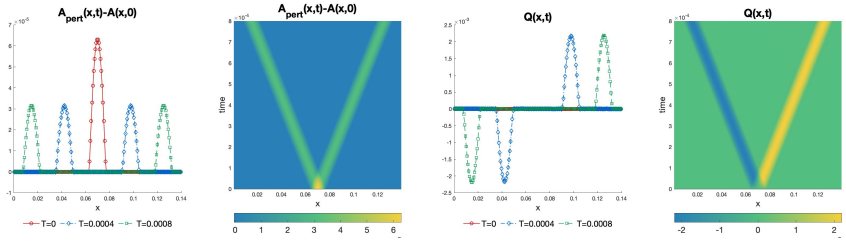
3.4.1 Numerical Initial Conditions

In general, the L^2 projection of the true initial condition $\boldsymbol{U}_0$ is taken to be the numerical initial condition $\boldsymbol{U}_\tau^0$ for modal DG schemes, as was done for the man-at-eternal-rest well-balanced DG scheme in Section 3.2. However, the projected polynomial $\boldsymbol{U}_\tau^0$ may not be in the equilibrium state. Thus the cell boundary values $\boldsymbol{U}^{\pm}_{\tau,j+\frac{1}{2}}$, as well as the function values at the quadrature points used to evaluate the volume integral, may also not be in equilibrium. This contributes to the challenges of how to recover the equilibrium information from these polynomials. We would like to comment that this difficulty disappears for the finite difference methods, because the points values of the initial condition in any finite difference methods, by design, automatically satisfy the equilibrium.

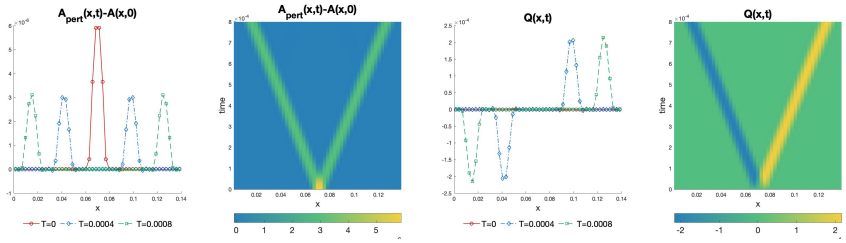
The same difficulty also appears in high order well-balanced finite volume methods, whose numerical initial condition is simply the cell average. In [110] where well-balanced methods were designed for the SWEs with moving-water equilibrium state, this difficulty



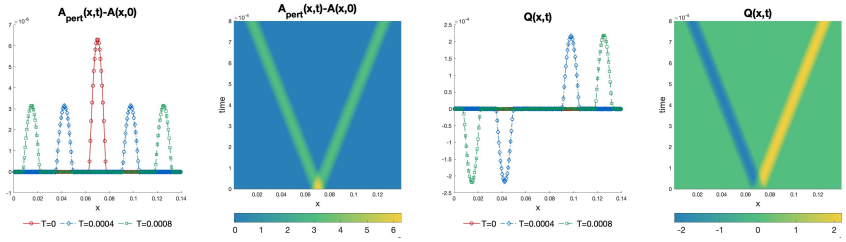
(a)$J = 50$ cells, $\epsilon = 10^{-3}$



(b)$J = 200$ cells, $\epsilon = 10^{-3}$

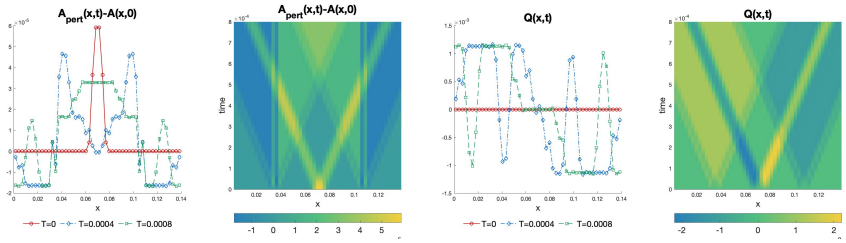


(c)$J = 50$ cells, $\epsilon = 10^{-4}$

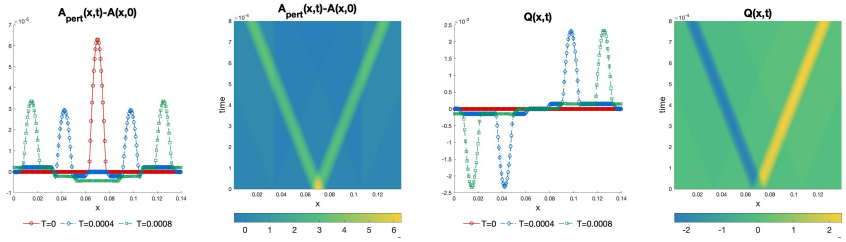


(d)$J = 200$ cells, $\epsilon = 10^{-4}$

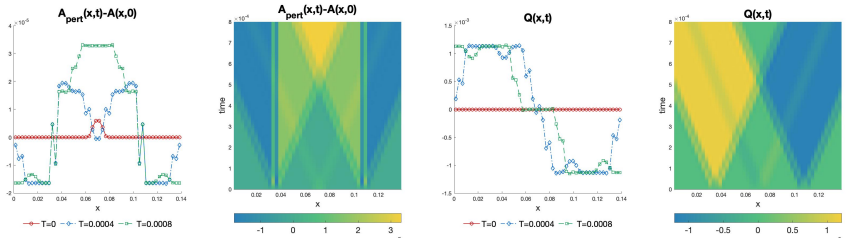
Figure 3.10: Perturbation of non-zero pressure man-at-eternal-rest well-balanced problem in Section 3.3.3 for all time up until $t = 0.008$ using the *non-zero pressure well-balanced DG scheme*. The scheme performs well for both mesh sizes and for both perturbation sizes.



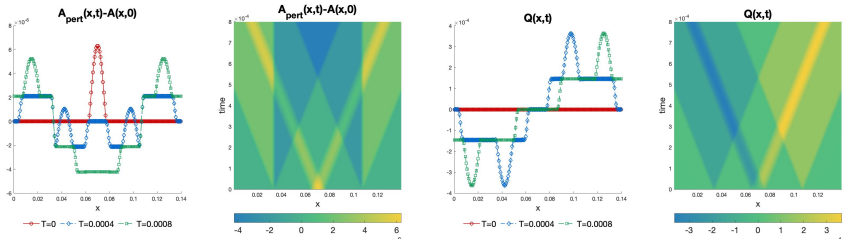
(a)$J = 50$ cells, $\epsilon = 10^{-3}$



(b)$J = 200$ cells, $\epsilon = 10^{-3}$



(c)$J = 50$ cells, $\epsilon = 10^{-4}$



(d)$J = 200$ cells, $\epsilon = 10^{-4}$

Figure 3.11: Perturbation of non-zero pressure man-at-eternal-rest well-balanced problem in Section 3.3.3 for all time up until $t = 0.008$ using the *traditional DG scheme*. The scheme performs poorly on the coarse mesh of 50 uniform cells for either size perturbation. For the refined mesh of 200 uniform cells, the scheme improves slightly for the larger perturbation, but still does not perform as well as the well-balanced scheme.

was identified. The issue was addressed in that paper by defining the well-balanced states as the solutions of nonlinear equations and then solving them using Newton's method. The same idea was later extended to construct numerical initial conditions of well-balanced DG methods in [150]. A simpler approach, without involving the nonlinear equations and the Newton's method, is proposed in [95], by introducing a special projection of the initial condition to take advantage of the flexibility of the DG method. A further modification of the projection from [95] is proposed in this chapter to produce the numerical initial condition $\boldsymbol{U}_\tau^0$.

We introduce the following projection $\mathbb{P}_\tau\omega$ of any function ω into the space $\mathbb{V}_\tau^k$ satisfying, on each interval I_j,

$$\int_{I_j} \mathbb{P}_\tau\omega v \, dx = \int_{I_j} \omega v \, dx, \tag{3.44}$$

for any $v \in P^{k-1}$ on I_j, and

$$(\mathbb{P}_\tau\omega)\left(x^-_{j+\frac{1}{2}}\right) = \omega\left(x^-_{j+\frac{1}{2}}\right), \tag{3.45}$$

at the right boundary value $x_{j+\frac{1}{2}}$ of the cell I_j. This projection is known as the Radau projection. The polynomial $\mathbb{P}_\tau\omega$ for each cell I_j can be determined by solving a local linear algebra problem of the size $(k+1)\times(k+1)$ derived from the discretized versions of (3.44) and (3.45). This is a local projection defined on each interval I_j. One can show that the error of this projection has optimal order of h^{k+1}.

The projections of the initial condition $\boldsymbol{U}_\tau^0$ and the cross-sectional area at rest $(A_0)_\tau(x)$ are defined to be

$$\boldsymbol{U}_\tau^0(x) = \mathbb{P}_\tau\boldsymbol{U}_0(x), \qquad (A_0)_\tau(x) = \mathbb{P}_\tau A_0(x). \tag{3.46}$$

At the right boundary point of each cell, it can be shown that

$$\boldsymbol{U}_\tau^0(x^-_{j+\frac{1}{2}}) = \boldsymbol{U}_0(x^-_{j+\frac{1}{2}}), \qquad (A_0)_\tau(x^-_{j+\frac{1}{2}}) = A_0(x^-_{j+\frac{1}{2}}), \qquad \text{for all } j, \tag{3.47}$$

which means that the equilibrium states (3.49) are recovered at these points

$$\left(\frac{u_\tau^2}{2} + \beta\left(\sqrt{A_\tau} - \sqrt{(A_0)_\tau}\right)\right)\left(x^-_{j+\frac{1}{2}}\right) = constant, \qquad \text{for all } j.$$

This information will be very useful when decomposing the solutions into the equilibrium and fluctuation parts in the following section.

Remark 16. *In [95], a slightly different projection was introduced to compute the numerical initial conditions. That was defined by the formula* (3.44)*, combined with*

$$(\mathbb{P}_\tau\omega)(x_j) = \omega(x_j), \tag{3.48}$$

which requires the projected polynomial overlaps with the original function at the center x_j of each cell I_j. However, this projection may not be optimal for some polynomial degree k, therefore we introduce a different projection $\mathbb{P}_\tau$ in this work. Note that the choice of this projection is not unique. Alternatively, we could have also chosen to fix the numerical initial conditions to be equal to the true solution at the left side of each computational cell.

3.4.2 Conservative, Equilibrium Variables and the Decomposition of Solutions

The living-man equilibrium variables from (3.5) will be denoted as

$$\boldsymbol{V} = \begin{pmatrix} Q \\ E \end{pmatrix} = \begin{pmatrix} Q \\ \frac{Q^2}{2A^2} + \beta\left(\sqrt{A} - \sqrt{A_0}\right) \end{pmatrix}. \tag{3.49}$$

We need to transform the conservative variables $\boldsymbol{U}$ to the equilibrium variables $\boldsymbol{V}$ and vice versa, during the construction of well-balanced numerical flux. The equilibrium variables can be easily computed from $\boldsymbol{U}$ and the cross-sectional area at rest A_0, and we denote it by $\boldsymbol{V} = \boldsymbol{V}(\boldsymbol{U}, A_0)$. On the other hand, suppose $\boldsymbol{V}$ and the cross-sectional area at rest A_0 are given, we can evaluate $\boldsymbol{U} = \boldsymbol{U}(\boldsymbol{V}, A_0)$ (or simply $A = A(\boldsymbol{V}, A_0)$ as Q can be directly obtained from $\boldsymbol{V}$) in the following way. The equilibrium variable E is defined as

$$E = \frac{Q^2}{2A^2} + \beta\left(\sqrt{A} - \sqrt{A_0}\right),$$

which is equivalent to

$$\beta A^{\frac{5}{2}} - \left(\beta\sqrt{A_0} + E\right)A^2 + \frac{1}{2}Q^2 = 0. \tag{3.50}$$

The conservative variable A can be recovered by finding the root of the equation (3.50). One can use Newton's method to find the root, by using $A_\tau(x_i)$ as the initial guess, where x_i is either a quadrature point or a cell-boundary value depending on where we are solving the problem. Müller et al. [105] address the recovery of A from the living-man equilibrium by solving the similar nonlinear equation and considering the subcritical, supercritical, or critical cases. In [60] by Ghigo et al., they assume that values for Q are small enough that living-man equilibrium variables (3.5) can be approximated by (3.3). This eliminates the need to recover A from a fractional-degree equality.

Next, we propose the decomposition of the solution $\boldsymbol{U}_\tau$ into the the reference equilibrium state $\boldsymbol{U}_\tau^e$ and the fluctuation state $\boldsymbol{U}_\tau^f$. The reference equilibrium values $\hat{\boldsymbol{V}}_j$ in cell I_j are defined as

$$\hat{\boldsymbol{V}}_j = \begin{pmatrix} \hat{Q}_j \\ \hat{E}_j \end{pmatrix} = \begin{pmatrix} Q_\tau\left(x^-_{j+\frac{1}{2}}\right) \\ E_\tau\left(x^-_{j+\frac{1}{2}}\right) \end{pmatrix}. \tag{3.51}$$

The equilibrium state $\boldsymbol{U}_\tau^e(x)$ can then be computed from these values and the true function $A_0(x)$ (not $(A_0)_\tau(x)$)

$$\boldsymbol{U}_{\tau,j}^e(x) = \begin{pmatrix} A_{\tau,j}^e(x) \\ \\ Q_{\tau,j}^e(x) \end{pmatrix} = \mathbb{P}_\tau \boldsymbol{U}\left(\hat{\boldsymbol{V}}_j, A_0(x)\right), \tag{3.52}$$

in each cell I_j. The projection $\mathbb{P}_\tau$ is used to ensure that $\boldsymbol{U}_\tau^e \in \mathbb{V}_\tau^k$, since the functions $\boldsymbol{U}\left(\hat{\boldsymbol{V}}_j, A_0(x)\right)$ may no longer be piecewise polynomials due to the nonlinear mapping. Finally, we can decompose the numerical solution $\boldsymbol{U}_\tau$ as

$$\boldsymbol{U}_\tau = \boldsymbol{U}_\tau^e + \boldsymbol{U}_\tau^f \tag{3.53}$$

with the fluctuation part $\boldsymbol{U}_\tau^f = \boldsymbol{U}_\tau - \boldsymbol{U}_\tau^e \in \mathbb{V}_\tau^k$. Note that this decomposition will be computed at each time step t^n. Both $\boldsymbol{U}_\tau^e$ and $\boldsymbol{U}_\tau^f$ will be used in the computing the well-balanced fluxes and source term approximations.

3.4.3 Well-Balanced Numerical Fluxes

As explained in Section 3.2, the well-balanced DG scheme takes the form of

$$\int_{I_j} \partial_t \boldsymbol{U}_\tau^n v \, dx - \int_{I_j} \boldsymbol{F}(\boldsymbol{U}_\tau^n) \partial_x v \, dx + \hat{\boldsymbol{F}}_{j+\frac{1}{2}}^l v_{j+\frac{1}{2}}^- - \hat{\boldsymbol{F}}_{j-\frac{1}{2}}^r v_{j-\frac{1}{2}}^+ = \int_{I_j} \boldsymbol{S}(\boldsymbol{U}_\tau^n, (A_0)_\tau) v \, dx. \tag{3.54}$$

However, to achieve living-man well-balanced method, different ways to construct the well-balanced numerical flux and source term from those in Section 3.2 are needed.

Here, we would like to avoid the usage of Gauss-Lobatto nodes, and use the idea of hydrostatic reconstruction to determine the numerical fluxes, following the study in [153, 150, 160] in the context of the SWEs.

Prior to redefining the boundary values and constructing our fluxes, we first define the unique cell interface value of A_0 as

$$(A_0)^*_{\tau,j+\frac{1}{2}} = \max\left((A_0)^+_{\tau,j+\frac{1}{2}}, (A_0)^-_{\tau,j+\frac{1}{2}}\right), \tag{3.55}$$

for all j. This choice of $(A_0)^*_{\tau,j+\frac{1}{2}}$ will aid in guaranteeing continuity across the cell interface at the equilibrium state for the conservative variables. Now we set the redefined boundary values to be

$$\begin{aligned} \boldsymbol{U}^{*,\pm}_{\tau,j+\frac{1}{2}} = \begin{pmatrix} A^{*,\pm}_{\tau,j+\frac{1}{2}} \\ Q^{*,\pm}_{\tau,j+\frac{1}{2}} \end{pmatrix} &= \begin{pmatrix} \max\left(0, A(\hat{\boldsymbol{V}}_j, (A_0)^*_{\tau,j+\frac{1}{2}})\right) \\ \hat{Q}^{\pm}_{\tau,j+\frac{1}{2}} \end{pmatrix} + \boldsymbol{U}^f_\tau(x^{\pm}_{j+\frac{1}{2}}) \\ &= \begin{pmatrix} \max\left(0, A(\hat{\boldsymbol{V}}_j, (A_0)^*_{\tau,j+\frac{1}{2}})\right) + A^f_\tau(x^{\pm}_{j+\frac{1}{2}}) \\ Q^{\pm}_{\tau,j+\frac{1}{2}} \end{pmatrix}, \end{aligned} \tag{3.56}$$

where the values $\hat{\boldsymbol{V}}_j$ are given in (3.51) and $\boldsymbol{U}^f_\tau$ is given in (3.53). When the system is in equilibrium, $A^f_\tau\big(x^{\pm}_{j+\frac{1}{2}}\big) = 0$ and $\hat{\boldsymbol{V}}_j = \hat{\boldsymbol{V}}_{j+1}$, hence $\boldsymbol{U}^{*,-}_{\tau,j+\frac{1}{2}} = \boldsymbol{U}^{*,+}_{\tau,j+\frac{1}{2}}$ for all j.

It would also be satisfactory to define $(A_0)^*_{\tau,j+\frac{1}{2}} = \min\left((A_0)^+_{\tau,j+\frac{1}{2}}, (A_0)^-_{\tau,j+\frac{1}{2}}\right)$, or other combinations of $(A_0)^{\pm}_{\tau,j+\frac{1}{2}}$ following the generalization of the hydrostatic reconstruction idea and designed to ensure continuity across the cell interfaces. One possible choice is to let $(A_0)^*_{\tau,j+\frac{1}{2}} = (A_0)^-_{\tau,j+\frac{1}{2}}$, then one can show that $A\left(\hat{\boldsymbol{V}}_j, (A_0)^*_{\tau,j+\frac{1}{2}}\right) = A^-_{\tau,j+\frac{1}{2}}$ which eliminates the need to employ Newton's method to recover A.

Lastly, the well-balanced numerical fluxes can be computed in the same way as was introduced for the man-at-eternal-rest well-balanced scheme

$$\begin{aligned} \hat{\boldsymbol{F}}^l_{j+\frac{1}{2}} &= \boldsymbol{f}\left(\boldsymbol{U}^{*,-}_{\tau,j+\frac{1}{2}}, \boldsymbol{U}^{*,+}_{\tau,j+\frac{1}{2}}\right) + \boldsymbol{F}\left(\boldsymbol{U}^{-}_{\tau,j+\frac{1}{2}}\right) - \boldsymbol{F}\left(\boldsymbol{U}^{*,-}_{\tau,j+\frac{1}{2}}\right), \\ \hat{\boldsymbol{F}}^r_{j-\frac{1}{2}} &= \boldsymbol{f}\left(\boldsymbol{U}^{*,-}_{\tau,j-\frac{1}{2}}, \boldsymbol{U}^{*,+}_{\tau,j-\frac{1}{2}}\right) + \boldsymbol{F}\left(\boldsymbol{U}^{+}_{\tau,j-\frac{1}{2}}\right) - \boldsymbol{F}\left(\boldsymbol{U}^{*,+}_{\tau,j-\frac{1}{2}}\right), \end{aligned} \tag{3.57}$$

where $\boldsymbol{f}(a,b)$ is a numerical flux, such as the Lax-Friedrichs flux defined in (1.18). Notice that when $\boldsymbol{U}^{*,+}_{\tau,j+\frac{1}{2}} = \boldsymbol{U}^{*,-}_{\tau,j+\frac{1}{2}}$, then the fluxes reduce to

$$\hat{\boldsymbol{F}}^{l}_{j+\frac{1}{2}} = \boldsymbol{F}(\boldsymbol{U}^{-}_{\tau,j+\frac{1}{2}}), \qquad \hat{\boldsymbol{F}}^{r}_{j-\frac{1}{2}} = \boldsymbol{F}(\boldsymbol{U}^{+}_{\tau,j-\frac{1}{2}}) \tag{3.58}$$

as a result of the numerical flux $\boldsymbol{f}(a,b)$ being consistent.

3.4.4 Well-Balanced Source Term Decomposition

The source term approximation for the living-man well-balanced scheme is approached in the same manner as in Section 3.2.2 for the man-at-eternal-rest well-balanced scheme. Due to the fact that the source term is linear with respect to the variable A, the source term can take on a decomposition similar to the form found in (3.53)

$$\int \boldsymbol{S}(\boldsymbol{U}_\tau, (A_0)_\tau)v \; dx = \int \boldsymbol{S}(\boldsymbol{U}^e_\tau, (A_0)_\tau)v \; dx + \int \boldsymbol{S}(\boldsymbol{U}^f_\tau, (A_0)_\tau)v \; dx. \tag{3.59}$$

The second term on the right hand side can be directly computed by any quadrature rule with sufficient accuracy. Since $\boldsymbol{U}^e_\tau$ is the equilibrium state, we can follow the discussion in Section 3.2.2 to approximate the first term. Thus the approximation for the source term takes the form

$$\begin{aligned}
&\int_{I_j} \boldsymbol{S}(\boldsymbol{U}_\tau, (A_0)_\tau)v \; dx \\
&\quad = -\int_{I_j} \boldsymbol{F}(\boldsymbol{U}^e_\tau)v_x \; dx + \boldsymbol{F}(\boldsymbol{U}^{e,-}_{\tau,j+\frac{1}{2}})v^-_{j+\frac{1}{2}} - \boldsymbol{F}(\boldsymbol{U}^{e,+}_{\tau,j-\frac{1}{2}})v^+_{j-\frac{1}{2}} + \int_{I_j} \boldsymbol{S}(\boldsymbol{U}^f_\tau, (A_0)_\tau)v \; dx.
\end{aligned} \tag{3.60}$$

The details of the derivation and some explanations can be found in Section 3.2.2.

3.4.5 A TVB Slope Limiter

In this section we discuss the implementation of a slope limiter, which is necessary when the solution contains discontinuities. We employ the modified minmod slope limiter with a total variation bounded (TVB) parameter M, defined as

$$\tilde{m}(a_1, ..., a_l) = \begin{cases} a_1, & \text{if } |a_1| \leq M\Delta x^2, \\ \text{minmod}(a_1, ..., a_l), & \text{otherwise,} \end{cases} \tag{3.61}$$

with the minmod function given by

$$\text{minmod}(a_1, ..., a_l) = \begin{cases} s \min(|a_1|, ..., |a_l|), & \text{if } s = \text{sign}(a_1) = ... = \text{sign}(a_l), \\ 0, & \text{otherwise .} \end{cases} \tag{3.62}$$

At each cell I_j, we define the modified cell boundary values to be

$$\begin{aligned} \boldsymbol{U}_\tau^{(\text{mod})}(x^-_{j+\frac{1}{2}}) &= \bar{\boldsymbol{U}}_{\tau,j} + \text{minmod}\Big(\boldsymbol{U}_\tau(x^-_{j+\frac{1}{2}}) - \bar{\boldsymbol{U}}_{\tau,j}, \bar{\boldsymbol{U}}^n_{\tau,j} - \bar{\boldsymbol{U}}^n_{\tau,j-1}, \bar{\boldsymbol{U}}^n_{\tau,j+1} - \bar{\boldsymbol{U}}^n_{\tau,j}\Big), \\ \boldsymbol{U}_\tau^{(\text{mod})}(x^+_{j-\frac{1}{2}}) &= \bar{\boldsymbol{U}}_{\tau,j} - \text{minmod}\Big(\bar{\boldsymbol{U}}_{\tau,j} - \boldsymbol{U}_\tau(x^+_{j-\frac{1}{2}}), \bar{\boldsymbol{U}}^n_{\tau,j} - \bar{\boldsymbol{U}}^n_{\tau,j-1}, \bar{\boldsymbol{U}}^n_{\tau,j+1} - \bar{\boldsymbol{U}}^n_{\tau,j}\Big), \end{aligned} \tag{3.63}$$

where $\bar{\boldsymbol{U}}^n_j$ is the cell average in cell I_j at time t^n. Note that the slope limiting procedure may not be required in every cell. If $\tilde{m}(a_1, ..., a_l) = a_1$ (i.e., $\boldsymbol{U}_\tau^{(\text{mod})}\left(x^{\pm}_{j\mp\frac{1}{2}}\right) = \boldsymbol{U}_\tau\left(x^{\pm}_{j\mp\frac{1}{2}}\right)$) in cell I_j, which implies that the solution is smooth, then limiting is not necessary in that cell. Otherwise, limiting is required and we can recover a new P^k polynomial $\boldsymbol{U}^n_{\tau,j}(x)$ from the cell average $\bar{\boldsymbol{U}}^n_{\tau,j}$ and the updated cell boundary values (3.63) for $k \geq 2$ that preserves the original cell average in I_j. This new polynomial then replaces the old one in this cell and will be used in the computation.

Note that when the living-man equilibrium state (3.5) is reached, the equilibrium may not be preserved if the slope limiter is activated. Therefore we wish to carefully

determine which cells need limiting by applying the above procedure to the fluctuation part of the variables $\boldsymbol{U}_\tau^f$. If limiting is required in a cell, the slope limiting procedure is then performed on $\boldsymbol{U}_\tau$. Recall that when a steady state is reached, $\boldsymbol{U}_{\tau,j}^f = 0$ for all values of j. Therefore, limiting is not required and the limiter has no affect on the well-balanced property.

It is important to note this procedure meets the necessary conditions of a limiter: it does not change the solution in smooth and well-balanced regions, and it does not change cell averages (hence maintains the mass conservation property of the DG method).

3.4.6 Verification of the Living-Man Well-Balanced Property

Proposition 17. *The proposed RKDG scheme (2.10) with numerical fluxes (3.57) and source term approximation (3.60) is exactly well-balanced for the living-man equilibrium (3.5).*

Proof. We assume the initial data are in the living-man equilibria state, and the same analysis applies when the solution reaches the living-man equilibria at the time step t^n. The projection $\mathbb{P}_\tau$ guarantees the numerical initial conditions will satisfy the equilibrium condition at the values $x_{j+\frac{1}{2}}^-$ for all j. This implies that $\hat{\boldsymbol{V}}_j = constant$ for all j. Therefore, the equilibrium part, $\boldsymbol{U}_\tau^e$, computed from $\hat{\boldsymbol{V}}_j$ and $A_0(x)$ is equivalent to the conservative variable $\boldsymbol{U}_\tau$, and this further implies that $\boldsymbol{U}_\tau^f = 0$. The source term approximation (3.60) can then be rewritten as

$$\int_{I_j} \boldsymbol{S}(\boldsymbol{U}_\tau, (A_0)_\tau) v \, dx = -\int_{I_j} \boldsymbol{F}(\boldsymbol{U}_\tau) v_x \, dx + \boldsymbol{F}(\boldsymbol{U}_{\tau,j+\frac{1}{2}}^-) v_{j+\frac{1}{2}}^- - \boldsymbol{F}(\boldsymbol{U}_{\tau,j-\frac{1}{2}}^+) v_{j-\frac{1}{2}}^+ \tag{3.64}$$

The modified cell boundary values become

$$\boldsymbol{U}^{*,-}_{\tau,j+\frac{1}{2}} = \begin{pmatrix} \max\left(0, A(\bar{\boldsymbol{V}}_j, (A_0)^*_{j\pm\frac{1}{2}})\right) \\ \\ \bar{m}^+_{j\pm\frac{1}{2}} \end{pmatrix} = \boldsymbol{U}^{*,+}_{\tau,j+\frac{1}{2}}, \tag{3.65}$$

as a result of $\boldsymbol{U}^f_\tau = 0$. Due to the consistency of the Lax-Friedrichs flux and the definition of the left and right fluxes, it can be shown that

$$\hat{\boldsymbol{F}}^l_{j+\frac{1}{2}} = \boldsymbol{F}(\boldsymbol{U}^-_{\tau,j+\frac{1}{2}}), \qquad \hat{\boldsymbol{F}}^r_{j-\frac{1}{2}} = \boldsymbol{F}(\boldsymbol{U}^+_{\tau,j-\frac{1}{2}}) \tag{3.66}$$

at the steady state. Therefore, one can easily observe that the flux terms exactly balance the source term approximation, which shows the well-balanced property. ■

This section concludes with two remarks about the well-balanced RKDG methods for the arterial blood flow model when the cross-sectional area at rest is constant, and the comparison of living-man and the man-at-eternal-rest well-balanced methods.

Remark 18. *When the cross-sectional area at rest, A_0, is constant, the traditional DG scheme is recovered, that is, the source term approximation reduces to 0 and the left and right numerical fluxes reduce to the original fluxes. First, we look at the source term. Definition* (3.52) *implies that $\boldsymbol{U}^e_{\tau,j} = \mathbb{P}_\tau \boldsymbol{U}(\hat{\boldsymbol{V}}_j, A_0) =$ constant in each cell I_j when $A_0 =$ constant. Therefore, it can be shown that*

$$\int_{I_j} \boldsymbol{S}\left(\boldsymbol{U}^e_\tau, (A_0)_\tau\right) v \, dx = -\int_{I_j} \boldsymbol{F}\left(\boldsymbol{U}^e_\tau\right) v_x \, dx + \boldsymbol{F}\left(\boldsymbol{U}^{e,-}_{\tau,j+\frac{1}{2}}\right) v^-_{j+\frac{1}{2}} - \boldsymbol{F}\left(\boldsymbol{U}^{e,+}_{\tau,j-\frac{1}{2}}\right) v^+_{j-\frac{1}{2}} = 0.$$

Additionally, $((A_0)_\tau)_x = 0$, hence, the numerical integral $\int_{I_j} \boldsymbol{S}\left(\boldsymbol{U}^f_\tau, (A_0)_\tau\right) v \, dx = 0$. Together, this implies the source term approximation is zero.

Second, we will show that the left and right numerical fluxes reduce to the original DG fluxes. When A_0 is constant, then $(A_0)^*_{\tau,j+\frac{1}{2}} = (A_0)^+_{\tau,j+\frac{1}{2}} = (A_0)^-_{\tau,j+\frac{1}{2}}$ which implies that

$A\left(\hat{\boldsymbol{V}}_j, (A_0)^*_{\tau,j+\frac{1}{2}}\right)$ *recovers the original value of* A^e_τ *at the cell interface exactly. Therefore,*

$$A^{*,\pm}_{\tau,j+\frac{1}{2}} = \max\left(0, A(\hat{\boldsymbol{V}}_j, (A_0)^*_{\tau,j+\frac{1}{2}}) + A^{f,\pm}_{\tau,j+\frac{1}{2}}\right) = \max\left(0, A^{e,\pm}_{\tau,j+\frac{1}{2}} + A^{f,\pm}_{\tau,j+\frac{1}{2}}\right) = A^{\pm}_{\tau,j+\frac{1}{2}}. \tag{3.67}$$

More generally that $\boldsymbol{U}^{*,\pm}_{\tau,j+\frac{1}{2}} = \boldsymbol{U}^{\pm}_{\tau,j+\frac{1}{2}}$. *Therefore the left and right numerical fluxes reduce to the original DG fluxes:* $\hat{\boldsymbol{F}}^l_{j+\frac{1}{2}} = \hat{\boldsymbol{F}}_{j+\frac{1}{2}}$, $\hat{\boldsymbol{F}}^r_{j-\frac{1}{2}} = \hat{\boldsymbol{F}}_{j-\frac{1}{2}}$.

Remark 19. *Although the well-balanced methods presented in this section were designed to preserve the living-man equilibria, it also preserve the simpler man-at-eternal-rest steady state. Notice that when* $Q = 0$, *the equilibrium values* $\hat{\boldsymbol{V}}_j$ *become*

$$\hat{\boldsymbol{V}}_j = \begin{pmatrix} \hat{Q}_j \\ \hat{E}_j \end{pmatrix} = \begin{pmatrix} 0 \\ \beta\left(\sqrt{A_\tau} - \sqrt{(A_0)_\tau}\right)\left(x^-_{j+\frac{1}{2}}\right) \end{pmatrix}, \tag{3.68}$$

and the decomposition of $\boldsymbol{U}$, *as in (3.52) and (3.53) becomes*

$$\boldsymbol{U}^e_{\tau,j} = \begin{pmatrix} \mathbb{P}_\tau\left(\frac{\hat{E}_j}{\beta} + \sqrt{(A_0)_\tau}\right)^2 \\ \hat{Q}_j \end{pmatrix}, \qquad \boldsymbol{U}^f_{\tau,j} = \begin{pmatrix} A_{\tau,j} \\ Q_{\tau,j} \end{pmatrix} - \begin{pmatrix} \mathbb{P}_\tau\left(\frac{\hat{E}_j}{\beta} + \sqrt{(A_0)_\tau}\right)^2 \\ \hat{Q}_j \end{pmatrix}. \tag{3.69}$$

Therefore, the living-man well-balanced scheme reduces to the man-at-eternal-rest well-balanced scheme presented in Section 3.2.3.

3.5 Numerical Tests for the Living-Man Well-Balanced Methods

In this section, we present numerical results for the one-dimensional blood flow system (1.15) using the generalized living-man well-balanced methods described in Section 3.4. We implement our scheme using piecewise quadratic polynomials ($k = 2$) paired with

the third order TVD Runge-Kutta time discretization (1.19). The CFL number is take to be 0.15 and the constant M in the TVB limiter is taken to be 0, unless otherwise stated. Multiple types of tests: accuracy test, well-balanced test, perturbations of steady states, and tests for discontinuous solutions, are presented in this section.

3.5.1 Test for Accuracy

In this section, we will test the accuracy of our living-man well-balanced DG scheme for smooth solutions with non-zero velocity. We have chosen strictly positive functions for both A and A_0 to avoid difficulties with square roots and division by a small number. The initial conditions in the domain $x \in [0, 10]$ are given by

$$A(x,0) = \sin\left(\frac{\pi}{5}x\right) + 10, \qquad Q(x,0) = e^{\cos(\frac{\pi}{5}x)}, \tag{3.70}$$

with the cross-sectional area at rest

$$A_0(x) = \frac{1}{2}\cos^2\left(\frac{\pi}{5}x\right) + 5,$$

and $K = 10^8\ {}^{Pa}/_{m}$, $\rho = 1060\ {}^{kg}/_{m^3}$. Periodic boundary conditions are employed in this test. We run the simulation until time $t = 0.01$ when the solution is still smooth. Since there is no explicitly known solution in this case, the errors are computed by comparing results from meshes of uniform cell widths h and $h/2$. Table 3.8 contains the L^1 errors and numerical orders of accuracy for P^0, P^1 and P^2 polynomials. For each polynomial degree k, $(k+1)^{\text{th}}$ order is observed, which indicates the optimal convergence rate is achieved.

		$k=0$		$k=1$		$k=2$	
Variable	J	L^1 Error	Order	L^1 Error	Order	L^1 Error	Order
	25	6.1773e-01		4.0942e-02		1.7301e-03	
	50	4.0751e-01	0.6002	1.0068e-02	2.0237	2.2514e-04	2.9420
A	100	2.4607e-01	0.7278	2.5003e-03	2.0097	2.8747e-05	2.9693
	200	1.3667e-01	0.8484	6.2353e-04	2.0036	3.6285e-06	2.9860
	400	7.2197e-02	0.9207	1.5570e-04	2.0017	4.5571e-07	2.9932
	25	2.5275e02		6.2138e00		3.4179e-01	
	50	1.4838e02	0.7684	1.5290e00	2.0229	4.1730e-02	3.0340
Q	100	8.0758e01	0.8776	3.8080e-01	2.0055	5.1098e-03	3.0297
	200	4.2223e01	0.9356	9.5164e-02	2.0005	6.2939e-04	3.0212
	400	2.1591e01	0.9676	2.3794e-02	1.9998	7.8033e-05	3.0118

Table 3.8: L^1 errors and convergence orders of the accuracy test in Section 3.5.1, using P^0, P^1 and P^2 piecewise polynomials and the living-man well-balanced method. In each case, $k+1$ order of accuracy is achieved.

3.5.2 Test for Well-Balanced Property

In this section, we will demonstrate that the proposed living-man well-balanced DG scheme preserves the steady state (3.5) with non-zero velocity. We will examine three examples that represent the physiological conditions of an aneurysm, stenosis, and a decreasing step. The initial conditions for each of the examples in this section will be determined from the equilibrium variables of the steady state, which take the form

$$Q_s = Q_{in}, \qquad E_s = \frac{Q_{in}^2}{2(A_{out})^2} + \beta\left(\sqrt{A_{out}} - \sqrt{(A_0(L)}\right), \tag{3.71}$$

where the subscript 'in' represents the value at the inlet or left side of the domain, 'out' represents the value at the outlet or the right side of the domain, and L is the length of the artery. The function for $A(x,0)$ can be determined from (3.71) and the cross-sectional area at rest, which is unique to each example.

The values of A_{in} and A_{out} are given by

$$A_{in} = A_0(0)[1+S_{in}]^2, \qquad A_{out} = A_0(L)[1+S_{in}]^2, \tag{3.72}$$

where S_{in} is the Shapiro number at the inlet. The Shapiro number is the equivalent of the Froude number for the SWEs and is determined by the formula $S = u/C$ where C is the Moens-Korteweg wave velocity. The Shapiro number determines whether the system is in subcritical ($S < 1$), critical ($S = 1$), or supercritical ($S > 1$) flow. Blood flow is typically subcritical, hence that is the only case we will consider in the numerical examples. We will consider $S_{in} = \{0.5, 0.1, 0.01\}$ in our numerical examples. The Moens-Korteweg velocity at the inlet is defined as

$$C_{in} = \sqrt{\frac{K\sqrt{A_{in}}}{2\rho\sqrt{\pi}}}. \tag{3.73}$$

Finally, the value for Q_{in} can then be determined as a function of the Shapiro number in the following way

$$Q_{in} = A_{in} S_{\tau,in} C_{in}. \tag{3.74}$$

We also introduce the notation ΔR to represent the wall deformation parameter. Other important parameters in this section are found in Table 3.9. In each of the examples, we fix the boundary conditions to be Q_{in} at the inlet and A_{out} at the outlet of the domain.

R_{in}	ΔR	K	ρ	L
$4 \times 10^{-3}\ m$	$1 \times 10^{-3}\ m$	$10^8\ Pa/m$	$1060\ kg/m^3$	$0.16\ m$

Table 3.9: Parameters for well-balanced living-man problems.

Table 3.10 contains the some of the important constants used in the following examples. The constants all depend on the Shapiro number at the inlet. It can be seen that the smaller the Shapiro number, the slower the discharge value Q_{in}. We expect the living-man well-balanced scheme to maintain the steady states with machine zero error. The man-at-eternal-rest well-balanced scheme may not be able to preserve the non-zero

velocity steady states. However, we expect the man-at-eternal-rest well-balanced scheme to perform better for lower Shapiro numbers because the lower the Shapiro number, the closer the living-man steady state will be towards the man-at-eternal-rest steady state (3.3) (i.e. zero velocity).

S_{in}	0.5	0.1	0.01
A_{in}	1.1310×10^{-4}	6.0821×10^{-5}	5.1276×10^{-5}
C_{in}	16.8232	14.4065	13.8046
Q_{in}	9.5133×10^{-4}	8.7622×10^{-5}	7.0784×10^{-6}

Table 3.10: The parameters used in the following examples that depend on the Shapiro number at the inlet, S_{in}. The smaller the Shapiro number S_{in}, the slower the discharge Q_{in}.

An Aneurysm

In this subsection, we consider the living-man equilibrium (3.5) with non-zero velocity where the choice of cross-sectional radii is meant to represent that of an aneurysm. We set the cross-sectional radii at rest to be

$$R_0(x) = \begin{cases} R_{in}, & \text{if } x \in [0, x_1] \cup [x_4, L], \\ R_{in} + \frac{\Delta R}{2}\left[1 - \cos\left(\frac{x-x_1}{x_2-x_1}\pi\right)\right], & \text{if } x \in [x_1, x_2], \\ R_{in} + \Delta R, & \text{if } x \in [x_2, x_3], \\ R_{in} + \frac{\Delta R}{2}\left[1 + \cos\left(\frac{x-x_3}{x_4-x_3}\pi\right)\right], & \text{if } x \in [x_3, x_4], \end{cases} \tag{3.75}$$

with $x_1 = \frac{9L}{40}, x_2 = \frac{L}{4}, x_3 = \frac{3L}{4}, x_4 = \frac{31L}{40}$ and the cross-sectional area at rest given by $A_0(x) = \pi R_0(x)^2$. The radii at rest is shown in Figure 3.12.

The living-man equilibrium state should be exactly preserved. We run the problem using a uniform mesh of 200 cells until time $t = 5$. The L^1 and L^∞ errors shown in Table 3.11

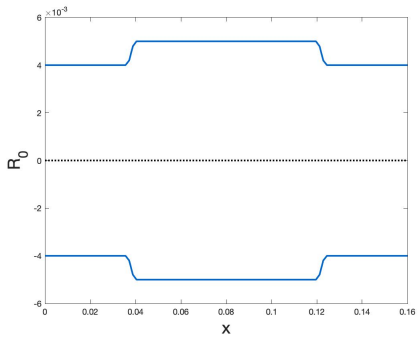
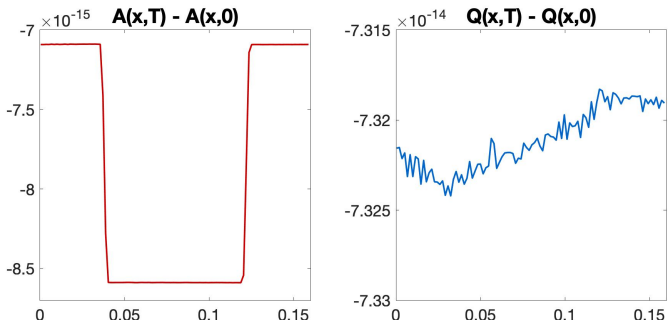


Figure 3.12: Radii at rest for the artery with an aneurysm defined by (3.75).

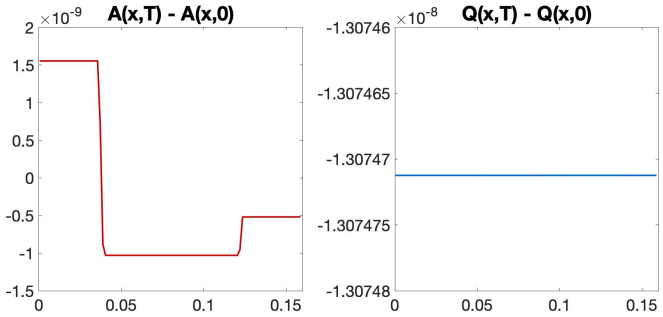
demonstrate that the well-balanced property is indeed maintained. We also demonstrate that the man-at-eternal-rest well-balanced DG methods presented in Section 3.2 cannot maintain this general steady state with non-zero velocity for larger Shapiro numbers. The corresponding L^1 and L^∞ errors are found in Table 3.11 as well. The difference between the numerical solution at the final time $t = 5$ and the numerical initial conditions is plotted in Figure 3.13, comparing both the living-man and man-at-eternal-rest well-balanced DG schemes when $S_{in} = 0.5$.

Stenosis

The function choice for the radius at rest representing aortic stenosis was first introduced in [60], and we changed to the parameters in this work so that the units of measure are consistent with the units used in all other examples. The definition for the



(a)Living-man well-balanced DG scheme



(b)Non-zero pressure man-at-eternal-rest well-balanced DG scheme

Figure 3.13: The difference between A, Q at the final time $t = 5$ and the corresponding numerical initial conditions for the artery with an aneurysm problem from Section 3.5.2. We compare the living-man well-balanced DG method (top row) and the man-at-eternal-rest well-balanced DG method (bottom row). Both plots were computed with a mesh of 200 uniform cells and $S_{in} = 0.5$. The man-at-eternal-rest method does not handle the non-zero velocity equilibria as well as the living-man scheme.

radius at rest is

$$R_0(x) = \begin{cases} R_{in}, & \text{if } x \in [0, x_1] \cup [x_2, L], \\ R_{in}\left(1 - \dfrac{\Delta R}{2}\left[1 + \cos\left(\pi + 2\pi\dfrac{x - x_1}{x_2 - x_1}\right)\right]\right), & \text{if } x \in [x_1, x_2], \end{cases} \tag{3.76}$$

where $x_1 = \frac{3L}{10}$ and $x_2 = \frac{7L}{10}$. The radii at rest is shown in Figure 3.14.

The problem is computed until the final time $t = 5$ with a uniform mesh of 200 cells using both the living-man and man-at-eternal-rest well-balanced schemes. The L^1 and

		L^1 Error			L^∞ Error		
Var.	Error	$S_{in} = 0.5$	$S_{in} = 0.1$	$S_{in} = 0.01$	$S_{in} = 0.5$	$S_{in} = 0.1$	$S_{in} = 0.01$
Living-Man Well-Balanced Scheme							
A	Abs.	1.01e-18	3.49e-19	3.33e-19	8.59e-15	2.99e-15	2.87e-15
	Rel.	7.36e-15	4.62e-15	5.15e-15	6.27e-11	4.01e-11	4.49e-11
Q	Abs.	9.37e-18	3.47e-19	1.47e-19	7.32e-14	2.74e-15	1.16e-15
	Rel.	9.85e-15	3.96e-15	2.07e-14	7.70e-11	3.12e-11	1.64e-10
Man-at-Eternal-Rest Well-Balanced Scheme							
A	Abs.	1.56e-13	4.18e-15	4.05e-17	1.90e-08	2.46e-10	1.98e-12
	Rel.	1.17e-09	5.47e-11	6.16e-13	1.60e-04	3.83e-06	3.57e-08
Q	Abs.	1.95e-12	5.03e-14	4.56e-16	1.98e-07	4.01e-09	3.46e-11
	Rel.	2.04e-09	5.74e-10	6.45e-11	2.08e-04	4.58e-05	4.89e-06

Table 3.11: Table of absolute and relative L^1 and L^∞ errors for aneurysm problem in Section 3.5.2, using the living-man well-balanced scheme and the man-at-eternal-rest well-balanced scheme. The living-man scheme demonstrates the well-balanced property for each value of S_{in}. The man-at-eternal-rest DG scheme does not preserve the more general non-zero equilibrium state, but does improve as S_{in}, and thus Q_{in}, decreases.

L^∞ errors shown in Table 3.12 and Figure 3.15 displays the numerical solutions via the living-man and the man-at-eternal-rest schemes at the final time with $S_{in} = 0.5$.

A Decreasing Step

The example in this section represents blood flow from a parent to a daughter artery in which the transition is idealized, that is, the artery radii instantaneous changes from one value to a smaller value. The function choice for the radius at rest representing a decreasing step was first introduced in [60]. The radius at rest is given by

$$R_0(x) = \begin{cases} R_{in} & \text{if } x < \frac{L}{2}, \\ R_{in}\,(1 - \Delta R) & \text{if } x \geq \frac{L}{2}. \end{cases} \tag{3.77}$$

The radii at rest is shown in Figure 3.16.

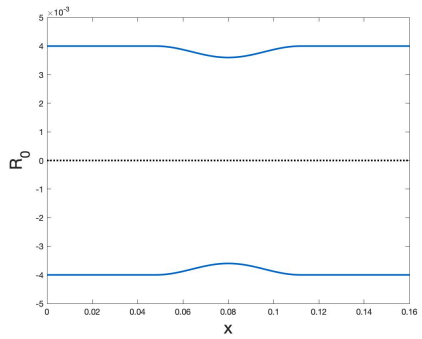
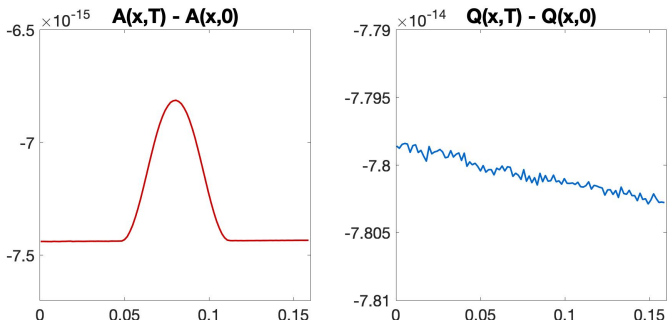


Figure 3.14: Radii at rest for the artery with stenosis defined by (3.76).

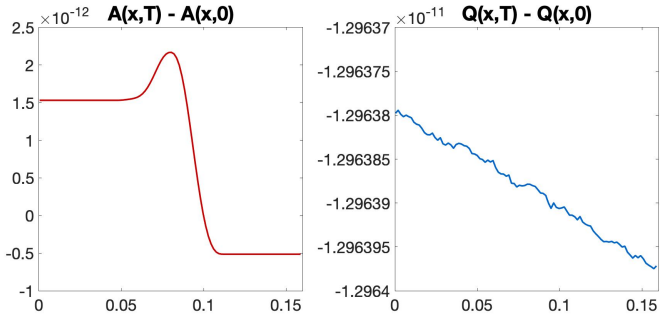
		L^1 Error			L^∞ Error		
Var.	Error	$S_{in} = 0.5$	$S_{in} = 0.1$	$S_{in} = 0.01$	$S_{in} = 0.5$	$S_{in} = 0.1$	$S_{in} = 0.01$
Living-Man Well-Balanced Scheme							
A	Abs.	9.35e-19	2.93e-19	2.56e-19	7.44e-15	2.33e-15	2.04e-15
	Rel.	8.61e-15	5.00e-15	5.21e-15	7.42e-11	4.22e-11	4.42e-11
Q	Abs.	9.98e-18	5.82e-19	1.17e-20	7.80e-14	4.56e-15	9.62e-17
	Rel.	1.05e-14	6.65e-15	1.66e-15	8.20e-11	5.20e-11	1.36e-11
Man-at-Eternal-Rest Well-Balanced Scheme							
A	Abs.	7.15e-16	4.94e-18	1.84e-19	6.99e-11	3.17e-13	2.76e-15
	Rel.	7.21e-12	8.70e-14	3.73e-15	7.31e-07	6.04e-09	6.23e-11
Q	Abs.	6.26e-15	8.61e-17	8.47e-19	4.60e-10	4.67e-12	3.97e-14
	Rel.	6.58e-12	9.83e-13	1.20e-13	4.83e-07	5.32e-08	5.61e-09

Table 3.12: Table of absolute and relative L^1 and L^∞ errors for the stenosis problem in Section 3.5.2, using the living-man well-balanced scheme and the man-at-eternal-rest well-balanced scheme.

The problem is computed using both the living-man and man-at-eternal-rest well-balanced schemes with a uniform mesh of 200 cells until the final time $t = 5$. The L^1 and L^∞ errors shown in Table 3.13 demonstrate that the well-balanced property is indeed maintained for the living-man scheme, but not for the man-at-eternal-rest scheme. Figure 3.17 displays the numerical solutions using both schemes with $S_{in} = 0.5$. Again, we observe that the man-at-eternal-rest well-balanced DG methods cannot maintain this general steady



(a)Living-man well-balanced DG scheme



(b)Non-zero pressure man-at-eternal-rest well-balanced DG scheme

Figure 3.15: The difference between A, Q at the final time $t = 5$ and the corresponding numerical initial conditions for the artery with stenosis problem from Section 3.5.2, when using the living-man well-balanced DG method (top row) and the man-at-eternal-rest well-balanced DG method (bottom row). Both plots were computed with a mesh of 200 uniform cells and $S_{in} = 0.5$.

state with non-zero velocity well.

3.5.3 Tests for Nearly Equilibrium Flows

In this section, numerical tests are provided to demonstrate that the living-man well-balanced DG scheme can aptly handle small perturbations to living-man steady states, and capture the nearly equilibrium flows well. We will also compare the performance of

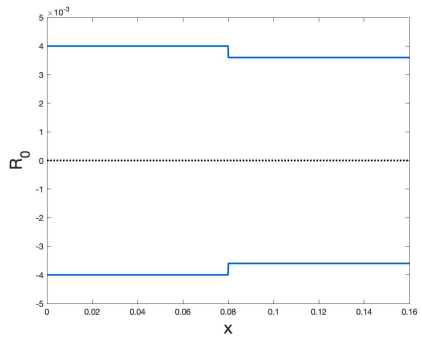
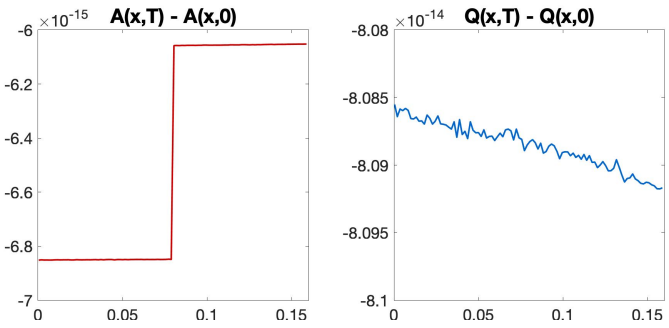


Figure 3.16: Radii at rest for the artery with a decreasing step defined by (3.77).

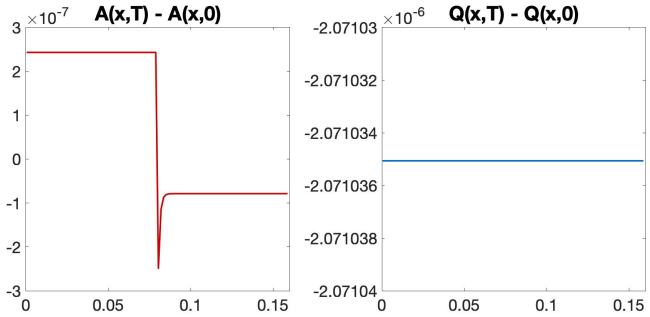
		L^1 Error			L^∞ Error		
Var.	Error	$S_{in} = 0.5$	$S_{in} = 0.1$	$S_{in} = 0.01$	$S_{in} = 0.5$	$S_{in} = 0.1$	$S_{in} = 0.01$
Living-Man Well-Balanced Scheme							
A	Abs.	8.26e-19	2.70e-19	1.87e-19	6.85e-15	2.21e-15	1.54e-15
	Rel.	8.10e-15	4.97e-15	4.04e-15	6.60e-11	4.07e-11	3.32e-11
Q	Abs.	1.04e-17	4.88e-19	1.12e-18	8.09e-14	3.82e-15	8.74e-15
	Rel.	1.09e-14	5.56e-15	1.58e-13	8.51e-11	4.36e-11	1.24e-09
Man-at-Eternal-Rest Well-Balanced Scheme							
A	Abs.	2.45e-11	1.03e-13	2.96e-16	4.06e-06	5.85e-08	4.77e-10
	Rel.	2.35e-07	1.92e-09	6.78e-12	4.42e-02	1.19e-03	1.15e-05
Q	Abs.	2.78e-10	1.37e-12	3.94e-15	2.77e-05	7.12e-07	6.14e-09
	Rel.	2.93e-07	1.57e-08	5.56e-10	2.92e-02	8.12e-03	8.68e-04

Table 3.13: Table of absolute and relative L^1 and L^∞ errors for the decreasing step test of in Section 3.5.2, using the living-man well-balanced scheme and the man-at-eternal-rest well-balanced scheme.

the living-man and man-at-eternal-rest well-balanced schemes. Since each example from Section 3.5.2 contains arteries with non-constant area, we expect to see the formation of transmission and reflection pulses when the perturbation wave crosses through a portion of the domain that changes shape. The values of the reflection and transmission coefficients in (3.36) become valid only for small $S_{\tau,in}$ since they were derived from linear analytic solutions and the flow is now nonlinear.



(a)Living-man well-balanced DG scheme



(b)Non-zero pressure man-at-eternal-rest well-balanced DG scheme

Figure 3.17: The difference between A, Q at the final time $t = 5$ and the corresponding numerical initial conditions for the decreasing step problem with $S_{in} = 0.5$ from Section 3.5.2, when using the living-man well-balanced DG method (top row) and the man-at-eternal-rest well-balanced DG method (bottom row).

Perturbation of A for the Aneurysm

We consider a small perturbation to the living-man equilibrium state for an artery with an aneurysm, which was described in Section 3.5.2. The initial conditions can be determined from (3.71) and the cross-sectional radii at rest given by (3.75). The original initial condition for the cross-sectional area is denoted $A(x, 0)$ and we denote the perturbed initial conditions of A by

$$A_{pert}(x, 0) = A(x, 0) + \pi p(x)^2,$$

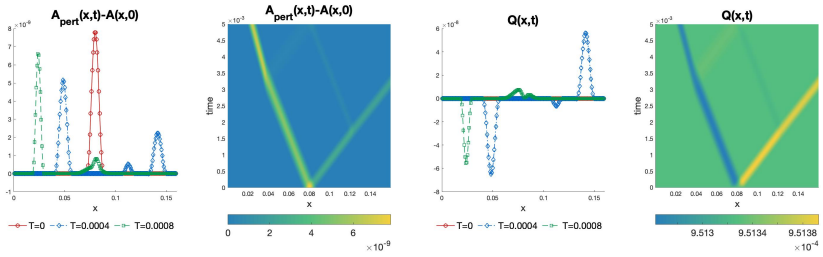
where

$$p(x) = \begin{cases} \epsilon \sin\left(\frac{100}{10L}\pi\left(x - \frac{45L}{100}\right)\right), & \text{if } x \in \left[\frac{45L}{100}, \frac{55L}{100}\right], \\ 0, & \text{otherwise,} \end{cases} \tag{3.78}$$

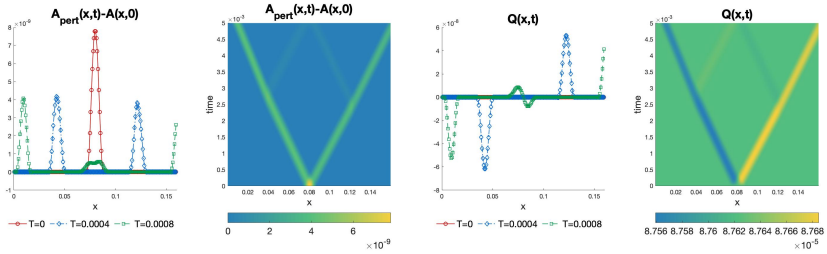
with $\epsilon = 5 \times 10^{-5}$. The wave splits in two and moves in opposite directions away from the initial perturbation. The test is run with 200 uniform cells until the stopping time $t = 0.005$ and the solutions are shown in Figure 3.18 for the living-man well-balanced scheme and in Figure 3.19 for the man-at-eternal-rest well-balanced scheme. It can be seen that only in the case with $S_{in} = 0.5$, the man-at-eternal-rest well-balanced scheme does not handle the perturbation very well. This is because, for small Shapiro number, the living-man equilibrium state is near to a man-at-eternal-rest equilibrium, so the error in using the man-at-eternal-rest well-balanced scheme is smaller than the error that arises from the perturbation.

Inflow Pulse to Q for an Artery with Stenosis & a Discontinuous Step

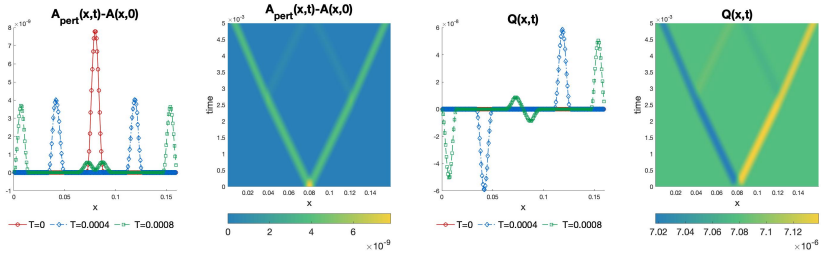
In this section we will simulate a pulse inflicted on the flow Q at the inlet of the domain. This pulse will be applied to both the stenosis and discontinuous step problems in Section 3.5.2. The initial conditions for both A and Q are determined in the same way as in Section 3.5.2, however we will introduce a different boundary condition for Q which simulates a pulse to the flow of blood. The boundary condition for Q at the inlet, denoted $\tilde{Q}_{in}$, is defined in the following way



(a)$S_{in} = 0.5$

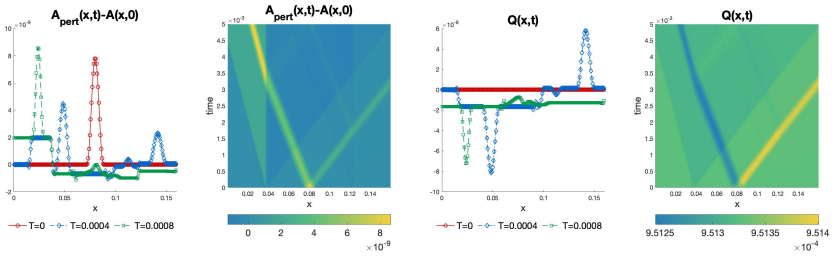


(b)$S_{in} = 0.1$

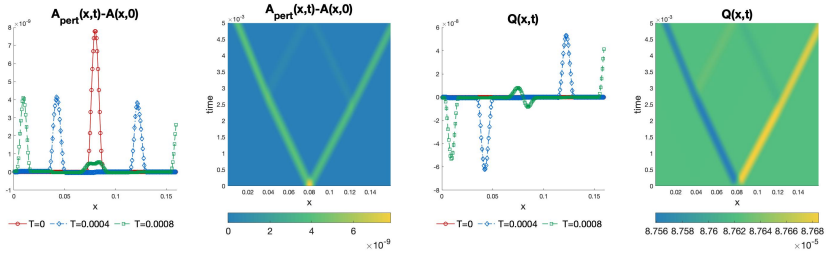


(c)$S_{in} = 0.01$

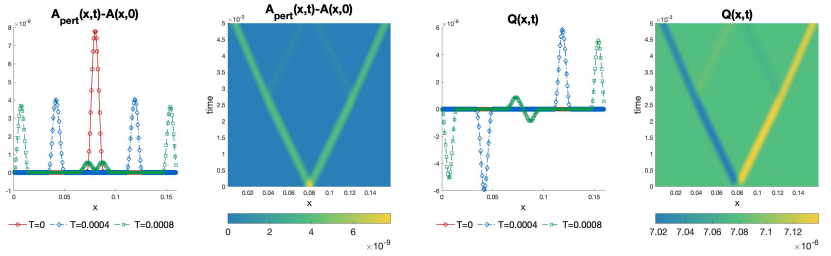
Figure 3.18: The perturbation to the *aneurysm* problem for different values of S_{in} when the *living-man well-balanced scheme* is used. The two types of plots include snapshots of the solution at times $t = 0, 0.0025, 0.005$, as well as plots that demonstrate how the perturbation propagates throughout the domain as a function of time. It can be seen that the larger the Shapiro number, the faster the perturbation propagates, especially the right moving wave. We can also see the formation of reflection waves.



(a)$S_{in} = 0.5$



(b)$S_{in} = 0.1$



(c)$S_{in} = 0.01$

Figure 3.19: The perturbation to the *aneurysm* problem for different values of S_{in} when the *man-at-eternal-rest well-balanced scheme* is used. The two types of plots include snapshots of the solution at times $t = 0, 0.0025, 0.005$, as well as plots that demonstrate how the perturbation propagates throughout the domain as a function of time. It can be seen that the smaller the Shapiro number, the better the scheme performs because the living-man steady state becomes nearer to a non-zero pressure man-at-eternal-rest steady state.

$$\tilde{Q}_{in}(t) = \begin{cases} Q_{in}\left(1 + \epsilon \sin\left(2\pi \frac{t}{T}\right)\right) & \text{if } t \leq \frac{T}{2}, \\ Q_{in} & \text{otherwise,} \end{cases} \tag{3.79}$$

where the pulse is inflicted until halfway through the computational time T, and then no more pulse is inflicted after that. This problem is similar to the one introduced in [60] with some modifications and the introduction of parameter ϵ. The value for Q_{in} is defined in (3.74) and depends on the Shapiro number $S_{in} = \{0.5, 0.1, 0.01\}$. We set the amplitude parameter $\epsilon = 1 \times 10^{-7}$ for the artery with stenosis and $\epsilon = 5 \times 10^{-2}$ for the decreasing step problem. The boundary condition is shown in Figure 3.20.

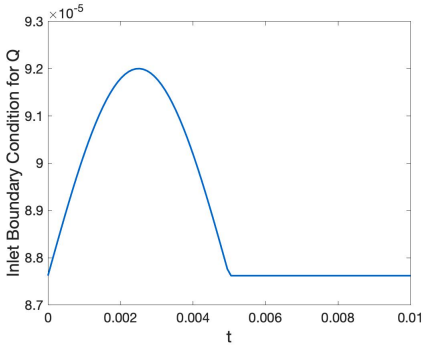


Figure 3.20: The boundary condition (3.79) with $S_{in} = 0.1, \epsilon = 5 \times 10^{-2}$, and $T = 0.01$.

All tests in this section are run until the final time of $t = 0.01$ with a mesh of 200 uniform cells. For each variation of the test, we will compare the performance of the living-man and man-at-eternal-rest well-balanced DG schemes. We present the results for an artery with stenosis in Figures 3.21 and 3.22. Similarly to the aneurysm perturbation problem, the man-at-eternal-rest scheme handles the perturbation as well as the living-man scheme for $S_{in} = 0.1$ and 0.01. When $S_{in} = 0.5$, obvious error is observed in the numerical

results of the man-at-eternal-rest well-balanced scheme. In Figures 3.23 and 3.24, we list the numerical results for the decreasing step problem, using both living-man and man-at-eternal-rest well-balanced DG schemes. Similar behavior can be observed.

3.5.4 Tests for Discontinuous Initial Conditions

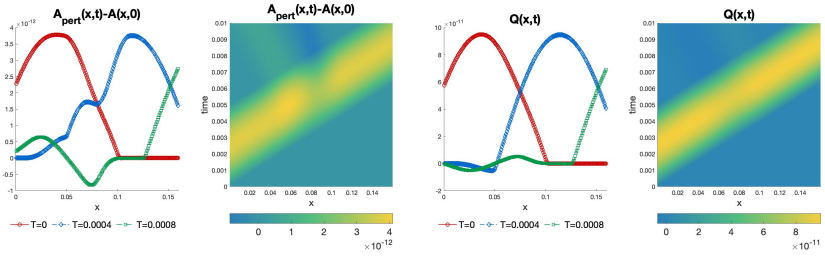
Two Riemann problems with discontinuous initial conditions will be considered in this section. We test the performance of the well-balanced DG methods in capturing discontinuous solutions.

The Ideal Tourniquet

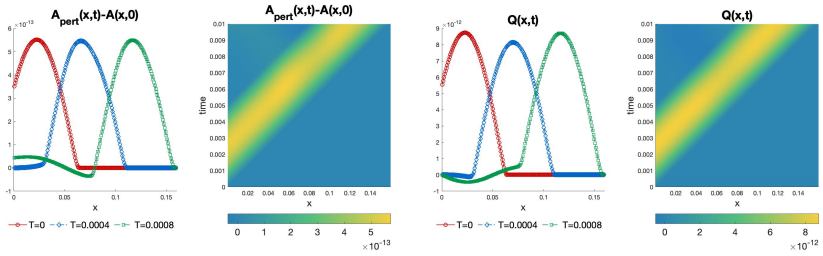
Dam break problems are frequently studied for the SWEs. For the blood flow problem, the analogue is the ideal tourniquet problem. We consider a tourniquet that is applied and instantaneously removed. The computational domain for this problem is $[-0.04, 0.04]$ and the initial conditions are given by

$$A(x,0) = \begin{cases} \pi(R_{in})^2, & \text{if } x \leq 0, \\ \pi(R_{out})^2, & \text{otherwise,} \end{cases} \qquad Q(x,0) = 0, \tag{3.80}$$

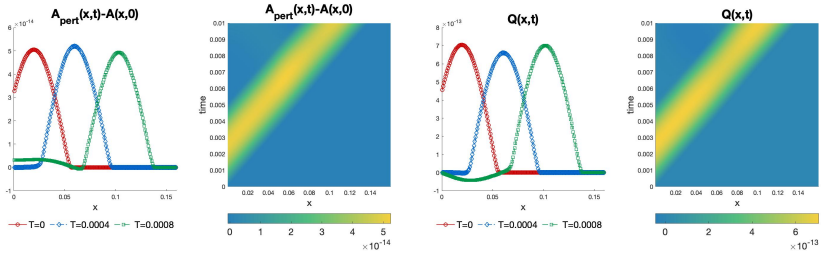
with the parameters listed in Table 3.14. The cross-sectional area at rest is defined as $A_0(x) = \pi(R_{out})^2$. Transmissive boundary conditions are implemented at the endpoints of the computational domain. The numerical solution is computed up to time $t = 0.005$. The discontinuity in the center becomes a shock wave propagating to the right and a rarefaction wave moving to the left. The numerical results with 200 uniform cells are presented in Figure



(a)$S_{in} = 0.5$

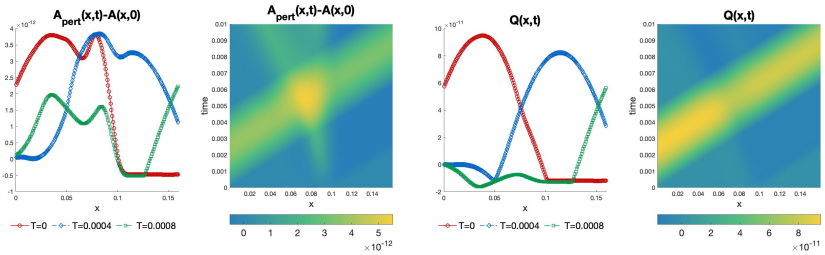


(b)$S_{in} = 0.1$

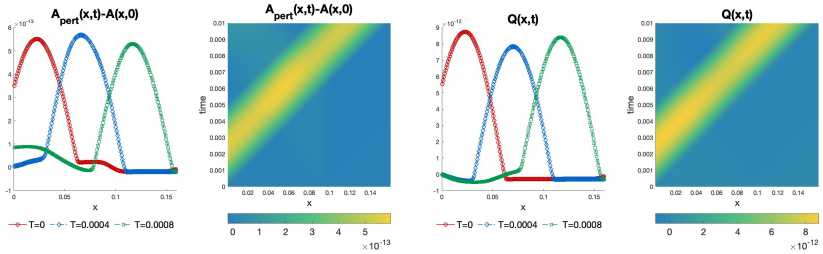


(c)$S_{in} = 0.01$

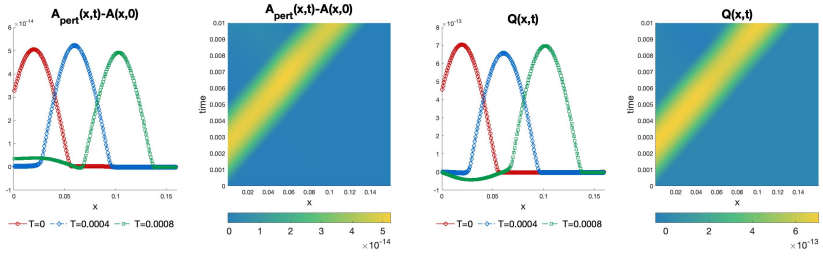
Figure 3.21: The perturbation to the *stenosis* problem for different values of S_{in} when the *living-man well-balanced scheme* is used. The two types of plots include snapshots of the solution at times $t = 0.004, 0.007, 0.01$, as well as plots that demonstrate how the perturbation propagates throughout the domain as a function of time. It can be seen that the larger the Shapiro number, the faster the perturbation propagates.



(a)$S_{in} = 0.5$

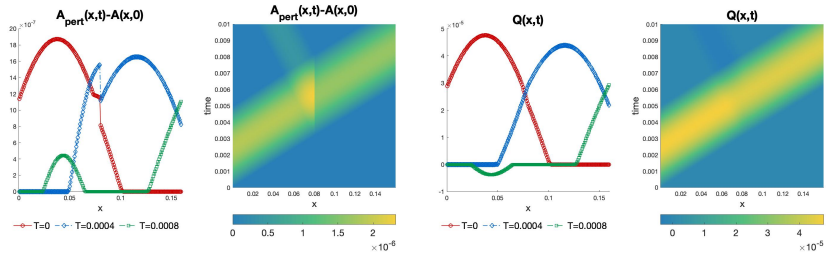


(b)$S_{in} = 0.1$

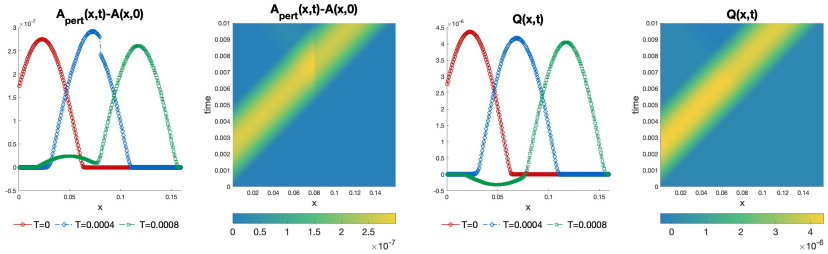


(c)$S_{in} = 0.01$

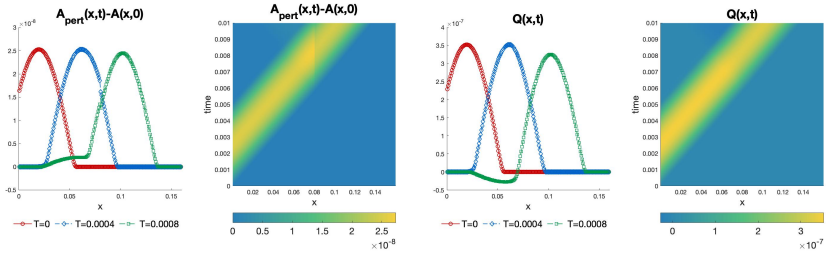
Figure 3.22: The perturbation to the *stenosis* problem for different values of S_{in} when the *non-zero pressure man-at-eternal-rest well-balanced scheme* is used. The two types of plots include snapshots of the solution at times $t = 0.004, 0.007, 0.01$, as well as plots that demonstrate how the perturbation propagates throughout the domain as a function of time.



(a)$S_{in} = 0.5$

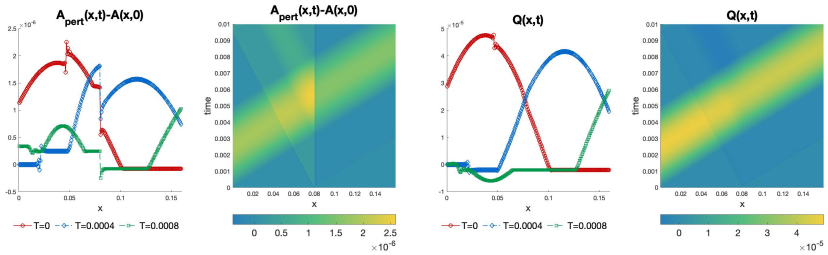


(b)$S_{in} = 0.1$

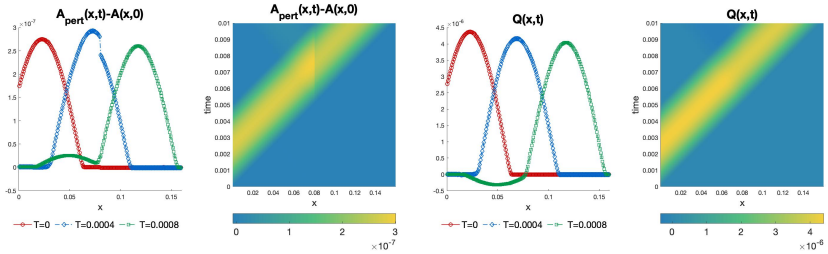


(c)$S_{in} = 0.01$

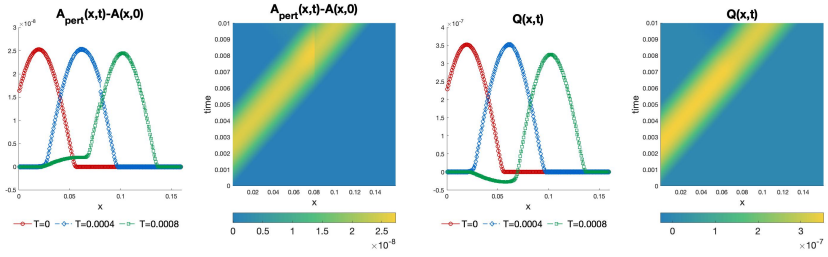
Figure 3.23: The perturbation to the *decreasing step* problem for different values of S_{in} when the *living-man well-balanced scheme* is used. The two types of plots include snapshots of the solution at times $t = 0.004, 0.007, 0.01$, as well as plots that demonstrate how the perturbation propagates throughout the domain as a function of time.



(a)$S_{in} = 0.5$



(b)$S_{in} = 0.1$



(c)$S_{in} = 0.01$

Figure 3.24: The perturbation to the *decreasing step* problem for different values of S_{in} when the *non-zero pressure man-at-eternal-rest well-balanced scheme* is used. The two types of plots include snapshots of the solution at times $t = 0.004, 0.007, 0.01$, as well as plots that demonstrate how the perturbation propagates throughout the domain as a function of time.

3.25. For comparison, we also present the simulation results with refined 1600 uniform cells as a "reference" solution. We can see that the numerical solution agree well with the refined solutions. Our well-balanced DG methods can capture the shock wave well, and the slope limiter removes oscillatory near the discontinuities.

R_{in}	R_{out}	K	ρ
$5 \times 10^{-3}\ m$	$4 \times 10^{-3}\ m$	$10^7\ {}^{Pa}/_{m}$	$1060\ {}^{kg}/_{m^3}$

Table 3.14: Parameters for the ideal tourniquet problem (3.80) and the Riemann problem (3.81).

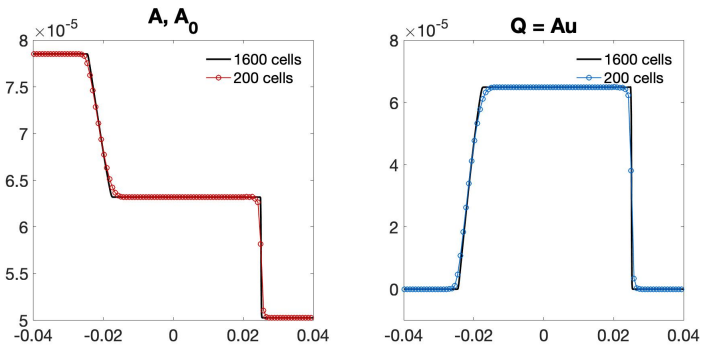


Figure 3.25: Numerical solutions at time $t = 0.005$ with quadratic basis function and mesh sizes of 200 and 1600 uniform cells for the ideal tourniquet problem (3.80).

Riemann Problem with Non-Flat Radius at Rest and Non-Zero Velocity

Next, we consider a problem similar to the ideal tourniquet problem, but with a non-zero velocity and discontinuous cross-sectional area at rest given by

$$A(x,0) = \begin{cases} \pi(4 \times 10^{-3})^2, & \text{if } x \leq 0, \\ \pi(3.5 \times 10^{-3})^2, & x > 0, \end{cases} \qquad Q(x,0) = \begin{cases} 1.5 \times 10^{-3}, & \text{if } x \leq 0, \\ 1 \times 10^{-3}, & \text{if } x > 0, \end{cases} \tag{3.81}$$

and

$$A_0(x) = \begin{cases} \pi(2.5 \times 10^{-3})^2, & \text{if } x \leq 0, \\ \\ \pi(3 \times 10^{-3})^2, & x > 0, \end{cases}, \tag{3.82}$$

The computational domain for this problem is $[-0.04, 0.04]$ and the test is run until time $t = 0.008$. The numerical results with 200 uniform cells and TVB minmod limiter are presented in Figure 3.26, and compared with the "reference" solution obtained with refined 1600 uniform cells. We can see that the numerical solution agrees well with the refined solutions. The minmod limiter marks the troubled cells based on $\boldsymbol{U}_\tau^f$ and performs the actual limiting procedure on $\boldsymbol{U}_\tau$. We also plotted the figures of the decomposed solutions $\boldsymbol{U}_\tau^f$ and $\boldsymbol{U}_\tau^e$ at the final time in Figure 3.27 in which the troubled cells marked by the limiter are distinguished from the non-limited cells. We have compared minmod slope procedure with the results of the standard minmod limiter (both trouble cell indicator and limiting on $\boldsymbol{U}_\tau$), and observed the same results. In these figures, it can be seen that the shock profiles are smeared with 200 cells. For comparison, we also included the numerical results in Figure 3.28 when WENO limiter is used instead, and this gives a sharp shock profile on the same mesh (200 cells).

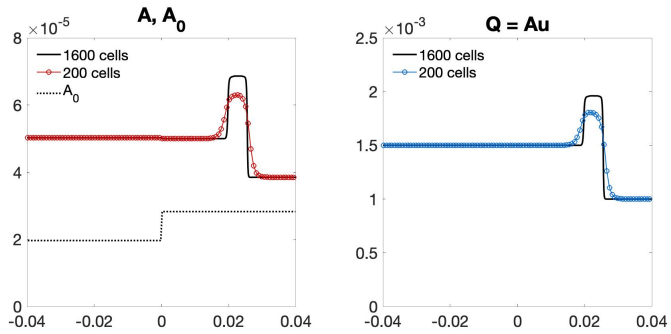


Figure 3.26: Numerical solutions at time $t = 0.008$ with quadratic basis function and mesh sizes of 200 and 1600 uniform cells for the Riemann problem (3.81). TVB minmod limiter is used.

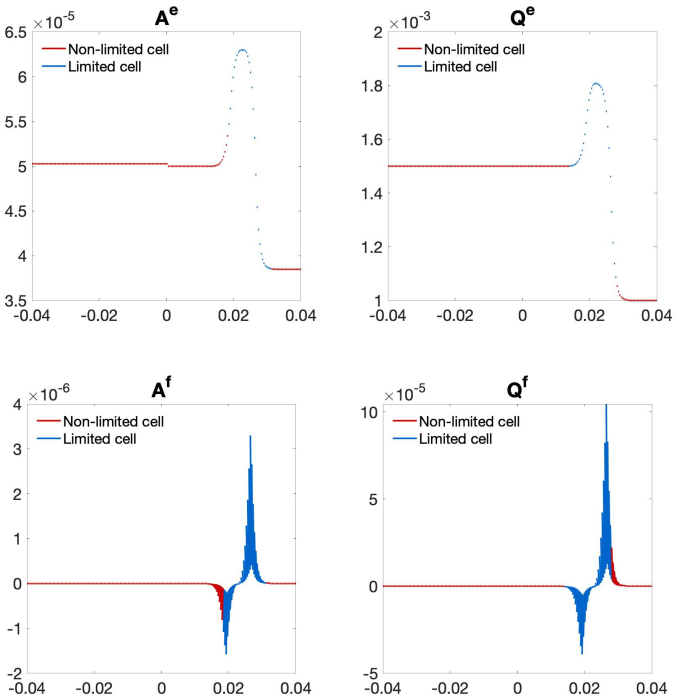


Figure 3.27: Numerical solutions of $\boldsymbol{U}_\tau^e$ (top row) and $\boldsymbol{U}_\tau^f$ (bottom row) at time $t = 0.008$ with quadratic basis function and mesh size of 200 uniform cells for the Riemann problem (3.81). Cells in which the minmod limiter was applied after the last RK step are colored in blue and non-limited cells are colored red.

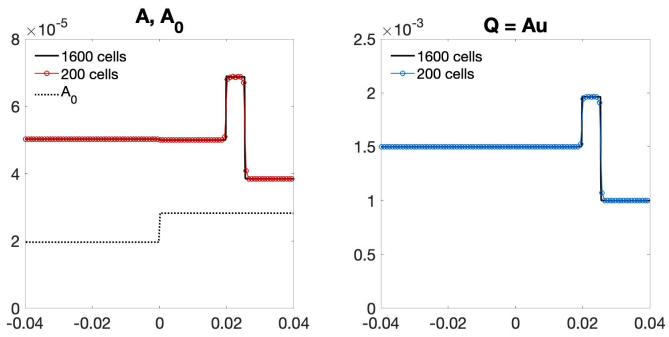


Figure 3.28: Numerical solutions at time $t = 0.008$ with quadratic basis function and mesh sizes of 200 and 1600 uniform cells for the Riemann problem (3.81). The WENO limiter is used.

Chapter 4

Recovery of a Time-Dependent Bottom Topography Function from the Shallow Water Equations via an Adjoint Approach

4.1 Introduction

Various phenomena arising frequently in natural, engineering and socio-economical applications can be modeled by hyperbolic conservation and balance laws. Examples of conservation laws include models for traffic flow [58], fluid dynamics [46] and supply chains [18]. Moreover, conservation laws with source terms, also known as balance laws, are used in different models, e.g. the gas pipeline flow [68], shallow water flow [151, 85], gas dynamics

under gravitational field [157], and blood flow through arteries [163]. A variety of theoretical studies have been conducted to understand the underlying mathematical structure and a wide range of numerical approaches dedicated to solving for the state variables have been developed in the past few decades. On the other hand, optimization, control, and recovery of the system parameters is a problem of great interest due to its high practical value. In this work, we derive an inverse problem algorithm for a specific system of hyperbolic balance laws, in which a time-dependent topographical profile is recovered based on the measurement from the boundary.

Various optimization and control methods have been developed for conservation and balance laws such as backstepping [144, 62], Lyapunov-based [144], derivative-free [88], and optimal control methods [79]. In this work we employ an adjoint approach, which is often praised for its efficiency. Its computational cost (of each iteration) is comparable to that of solving a partial differential equation (PDE) once, instead of depending on the number of control variables or design parameters as in other optimization methods.

Adjoint methods were introduced for optimal control problems in 1971 by Lions [99] in the context of shape optimization of aerodynamic bodies. A few years later the methods were extended to fluid dynamics by Pironneau [117]. Jameson popularized the techniques for potential flow and the Euler equations [80, 81]. The methods have also been applied to biological systems in the search of parameter identification [123]. Estimation of open water states [24] and traffic states on the freeways [78] have also been accomplished via the adjoint optimization method. With regards to scalar conservation laws, Holden et al. [75] developed a reconstruction procedure for the coefficient inverse problem in which a

spatially dependent coefficient of the flux term is recovered. Bürger et al. [21] solved the inverse scalar conservation law modelling sedimentation numerically by assuming a variational form of the problem. The flux function of a scalar conservation law was reconstructed using the information from the shock that forms in the work by Kang and Tanuma [83]. In a more general setting for balance laws, Montecinos et al. [103] derived a unified scheme for solving the forward and adjoint problems simultaneously. Methodology for the scalar Burger's equation was presented by Lellouche et al. [89] in which the authors aimed to find the best approximation for the measured data by means of boundary control and an adjoint approach. Ferlauto [49] obtained optimal geometric shapes for aerodynamic bodies by solving an inverse problem for the three-dimensional incompressible Euler equations.

Numerical computation of the optimization problems for conservation laws have been studied extensively due to the theoretical and numerical challenges that arise. As the exact solution of conservation laws often contains discontinuities, one challenge in the related optimization problems is that non-negligible numerical errors may occur in capturing the discontinuities. Some of these difficulties are mitigated via introduction of the Lax-Friedrichs schemes [61] or relaxation methods [2], for instance. Convergence analyses have been provided for optimization problems in the aforementioned works. The fact that many conservation laws are nonlinear presents another challenge because this can lead to non-convex formulations of the optimization problem. One method to tackle this difficulty is to use linear programming methods once the discretization scheme is 'relaxed' [64, 166]. This allows for a global optimum to be found and reduces computational cost, but the linearization may not be a good physical representation of the original system [125]. To

maintain the nonlinearity of the system, a more expensive method, such as gradient descent, can be used but may not ensure a global optimum is achieved.

In this chapter, we focus on the inverse problem that arises in first order nonlinear hyperbolic balance laws. Many difficulties arise in this research field as a result of uncertainties in data, measurements, and the use of complex data. It is very important to develop highly accurate, easy to implement, and cost-efficient methods with high resolution to study fluvial environments numerically. Although the method we employ can be quite general, for the sake of simplicity and better illustrative purposes, we focus on a prototype example of hyperbolic balance laws in this work, namely the nonlinear SWEs, also referred to as the Saint-Vernant equations. This system models fluvial environments such as flood plane dynamics, coastal and tidal flows, and flow and sediment transport. It has wide applications in ocean, environmental, hydraulic engineering and atmospheric modeling. The model parameter we aim to recover in this context is the riverbed topography, however other terms such as friction may also be of interest to reconstruct.

Ground surveys of riverbeds or direct topographical data collection are not always effective for determining underwater topography because the operations are costly and time consuming. Numerical techniques of constructing riverbed topography can offer many benefits over aerial and ground techniques in terms of cost reduction, efficiency, as well as flexibility. Heining and Aksel [70] used a direct approach to reconstruct the bottom topography of steady-state thin-film flow. Castaings et al. [24] presented an automatic differentiation technique and free surface information to reconstruct river bed topology. Honnorat et al. [76] derived a method for recovering channel topology from a steady-state

solution of the forward problem using an optimization technique called variational data assimilation. A direct approach from the one-dimensional SWEs was used by Gessese et al. [59] to reconstruct the river bed from free surface data. Lastly, a stable finite volume scheme in the presence of wetting-drying fronts and inverse computational algorithms (based on variational approach) were presented in [102]. All methods developed in these studies dealt with time-independent bottom topography functions.

The main objective of this chapter is to develop a robust algorithm that requires less data to reconstruct a dynamic bottom function, thereby allowing the construction of a more accurate and inexpensive model. We assume the measurements, possibly with noise, are taken only on two boundaries of the spatial domain in a given time period. Traditionally, the bottom topography in the SWEs is a function of space alone within the framework of the inverse problem construction. Here, we consider the time-dependent bottom topography function, which allows for the recovery of the bottom topography with less data considering the fact that movement in the forward problem solution coming from the change of the bottom topography allows more information propagation to the boundary measurements. Usually inverse problems are more difficult when we need to recover both the temporal and spatial profile. In this work, as a first step, we assume a special form of the topographical profile which represents two known spatial profiles and an unknown temporal interaction. This time-dependent bottom topography can practically describe a physical phenomenon when two platonic plates with known topography are moving against one another, e.g. an earthquake, underwater volcanic buildup, or a moving sand bottom. We aim to recover only the temporal profile representing, e.g. the pulse of the earthquake. After constructing the

adjoint formulation, we present the cost function with two regularization terms added to suppress noise and to handle the ill-posedness of this problem. An iterative update scheme based on a three-operator splitting scheme is employed to update the targeting function. This splitting scheme requires each operator to be computed only once per iteration and is straightforward to implement.

This chapter is organized as follows. In Section 4.2 we introduce the primal equations used throughout this work. The DG numerical scheme is also presented as the method used for solving the forward problem. Section 4.3 includes a discussion on the formulation of the inverse problem. The adjoint equations are derived from a linearized system and used to derive the gradient formulation of the cost function. This section also contains the description of the iterative updating procedure for determining the desired source term, as well as a discussion on the choice of regularization terms. Numerical examples are presented in Section 4.4.

4.2 Forward Problem

In this section, we present the hyperbolic PDE system used to define the forward (or primal) problem. The forward system will be used in Section 4.3 to determine the adjoint formulation, which in turn is employed to derive the gradient of a cost function and iteratively update the time-dependent bottom topography function. We will discuss the PDE system as well as the numerical scheme used to solve the forward problem.

4.2.1 Forward Problem Formulation

Hyperbolic balance laws are conservation laws with a source term. A few examples of such systems include the nonlinear SWEs, the arterial blood flow model, the Euler equations under gravity, and the telegrapher's equations. A source term might arise as a result of many factors, such as a friction term or a topographical term.

In this work, we only consider the one-dimensional systems of m hyperbolic balance laws which take the form

$$\begin{cases} \partial_t \boldsymbol{U} + \partial_x \boldsymbol{F}(\boldsymbol{U}) = \hat{\boldsymbol{S}}(\boldsymbol{U}, \boldsymbol{B}), & (x,t) \in (x_0, x_L) \times (0,T], \\ \boldsymbol{U}(x,0) = \boldsymbol{U}_0(x), & x \in (x_0, x_L), \end{cases} \tag{4.1}$$

where $\boldsymbol{U}$ are the state variables, $\boldsymbol{F}(\boldsymbol{U})$ are the fluxes, and $\boldsymbol{U}_0(x)$ are the initial conditions. The vector $\boldsymbol{B}$ represents the model parameters we aim to recover in the inverse problem and is only present in the source term, $\hat{\boldsymbol{S}}(\boldsymbol{U}, \boldsymbol{B})$. The source term can be rewritten in the form of $\hat{\boldsymbol{S}}(\boldsymbol{U}, \boldsymbol{B}) = \boldsymbol{S}(\boldsymbol{U}, \boldsymbol{B})\boldsymbol{U}$, in which $\boldsymbol{S}(\boldsymbol{U}, \boldsymbol{B}) \in \mathbb{R}^m \times \mathbb{R}^m$ is a matrix.

Alternatively, the system can be written in quasi-linear form using the Jacobian matrix $\boldsymbol{A}(\boldsymbol{U}) = \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{U}}$,

$$\begin{cases} \partial_t \boldsymbol{U} + \boldsymbol{A}(\boldsymbol{U}) \partial_x \boldsymbol{U} = \boldsymbol{S}(\boldsymbol{U}, \boldsymbol{B})\boldsymbol{U}, & (x,t) \in (x_0, x_L) \times (0,T], \\ \boldsymbol{U}(x,0) = \boldsymbol{U}_0(x), & x \in (x_0, x_L). \end{cases} \tag{4.2}$$

To determine the solution of a forward problem, one seeks to determine the state variables $\boldsymbol{U}$ with the model parameters $\boldsymbol{B}$ given. In this work, we will only consider the case when we have a single topographical function, denoted by B. While the information in B is traditionally a function of space alone, here we consider B as a function depending on both

space and time as follows:

$$B(x,t) = B_0(x) + p(t)B_1(x), \tag{4.3}$$

where we assume $B_0(x)$ and $B_1(x)$ to be known and $p(t)$ to be the component we wish to recover.

In this chapter we will focus on the SWEs with a non-flat bottom topography, one of the most well known systems of hyperbolic balance laws. In particular, we aim to recover the riverbed topography function, denoted by b. This term occurs only in the source term of the momentum equation in the form of its derivative, $\partial_x b$, so we define

$$B = \partial_x b(x,t) = \partial_x b_0(x) + p(t)\partial_x b_1(x). \tag{4.4}$$

The state variables, flux terms, and source term for the SWEs are given by

$$\boldsymbol{U} = \begin{bmatrix} h \\ hu \end{bmatrix}, \quad \boldsymbol{F}(\boldsymbol{U}) = \begin{bmatrix} hu \\ hu^2 + \frac{1}{2}gh^2 \end{bmatrix}, \quad \hat{\boldsymbol{S}}(\boldsymbol{U}, B) = \begin{bmatrix} 0 \\ -ghB \end{bmatrix} = \begin{bmatrix} 0 \\ -gh\partial_x b \end{bmatrix}, \tag{4.5}$$

by following the form (4.1). Here, $h(x,t) \geq 0$ is the water height, $(hu)(x,t)$ is the water discharge with $u(x,t)$ being the depth averaged velocity, and $g = 9.812$ is the gravitational constant. On the other hand, under the quasi-linear formulation, we write

$$\boldsymbol{A}(\boldsymbol{U}) = \begin{bmatrix} 0 & 1 \\ gh - u^2 & 2u \end{bmatrix}, \qquad \boldsymbol{S}(\boldsymbol{U}, B) = \begin{bmatrix} 0 & 0 \\ -g\partial_x b & 0 \end{bmatrix}. \tag{4.6}$$

Time-dependent bottom topography functions have been considered in the literature. In a more complicated model, e.g. [139, 72, 77, 97, 85], the change of the bottom function in the SWEs may depend on other state variables. For instance, the bottom function may be determined by erosion, sediment transport, dam breaks, or landslides due to

floods. In such cases, additional equations to model the evolution of bottom topography may arise in the system in order to better describe these dependence. However, these models are different from the one we consider in this work.

4.2.2 Discontinuous Galerkin Method for the Forward Problem

The DG method will be used to solve the forward problem (4.6). It is a high order accurate scheme that has gained significant attention in the last decade. The method is advantageous for hyperbolic conservation laws because it is both stable, similar to the finite volume method, and flexible, like the finite element method. The arbitrary order feature of the DG method can provide accurate results on a coarse mesh. In particular, in an inverse problem algorithm, a forward solver is usually employed during each iteration, hence the use of a coarser mesh is ideal in an effort to reduce computation cost in the iterative process. Lastly, the DG scheme is able to capture the discontinuous solutions well and help us locate the interfaces accurately.

We seek an approximation $\boldsymbol{U}_\tau$ of the solution $\boldsymbol{U}$, in which $\boldsymbol{U}_\tau^{(i)}$ for $i = 1, ..., m$ belongs to the finite dimensional piecewise polynomial space $\mathbb{V}_\tau^k$. The DG scheme takes the form (1.17) where we use the Lax-Friedrichs flux (1.18). In the case of the SWEs, let $\alpha = \max(|u| + \sqrt{gh})$. The scheme is advanced in time using the SSP-RK3 method (1.19). The full details of the method are presented in Section 1.3.2.

4.3 Inverse Problem

In this section, we describe the inverse problem of our focus, which is the reconstruction of the topographical source function B from boundary data of the hyperbolic conservation law (4.1) from a single measurement event. We reduce the inverse problem to an optimization problem of a residual functional coming from boundary measurements, with an addition of two regularization terms, which will be described more concretely later in the section.

In our work, we adopt the adjoint method to numerically obtain a gradient of our functional. We will describe the cost function we wish to minimize, the derivation of the adjoint formulation for the gradient calculation, and the iterative update scheme for determining the time component, $p(t)$, of the source function B. The numerical solution to the adjoint problem will be calculated using the DG method.

We assume that noisy measurements of $\boldsymbol{U}$ are taken only on both boundaries of the spatial domain, given a period of time $[0, T]$ in one single measurement event. During the numerical reconstruction process, we assume that only these noisy solutions at the boundary of the spatial domain is known to us. For notational sake, we denote these noisy measurements as $\hat{\Lambda}^{\text{noisy}} = \hat{\Lambda}\mu$ where the multiplicative noise is uniformly distributed, $\mu \sim \mathcal{U}\left[1 - \frac{1}{2}\eta_{meas}, 1 + \frac{1}{2}\eta_{meas}\right]$, with a given noise level η_{meas}. The goal of the inverse scheme is to find the function, $p(t)$, that provides the best approximation $\Lambda(B(p)) \approx \hat{\Lambda}^{\text{noisy}}$. The map $\Lambda(B(p)) = \boldsymbol{U}|_{\{x_0, x_L\}\times[0,T]}$ represents the forward map with the input, $B(p)$, as the topographical function and the output as the solutions, $\boldsymbol{U}$, at the boundary points, $\{x_0, x_L\}$, over the time interval $[0, T]$. Finding the best approximation reduces to minimizing the

error or residue of the predicted and measured data for all time at the boundary of the computational domain. This corresponds to minimizing the functional

$$\boldsymbol{J}(p) := \boldsymbol{J}_0(p) + \boldsymbol{R}(p) := \int_0^T \frac{1}{2} \left|\left[\mathcal{E}(p)\right](x_0, t)\right|^2 + \frac{1}{2} \left|\left[\mathcal{E}(p)\right](x_L, t)\right|^2 dt + \boldsymbol{R}(p), \tag{4.7}$$

where the error function, $\mathcal{E}$, for a given p is defined as

$$\left[\mathcal{E}(p)\right](x, t) = \Lambda(B(p))(x, t) - \hat{\Lambda}^{\text{noisy}}(x, t). \tag{4.8}$$

The term $\boldsymbol{R}(p)$ is a regularization term that will be discussed in Section 4.3.3.

The optimization problem becomes

$$\text{minimize } \boldsymbol{J}(p) := \boldsymbol{J}_0(p) + \boldsymbol{R}(p) \quad \text{subject to (4.2).} \tag{4.9}$$

We employ a descent method to minimize the above functional, in which the (formal) gradient $\nabla \boldsymbol{J}$ will be obtained via the adjoint method following a linearization process of the equation (4.2).

4.3.1 Gradient Derivation

In an effort to determine $\nabla \boldsymbol{J}_0$, we begin by calculating the variational derivative (in the sense of the Gateaux differential) of $\boldsymbol{J}_0$ and dualize it using L^2-pivoting. In what follows, we would like to denote, for a functional $\mathcal{F}$, the variational derivative of $\mathcal{F}$ at p along $\tilde{p}$ as

$$\delta\mathcal{F}(p; \tilde{p}) := \lim_{\epsilon \to 0} \frac{\mathcal{F}(p + \epsilon\tilde{p}) - \mathcal{F}(p)}{\epsilon} \tag{4.10}$$

whenever it exists. Furthermore, whenever $\delta\mathcal{F}(p; \tilde{p})$ is linear with respect to $\tilde{p}$, we (formally) dualize the variational derivative $\delta\mathcal{F}(p; \tilde{p})$ using L^2-pivoting and define the gradient, $\nabla\mathcal{F}(p)$,

such that it satisfies the relation

$$\delta\mathcal{F}(p;\tilde{p}) := \int_0^T \left[\nabla\mathcal{F}(p)\right](t)\,\tilde{p}(t)\;dt. \tag{4.11}$$

With these notions at hand, we readily compute that

$$\begin{aligned}\boldsymbol{\delta J}_0(p;\tilde{p}) &= \lim_{\epsilon\to 0}\frac{\boldsymbol{J}_0(p+\epsilon\tilde{p})-\boldsymbol{J}_0(p)}{\epsilon}\\ &= \int_0^T \left(\left[\delta\mathcal{E}^T(p;\tilde{p})\,\mathcal{E}(p)\right](x_0,t) + \left[\delta\mathcal{E}^T(p;\tilde{p})\,\mathcal{E}(p)\,\right](x_L,t)\right)\,dt\,,\end{aligned} \tag{4.12}$$

where the superscript ^{T} now represents the transpose of a matrix (and not the adjoint operator). From the definition of $\mathcal{E}$, we quickly realize that $\delta\mathcal{E}(p;\tilde{p})(x,t) = \delta\Lambda(B(p);B(\tilde{p}))$. Hence, (4.12) reduces to

$$\boldsymbol{\delta J}_0(p;\tilde{p}) = \int_0^T \left(\left[\delta\Lambda^T(B(p);B(\tilde{p}))\,\mathcal{E}(p)\,\right](x_0,t) + \left[\delta\Lambda^T(B(p);B(\tilde{p}))\,\mathcal{E}(p)\right](x_L,t)\right)\;dt\,. \tag{4.13}$$

We now see the necessity of evaluating the term $\delta\Lambda(B(p);B(\tilde{p}))$ explicitly. Albeit seemingly complicated, the difficulty of the evaluation will be mitigated via solving a related adjoint equation, which will be described in the next subsection.

4.3.2 Linearization & Adjoint Formulation

The adjoint formulation can be understood from multiple perspectives. One way is the Lagrange framework in which the adjoint variables are Lagrange multipliers. This method is commonly used in the aeronautical community, popularized by Jameson [80], because it provided a solid connection to theories of constrained optimal control and optimization. Another type of approach, the duality framework, requires one to linearize the system in order to derive the adjoint equations. We will use the duality framework in this work, however the Lagrange framework provides the exact same adjoint formulation.

Linearization of the forward system

In this subsection, we aim to linearize the forward system (4.2) as follows. We consider an ϵ-perturbation of B, $B^\varepsilon := B + \epsilon\tilde{B}$ along the direction $\tilde{B}$, and see how the resulting $\boldsymbol{U}$ that satisfies (4.2) is perturbed. We denote $\boldsymbol{U}^\epsilon$ as the solution to (4.2) given B^ε and define

$$\tilde{\boldsymbol{U}} := \lim_{\epsilon\to 0} \frac{\boldsymbol{U}^\epsilon - \boldsymbol{U}}{\epsilon}$$

whenever it exists. Now we quickly realize that

$$\tilde{\boldsymbol{U}}(x,0) = \lim_{\epsilon\to 0} \frac{\boldsymbol{U}^\epsilon(x,0) - \boldsymbol{U}(x,0)}{\epsilon} = 0 \tag{4.14}$$

as the initial conditions of $\boldsymbol{U}^\epsilon$ and $\boldsymbol{U}$ shall coincide. Moreover, taking the differences of the respective equations coming from (4.2) for $\boldsymbol{U}^\epsilon$ and $\boldsymbol{U}$ directly gives

$$0 = \lim_{\epsilon\to 0} \frac{1}{\epsilon} \left(\partial_t[\boldsymbol{U}^\epsilon - \boldsymbol{U}] + [\boldsymbol{A}(\boldsymbol{U}^\epsilon)\partial_x\boldsymbol{U}^\epsilon - \boldsymbol{A}(\boldsymbol{U})\partial_x\boldsymbol{U}] - [\boldsymbol{S}(\boldsymbol{U}^\epsilon, B^\epsilon)\boldsymbol{U}^\epsilon - \boldsymbol{S}(\boldsymbol{U}, B)\boldsymbol{U}]\right), \tag{4.15}$$

and each term in the bracket can be simplified whenever they exist. For instance, we directly have

$$\lim_{\epsilon\to 0} \frac{1}{\epsilon}\partial_t \left(\boldsymbol{U}^\epsilon - \boldsymbol{U}\right) \quad = \quad \partial_t\tilde{\boldsymbol{U}}. \tag{4.16}$$

Meanwhile, via product rule, together with the fact that the symmetry relationship $\partial_{U_k} \boldsymbol{A}_{ij}(U) = \partial_{U_k}\partial_{U_j} \boldsymbol{F}_i(U) = \partial_{U_j} \boldsymbol{A}_{ik}(U)$ holds, we may simplify the flux term as

$$\begin{aligned}
&\lim_{\epsilon\to 0} \frac{1}{\epsilon} \left(\boldsymbol{A}(\boldsymbol{U}^\epsilon)\partial_x \boldsymbol{U}^\epsilon - \boldsymbol{A}(\boldsymbol{U})\partial_x \boldsymbol{U}\right) \\
&\qquad = \boldsymbol{A}(\boldsymbol{U})\partial_x \tilde{\boldsymbol{U}} + \left(\sum_{k=1}^{m} \partial_{\boldsymbol{U}_k} A(\boldsymbol{U})\, \tilde{\boldsymbol{U}}_k\right) \partial_x \boldsymbol{U} \\
&\qquad = \boldsymbol{A}(\boldsymbol{U})\partial_x \tilde{\boldsymbol{U}} + \left(\sum_{j=1}^{m} \partial_{\boldsymbol{U}_j} A(\boldsymbol{U})\partial_x \boldsymbol{U}\right) \tilde{\boldsymbol{U}} \\
&\qquad = \boldsymbol{A}(\boldsymbol{U}) \left[\partial_x \tilde{\boldsymbol{U}}\right] + \left[\partial_x \boldsymbol{A}(\boldsymbol{U})\right] \tilde{\boldsymbol{U}} \\
&\qquad = \partial_x \left[\boldsymbol{A}(\boldsymbol{U})\tilde{\boldsymbol{U}}\right].
\end{aligned} \tag{4.17}$$

Likewise, we can simplify the source term and obtain

$$\begin{aligned}
&\lim_{\epsilon\to 0} \frac{1}{\epsilon} \left(\boldsymbol{S}(\boldsymbol{U}^\epsilon, B^\epsilon)\boldsymbol{U}^\epsilon - \boldsymbol{S}(\boldsymbol{U}, B)\boldsymbol{U}\right) \\
&\qquad = \boldsymbol{S}(\boldsymbol{U}, B)\tilde{\boldsymbol{U}} + \lim_{\epsilon\to 0} \frac{1}{\epsilon} \left(\boldsymbol{S}(\boldsymbol{U} + \epsilon\tilde{\boldsymbol{U}}, B + \epsilon\tilde{B}) - \boldsymbol{S}(\boldsymbol{U}, B)\right) \boldsymbol{U}^\epsilon \\
&\qquad = \boldsymbol{S}(\boldsymbol{U}, B)\tilde{\boldsymbol{U}} + \left(\sum_{i=1}^{m} \partial_{\boldsymbol{U}_i} \boldsymbol{S}(\boldsymbol{U}, B)\tilde{\boldsymbol{U}}_i + \partial_B \boldsymbol{S}(\boldsymbol{U}, B)\tilde{B}\right) \boldsymbol{U} \\
&\qquad := \left(\boldsymbol{S}(\boldsymbol{U}, B) + \boldsymbol{C}(\boldsymbol{U}, B)\right) \tilde{\boldsymbol{U}} + \partial_B \boldsymbol{S}(\boldsymbol{U}, B)\boldsymbol{U}\tilde{B},
\end{aligned} \tag{4.18}$$

where $\boldsymbol{C}$ denotes the matrix $\boldsymbol{C}_{ij} = \sum_{k=1}^{m} \frac{\partial \boldsymbol{S}_{ik}}{\partial \boldsymbol{U}_j} \boldsymbol{U}_k$. Substituting (4.16), (4.17) and (4.18) into (4.15), and combining that with the initial condition (4.14), we therefore obtain the following linear system for $\tilde{\boldsymbol{U}}$

$$\begin{cases}
\left(\partial_t - \boldsymbol{S} - \boldsymbol{C}\right) \tilde{\boldsymbol{U}} + \partial_x \left(\boldsymbol{A}\tilde{\boldsymbol{U}}\right) = \left(\partial_B \boldsymbol{S}(\boldsymbol{U}, B)\boldsymbol{U}\right) \tilde{B}, & (x,t) \in (x_0, x_L) \times (0, T], \\
\tilde{\boldsymbol{U}}(x, 0) = 0, & x \in [x_0, x_L],
\end{cases} \tag{4.19}$$

which serves as the linearization of the forward system (4.2).

The adjoint system

With the linearization process given in the previous subsection, we may proceed to obtain $\delta\Lambda(B(p);B(\tilde{p}))$ at the boundary points, and thereby evaluate $\boldsymbol{\delta J}_0(p;\tilde{p})$ appropriately.

We start by considering $\boldsymbol{\sigma}$ which satisfies the following adjoint system with final time condition and boundary conditions

$$\begin{cases} (\partial_t + \boldsymbol{A}^T\partial_x + \boldsymbol{S}^T + \boldsymbol{C}^T)\boldsymbol{\sigma} = 0, & x \in (x_0, x_L)\times(0,T], \\ \boldsymbol{\sigma}(x,T) = 0, & x \in (x_0, x_L), \\ \boldsymbol{\sigma}(x_0,t) = -(\boldsymbol{A}^T)^{-1}(x_0,t)\left[\mathcal{E}(p)\right](x_0,t), & t \in (0,T], \\ \boldsymbol{\sigma}(x_L,t) = (\boldsymbol{A}^T)^{-1}(x_L,t)\left[\mathcal{E}(p)\right](x_L,t) & t \in (0,T]\,. \end{cases} \tag{4.20}$$

In the particular case of the SWEs, the matrices appearing in (4.20) are given by

$$\boldsymbol{A}^T = \begin{bmatrix} 0 & gh-u^2 \\ 1 & 2u \end{bmatrix}, \qquad \boldsymbol{S}^T = \begin{bmatrix} 0 & -g\partial_x b \\ 0 & 0 \end{bmatrix}, \qquad \boldsymbol{C}^T = 0. \tag{4.21}$$

Taking the inner product of the solution $\boldsymbol{\sigma}$ of (4.20) and the weak formulation of the linearized system in (4.19), we get

$$\begin{aligned} &\int_0^T \int_{x_0}^{x_L} \boldsymbol{\sigma}^T \left(\partial_B \boldsymbol{S}(\boldsymbol{U},B)\boldsymbol{U}\right)\tilde{B}\; dxdt \\ &\quad = \int_0^T \int_{x_0}^{x_L} \boldsymbol{\sigma}^T \left(\partial_t + \partial_x \boldsymbol{A} - \boldsymbol{S} - \boldsymbol{C}\right)\tilde{\boldsymbol{U}}\; dxdt \\ &\quad = -\int_0^T \int_{x_0}^{x_L} \tilde{\boldsymbol{U}}^T \left(\partial_t + \boldsymbol{A}^T\partial_x + \boldsymbol{S}^T + \boldsymbol{C}^T\right)\boldsymbol{\sigma}\; dxdt \\ &\qquad + \int_{x_0}^{x_L} \tilde{\boldsymbol{U}}^T\boldsymbol{\sigma}\Big|_{t=0}^{t=T} dx + \int_0^T \tilde{\boldsymbol{U}}^T\boldsymbol{A}^T\boldsymbol{\sigma}\Big|_{x=x_0}^{x=x_L} dt, \end{aligned} \tag{4.22}$$

where we simplify further, with (4.20), to obtain

$$\begin{aligned}
&\int_0^T \int_{x_0}^{x_L} \boldsymbol{\sigma}^T(x,t)\left(\partial_B \boldsymbol{S}(\boldsymbol{U},B)\boldsymbol{U}\right)(x,t)\tilde{B}(x,t)\; dxdt \\
&= \int_0^T \left[\delta\Lambda^T(B(p);\tilde{B}(p))\,\mathcal{E}(p)\right](x_L,t)\; dt \\
&\qquad + \int_0^T \left[\delta\Lambda^T(B(p);\tilde{B}(p))\,\mathcal{E}(p)\right](x_0,t)\; dt.
\end{aligned} \tag{4.23}$$

Here the last equality follows from the choice of boundary conditions described in (4.20) and the fact that $\boldsymbol{U} = \Lambda(B(p))$ implies $\tilde{\boldsymbol{U}} = \delta\Lambda(B(p);\tilde{B}(p))$. We may now readily substitute (4.23) into the expression (4.13) to obtain

$$\boldsymbol{\delta J}_0(p;\tilde{p}) = \int_0^T \int_{x_0}^{x_L} \boldsymbol{\sigma}^T(x,t)\left(\partial_B \boldsymbol{S}(\boldsymbol{U},B)\boldsymbol{U}\right)(x,t)\tilde{B}(x,t)\; dxdt. \tag{4.24}$$

By utilizing the fact that

$$\left[\tilde{B}(p)\right](x,t) = \delta B(p;\tilde{p}) = [\delta(B_0 + pB_1)](p;\tilde{p}) = B_1(x)\,\tilde{p}(t), \tag{4.25}$$

we further simplify (4.24) to

$$\boldsymbol{\delta J}_0(p;\tilde{p}) = \int_0^T \left(\int_{x_0}^{x_L} \boldsymbol{\sigma}^T(x,t)\left(\partial_B \boldsymbol{S}(\boldsymbol{U},B)\boldsymbol{U}\right)(x,t)B_1(x)\; dx\right)\tilde{p}(t)\; dt. \tag{4.26}$$

Therefore from definition (4.11), we obtain the following (formal) gradient from (4.26)

$$\nabla \boldsymbol{J}_0(p)(t) = \int_{x_0}^{x_L} \boldsymbol{\sigma}^T(x,t)\left(\partial_B \boldsymbol{S}(\boldsymbol{U},B)\boldsymbol{U}\right)(x,t)B_1(x)\; dx\,. \tag{4.27}$$

We again remark that, in the case of the SWEs, we have

$$\partial_B \boldsymbol{S}(\boldsymbol{U},B)\boldsymbol{U} = \begin{bmatrix} 0 & 0 \\ -g & 0 \end{bmatrix}\begin{bmatrix} h \\ hu \end{bmatrix} = \begin{bmatrix} 0 \\ -gh \end{bmatrix}, \qquad B_1 = \partial_x b_1, \tag{4.28}$$

and therefore the gradient is simplified to the form

$$\begin{aligned}\nabla \boldsymbol{J}_0(p) &= \int_{x_0}^{x_L} \left(\begin{bmatrix} \sigma_1 \\ \\ \sigma_2 \end{bmatrix}^T \begin{bmatrix} 0 \\ \\ -gh \end{bmatrix} \right) (x,t)\partial_x b_1(x)\ dx \\ &= \int_{x_0}^{x_L} -g\sigma_2(x,t)h(x,t)\partial_x b_1(x)\ dx,\end{aligned} \tag{4.29}$$

where $\boldsymbol{\sigma} = \begin{bmatrix} \sigma_1 \\ \\ \sigma_2 \end{bmatrix}$ is the solution of the adjoint equation (4.20).

4.3.3 Numerical Scheme for the Inverse Problem

In this subsection we will discuss the numerical algorithms for the inverse problem. The DG scheme will be employed to solve the adjoint problem (4.20) and an iterative method will be presented to update the function p with the suitably chosen regularization terms.

Discontinuous Galerkin Method for the Adjoint Problem

Noting that the spatial derivative in the adjoint problem (4.20) is not in the conservative form, we start by reformulating the adjoint problem as a balance law of the form

$$\partial_t \boldsymbol{\sigma} + \partial_x \left(\boldsymbol{A}^T \boldsymbol{\sigma} \right) = (\partial_x \boldsymbol{A}^T - \boldsymbol{S}^T - \boldsymbol{C}^T)\boldsymbol{\sigma}, \tag{4.30}$$

where $\boldsymbol{A} = \boldsymbol{A}(\boldsymbol{U})$ does not depend on the unknown $\boldsymbol{\sigma}$. Following the same discretization strategy as presented in Section 4.2.2, we seek an approximate solution $\boldsymbol{\sigma}_\tau$ in which $\boldsymbol{\sigma}_\tau^{(i)}$ for $i = 1, ..., m$ belong to $\mathbb{V}_\tau^k$. The DG method in cell I_j becomes

$$\begin{aligned}&\int_{I_j} \partial_t \boldsymbol{\sigma}_\tau \boldsymbol{v}\ dx - \int_{I_j} \boldsymbol{A}(\boldsymbol{U}_\tau)^T \boldsymbol{\sigma}_\tau \partial_x v\ dx + \hat{\boldsymbol{G}}_{j+\frac{1}{2}} \boldsymbol{v}^-_{j+\frac{1}{2}} - \hat{\boldsymbol{G}}_{j-\frac{1}{2}} \boldsymbol{v}^+_{j-\frac{1}{2}} \\ &\qquad = \int_{I_j} \left(\partial_x \boldsymbol{A}^T(\boldsymbol{U}_\tau) - \boldsymbol{S}^T(B_\tau) - \boldsymbol{C}^T \right) \boldsymbol{\sigma}_\tau \boldsymbol{v}\ dx,\end{aligned} \tag{4.31}$$

where $\boldsymbol{v} \in \mathbb{V}_\tau^k$ is a vector of test functions and the Lax-Friedrichs numerical flux takes the form

$$\hat{\boldsymbol{G}}_{j+\frac{1}{2}} = \frac{1}{2}\left(\boldsymbol{A}(\boldsymbol{U}^-_{\tau,j+\frac{1}{2}})^T\boldsymbol{\sigma}^-_{\tau,j+\frac{1}{2}} + \boldsymbol{A}(\boldsymbol{U}^+_{\tau,j+\frac{1}{2}})^T\boldsymbol{\sigma}^+_{\tau,j+\frac{1}{2}} - \alpha\left(\boldsymbol{\sigma}^+_{\tau,j+\frac{1}{2}} - \boldsymbol{\sigma}^-_{\tau,j+\frac{1}{2}}\right)\right), \quad (4.32)$$

with the value of α being the same as in the forward DG scheme, described in Section 4.2.2.

Regularization & Update Scheme

In this subsection, we describe the numerical method designed to recover the function $p(t)$ via an iterative scheme. Usually, either a descent type [27], Newton type [74], or a trust region algorithm [141] is employed. A Newton type algorithm usually provides a certain acceleration to the convergence, but as a trade off, it is usually more computationally expensive. In this work, we employ a descent type algorithm, in light of the fact that our functional is highly non-linear and highly non-convex, applying a higher order method may result in getting stuck at a local optimum even more easily.

We employ an operator splitting algorithm to update the function $p(t)$. The scheme is initialized with a random initial guess for p, denoted by $p^{0,\text{noisy}}$. We use multiplicative noise following a uniform distribution, i.e., $p^{0,\text{noisy}} = p^0\nu$, where $\nu \sim \mathcal{U}\left[1-\frac{1}{2}\eta_p, 1+\frac{1}{2}\eta_p\right]$, to define the random initial guess. Not only is the update of p dependent on $\nabla \boldsymbol{J}_0$, but it also relies on a regularization term. The regularization term ensures the optimization problem is locally convex and makes it possible to solve an ill-posed problem efficiently by incorporating a-priori knowledge of the profile to be reconstructed. Various regularization terms have been constructed for different purposes. For instance, L^1 regularization [161, 65] results in a simpler sparse solution. On the other hand, TV regularization [130] favors piecewise

constant functions of the coefficients to be recovered, whereas Sobolev regularization [50] favors smoothness of the coefficients to be reconstructed.

Before we focus on our choice of regularization, we first discuss the update algorithm. To better motivate our choice of algorithm, we start by simplifying our discussion and considering the situation when there is only one regularization term. In this case, the proximal gradient descent method (or the forward-backward splitting) [4, 113] is a common choice. The explicit term is usually assigned as the term coming from the gradient of a more complicated functional. The implicit term is typically chosen so that the proximal map is easy to evaluate and the stability of the algorithm is increased. Consequently, the k^{th} iteration is given by

$$p^{k+1} = p^k - \ell_k \nabla \boldsymbol{J}_0 \left(p^k\right) - \ell_k \partial \boldsymbol{R} \left(p^{k+1}\right), \tag{4.33}$$

where ℓ_k is the step size or learning rate and ∂ represents the subgradient when the proximal map of $\boldsymbol{R}$ can be computed. In our work, we will choose a constant step size, i.e., $\ell_k = \ell$ for all k. The scheme can be rewritten so that the update for iteration $k+1$ only depends on the information from iteration k,

$$p^{k+1} = (\mathcal{I} + \ell \partial \boldsymbol{R})^{-1} \left(p^k - \ell \nabla \boldsymbol{J}_0 \left(p^k\right)\right), \tag{4.34}$$

where $\mathcal{I}$ is the identity matrix and

$$(\mathcal{I} + \ell \partial \boldsymbol{R})^{-1} (w) = \text{argmin}_y \left\{ \boldsymbol{R}(y) + \frac{1}{2\ell} \|w - y\|_2^2 \right\} = \text{prox}_{R,\ell}(w). \tag{4.35}$$

This leads to the formulation

$$p^{k+1} = \text{argmin}_y \left\{ \boldsymbol{R}(y) + \frac{1}{2\ell} \|p^k - \ell \nabla \boldsymbol{J}_0(p^k) - y\|_2^2 \right\}. \tag{4.36}$$

A common choice for the regularization is L^1 regularization, where $\boldsymbol{R}(y) = \gamma\|y - p_0\|_1$ with p_0 being a chosen coefficient of homogeneous background and γ a scalar parameter, aiming to impose sparsity of the difference between the resulting optimum and p_0. The proximal gradient method coming from this choice of regularizer is

$$\begin{aligned} p^{k+1} &= \text{argmin}_y \left\{ \gamma\|y - p_0\|_1 + \frac{1}{2\ell}\|p^k - \ell\nabla \boldsymbol{J}_0\left(p^k\right) - y\|_2^2 \right\} \\ &= \mathcal{S}_{\gamma\ell}\left(p^k - \ell\nabla \boldsymbol{J}_0\left(p^k\right) - p_0\right) + p_0, \end{aligned} \tag{4.37}$$

where the shrinkage operator $\mathcal{S}_{\gamma\ell}$ [39, 38, 161] is given as follows

$$\mathcal{S}_{\gamma\ell}\left(p\right) = \text{sign}\left(p\right)\max\left\{|p| - \ell\gamma, 0\right\}. \tag{4.38}$$

After briefly describing the simple motivating example which carries only one regularization term, we now describe the combination of regularization terms that we use in our work, and how we perform the operator splitting in our algorithm. In this work, the regularization term is taken as a sum of two regularizers

$$\boldsymbol{R}(p) = \boldsymbol{R}_{L^1}(p - p_0) + \boldsymbol{R}_{H^1}(p), \tag{4.39}$$

where $\boldsymbol{R}_{L^1}(p) = \|p\|_1$ represents L^1 regularization and $\boldsymbol{R}_{H^1}(p) = \|\nabla p\|_2^2$ represents H^1 regularization. The L^1 regularization term will aid in removing the noise by sparsifying it, while the H^1 regularization term will be beneficial for the purpose of smoothing out the noisy data, an advantage over total variation ($\|\nabla p\|_1$) regularization. H^1 regularization has been shown to be good for flow control problems [37, 71] as well as image reconstruction and deblurring [108, 84].

Now, we wish to minimize $\boldsymbol{J}_0(p) + \boldsymbol{R}_{L^1}(p) + \boldsymbol{R}_{H^1}(p)$, which reduces to finding p so that

$$0 \in \nabla \boldsymbol{J}_0(p) + \partial \boldsymbol{R}_{L^1}(p) + \nabla \boldsymbol{R}_{H^1}(p). \tag{4.40}$$

We must be careful in our approach and employ a more complicated splitting scheme than the proximal gradient descent since we now have an additional operator. In our work, we adapt the three-operator splitting algorithm [40], which we will describe in detail in Algorithm 1. For simplicity, we introduce the following notation corresponding to each regularization term

$$\begin{aligned}\mathcal{J}_{\ell\gamma_L \boldsymbol{R}_{L^1}}(\omega) &= \left(\mathcal{I} + \ell\gamma_L \boldsymbol{R}_{L^1}\right)^{-1}(\omega) \\ &= \operatorname{sign}(\omega - p_0)\max\left\{|\omega - p_0| - \ell\gamma_L, 0\right\} + p_0,\end{aligned} \tag{4.41}$$

and

$$\mathcal{J}_{\ell\gamma_H \boldsymbol{R}_{H^1}}(\omega) = \left(\mathcal{I} + \ell\gamma_H \boldsymbol{R}_{H^1}\right)^{-1}(\omega) = \left(\mathcal{I} - \ell\gamma_H \Delta\right)^{-1}(\omega), \tag{4.42}$$

where γ_L is the L^1 regularization parameter and γ_H is the H^1 regularization parameter. Furthermore, we would like to note that the gradient of the cost function is time-dependent, i.e. $\nabla \boldsymbol{J}_0(p^k, t)$, but we denote it as $\nabla \boldsymbol{J}_0(p^k)$ for the sake of simplifying notation. The update becomes

$$p^{k+1} = \mathcal{J}_{\ell\gamma_L \boldsymbol{R}_{L^1}} \circ \left[z^k + \lambda_k \left(\mathcal{J}_{\ell\gamma_H \boldsymbol{R}_{H^1}} \circ \left[2p^k - z^k - \ell\nabla \boldsymbol{J}_0(p^k)\right] - p^k\right)\right], \tag{4.43}$$

where z^0 is originally initialized to be p^0 and λ_k is the relaxation parameter which can be used to help speed up the rate of convergence of the iterative solutions. We are now ready to introduce our algorithm.

Algorithm 1: Three-Operator Splitting Algorithm

initialize p^0 to be the random initial guess;

initialize $z^0 = p^0$;

set regularization parameters γ_L, γ_H;

set relaxation parameter $(\lambda_k)_{k\geq 0}$;

set learning rate ℓ;

for $k = 0, 1, ...$ **do**

compute $\Lambda(B(p^k))$ from $B(p^k)$ by solving the forward problem (4.2) ;

compute $\boldsymbol{\sigma}$ from $(B(p^k), \Lambda(B(p^k)))$ by solving the adjoint problem (4.20) ;

evaluate $\nabla \boldsymbol{J}_0(p^k) = \int_{x_0}^{x_L} \boldsymbol{\sigma}^T(x,t) \left(\partial_B \boldsymbol{S}(\boldsymbol{U}, B)\boldsymbol{U}\right)(x,t) B_1(x)\, dx$;

define $\kappa^k = 2p^k - z^k - \ell \nabla \boldsymbol{J}_0(p^k)$;

evaluate $\omega^k = \mathcal{J}_{\ell\gamma_H \boldsymbol{R}_{H^1}}(\kappa^k) = (\mathcal{I} - \ell\gamma_H \nabla^2)^{-1}(\kappa^k)$;

update $z^{k+1} = z^k + \lambda_k(\omega^k - p^k)$;

update $p^{k+1} = \mathcal{J}_{\ell\gamma \boldsymbol{R}_{L^1}}(z^{k+1}) = \text{sign}(z^{k+1} - p_0) \max\{|z^{k+1} - p_0| - \ell\gamma_L, 0\} + p_0$;

end

4.4 Numerical Examples for the Shallow Water Equations

In this section, we will be considering the one-dimensional nonlinear SWEs (4.5). We aim to recover the temporal component $p(t)$ in the bottom topography function $b(x,t)$, see (4.3).

In all the numerical tests, we use a relaxation parameter of $\lambda_k = 1$, a noise parameter for the measured data of $\eta_{meas} = 0.1$ or 5% noise, a noise parameter for the initial

guess p^0 of $\eta_p = 0.25$ or 12.5% noise. A coefficient of homogeneous background is assumed to be known and taken as $p_0 = 1$. Each test is ran for 1000 iterations. The iteration with the smallest residue, $\boldsymbol{J}_0(p^k)$, is selected as the best recovered representation for the true temporal component of the bottom function $p(t)$. The measured data is computed using a high order accurate DG method with a uniform mesh of 400 cells and P^3 piecewise polynomials, with noise added to the DG solutions to represent noisy measurement. A mesh of 50 uniform cells with P^2 piecewise polynomials are used to solve the forward problem and a uniform mesh of 25 cells with P^1 piecewise polynomials are used to solve the adjoint problem. The measured data, forward, and adjoint solvers are designed with different meshes and polynomial degree approximations in an effort to avoid committing "inverse crime"[148]. The adjoint solver is chosen to be less accurate than the forward solver since it has a helpful regularization effect.

4.4.1 Tests for Recovering Different Time Profiles of $p(t)$

In this subsection, we will perform numerical experiments aiming to recover several unknown time profiles, $p_{true}(t)$, from noisy boundary measurements.

We solve the forward problem (4.1) with the DG method described in Section 4.2.2 where our computational domain is chosen to be $[x_0, x_L] = [0, 1]$, the initial conditions are given by

$$h(x, 0) = 7 + \exp(\sin(2\pi x)), \qquad hu(x, 0) = \cos(2\pi x), \tag{4.44}$$

and the spatial components of the bottom topography function are defined as

$$b_0(x) = \cos(\sin(2\pi x), \qquad b_1(x) = \sin^2(\pi x)\,. \tag{4.45}$$

The final time is set as $T = 0.05$ and periodic boundary conditions are used.

We examine several choices for the true value of $p(t)$ and the corresponding initial guesses, which are outlined in Table 4.1. A constant learning rate of $\ell = 0.6$ is used in each test. The regularization parameters are fixed with $\gamma_L = 1 \times 10^{-6}$ in all examples and $\gamma_H = 5 \times 10^{-8}$ in cases (4.4.1a),(4.4.1b), (4.4.1d), (4.4.1e), $\gamma_H = 1 \times 10^{-8}$ in cases (4.4.1c), (4.4.1d) and $\gamma_H = 5 \times 10^{-9}$ in case (4.4.1f).

Case	$p_{true}(t)$	$p^0(t)$
(4.4.1a)	$e^{\beta\left(t-\frac{1}{3}T\right)^2} + 1$	$e^{\beta\left(t-\frac{2}{3}T\right)^2} + 1$
(4.4.1b)	$e^{\beta\left(t-\frac{2}{3}T\right)^2} + 1$	$e^{\beta\left(t-\frac{1}{3}T\right)^2} + 1$
(4.4.1c)	$e^{2\beta\left(t-\frac{1}{4}T\right)^2} + e^{2\beta\left(t-\frac{3}{4}T\right)^2} + 1$	$\frac{3}{2}e^{\beta\left(t-\frac{1}{2}T\right)^2} + 1$
(4.4.1d)	$e^{\beta(t-0.3T)^2} + \frac{3}{2}e^{2\beta(t-0.7T)^2} + 1$	$3\cos^2\left(\frac{10\pi}{T}t\right) + \frac{3}{4}$
(4.4.1e)	$\frac{3}{2}e^{\beta(t-0.3T)^2} + e^{2\beta(t-0.7T)^2} + 1$	$3\cos^2\left(\frac{10\pi}{T}t\right) + \frac{3}{4}$
(4.4.1f)	$e^{4\beta\left(t-\frac{1}{4}T\right)^2} + \frac{3}{2}e^{4\beta\left(t-\frac{1}{2}T\right)^2} - \frac{1}{2}e^{4\beta\left(t-\frac{3}{4}T\right)^2} + 1$	$3\cos^2\left(\frac{10\pi}{T}t\right) + \frac{3}{4}$

Table 4.1: The true function for $p(t)$ denoted as p_{true} and the corresponding initial guess used, p^0 with $\beta = -10,000$. Multiplicative noise is applied to p^0 in the simulations.

Cases (4.4.1a) and (4.4.1b) represent the situation in which the true value of $p(t)$ is a bump function that is non-centered with respect to the time interval and the corresponding initial guess is a noisy horizontal shift of $p_{true}(t)$. The numerical results are shown in Figures 4.1 and 4.2. In both cases, the amplitude and shape of the true function and recovered numerical approximation are very close. The figures demonstrate that the scheme is robust even in the presence of multiplicative noise and the ill-posedness of the problem.

In case (4.4.1c), we examine a true function p that has two bumps of equal amplitude with an initial guess consisting of one bump with a larger amplitude. The results can be found in Figure 4.3. Cases (4.4.1d) and (4.4.1e) also include a true p function of two bumps, however they have different amplitudes and the corresponding initial guesses are

highly oscillatory trigonometric functions. The corresponding results are shown in Figures 4.4 and 4.5. In the examples with two bumps, the reconstructed function was also able to identify the two crests. The effect of the parameter γ_H is explored in these cases. The value $\gamma_H = 1 \times 10^{-8}$ is used for Cases (4.4.1c) and (4.4.1d). We can see that this smaller choice of γ_H results in a less smooth solution in comparison with the results from Cases (4.4.1d) and (4.4.1e) when $\gamma_H = 5 \times 10^{-8}$ is used. However, the plots corresponding to $\gamma_H = 5 \times 10^{-8}$ while more smooth, are more flattened.

The case (4.4.1f) contains two crests of different amplitudes and a trough for the true function with a highly oscillatory trigonometric function as the initial guess. Plots of the results corresponding to this case can be found in Figure 4.6. Additionally, for case (4.4.1f) we show the solutions of the forward problem in Figure 4.7 at different times ($t = \frac{T}{4}, \frac{T}{2}, \frac{3T}{4}$, and T). The water surface height, bottom topography, and water discharge of the measured data and the numerical solution at the iteration with the smallest residue are compared. The recovered bottom topography along with the recovered state variables match the true functions well, even for some more complicated choices of $p(t)$.

Figures 4.1-4.6 each contain a plot of the residues $\boldsymbol{J}_0$, defined by (4.7) and (4.8), at the endpoints of the spatial domain for each iteration on a log-log scale. We see in each case a similar behavior occurs in which an 'elbow'-like shape appears. The portion of this residue curve with a steeper slope corresponds to the situation when the term $\boldsymbol{J}_0$ has a greater impact on the update of the function p, which happens for the beginning iterations. The flat portion of the residue curve corresponds to the situation when the iteration starts to enter a small neighborhood where the regularization term $\boldsymbol{R}(p)$ convexifies the optimization

problem and dominates the update.

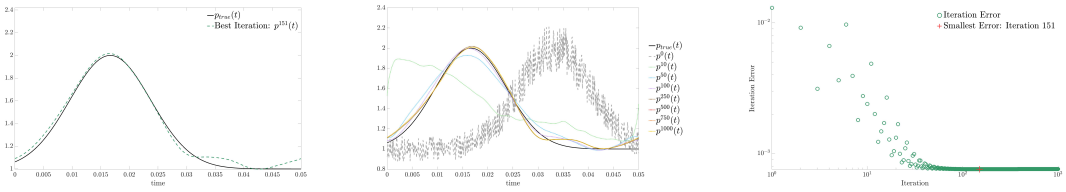


Figure 4.1: Results for Case (4.4.1a). Left: plots of the true p and the numerically recovered p at iteration 151, corresponding with the smallest residue; Middle: plots of the true p, the noisy initial guess, and various iteration values for p; Right: iteration errors on a log-log scale.

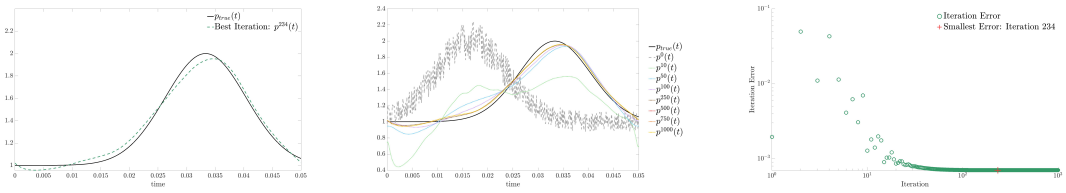


Figure 4.2: Results for Case (4.4.1b). Left: plots of the true p and the numerically recovered p at iteration 234, corresponding with the smallest residue; Middle: plots of the true p, the noisy initial guess, and various iteration values for p; Right: iteration errors on a log-log scale.

4.4.2 Tests for Recovering $p(t)$ from Different Initial Guesses

In this subsection, we run simulations with different initial guesses of $p^0(t)$ to recover the same $p_{true}(t)$. The goal is to demonstrate that the ability of our algorithm in recovering $p_{true}(t)$ does not depend on the initial guess.

We consider the forward problem with the initial conditions in (4.44) and the spatial bottom topography functions described in (4.45). The true time component of the

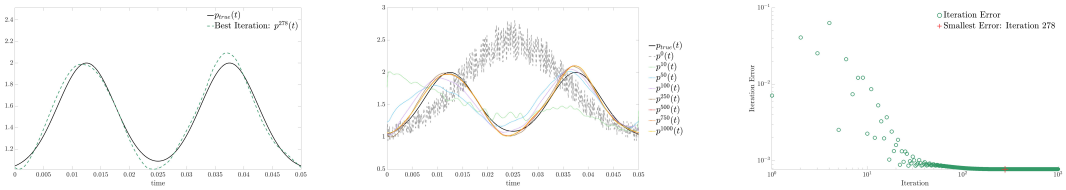


Figure 4.3: Results for Case (4.4.1c). Left: plots of the true p and the numerically recovered p at iteration 278, corresponding with the smallest residue; Middle: plots of the true p, the noisy initial guess, and various iteration values for p; Right: iteration errors on a log-log scale.

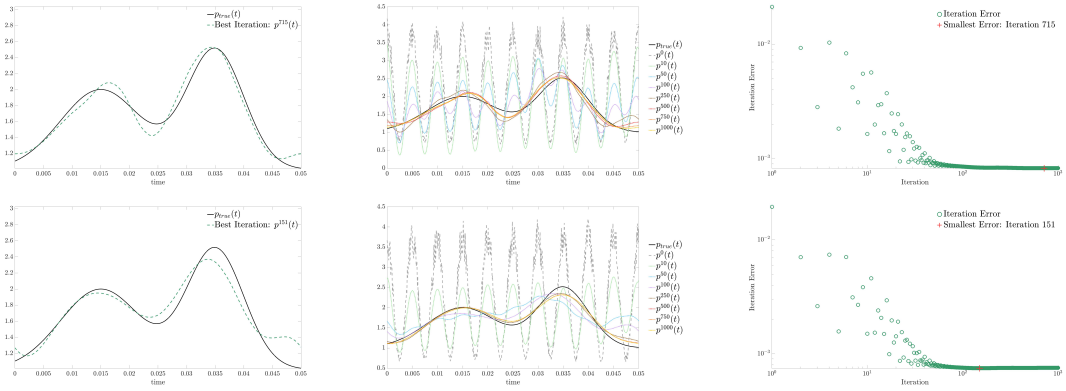


Figure 4.4: Results for Case (4.4.1d). Top row: $\gamma_H = 1\times10^{-8}$. Bottom row: $\gamma_H = 5\times10^{-8}$. Left Column: plots of the true p and the numerically recovered p at iteration 715 (top row) and 151 (bottom row), corresponding with the smallest residue; Middle Column: plots of the true p, the noisy initial guess, and various iteration values for p; Right Column: iteration errors on a log-log scale.

bottom topography function is fixed to be

$$p(t) = \exp\left(\beta(t-0.3T)^2\right) + \frac{3}{2}\exp\left(2\beta(t-0.7T)^2\right) + 1, \tag{4.46}$$

with $\beta = -10,000$. Four different representative initial guesses, $p^0(t)$, listed in Table 4.2, will be tested. In all cases the final time is $T = 0.05$ (while the solution is still smooth) and periodic boundary conditions are used. The remaining hyperparameters include a learning rate of $\ell = 0.6$, $\gamma_L = 1 \times 10^{-6}$, and $\gamma_H = 1 \times 10^{-8}$.

The numerical results for Cases (4.4.2a), (4.4.2b), and (4.4.2c) are shown in Figures

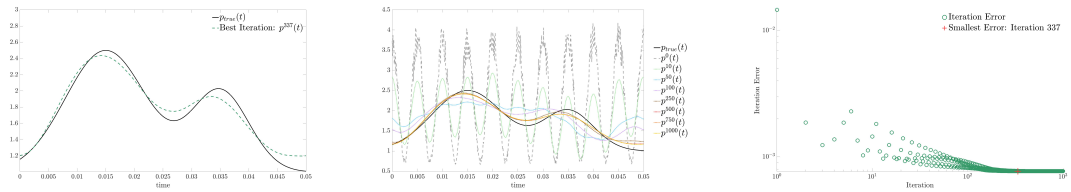


Figure 4.5: Results for Case (4.4.1e). Left: plots of the true p and the numerically recovered p at iteration 337, corresponding with the smallest residue; Middle: plots of the true p, the noisy initial guess, and various iteration values for p; Right: iteration errors on a log-log scale.

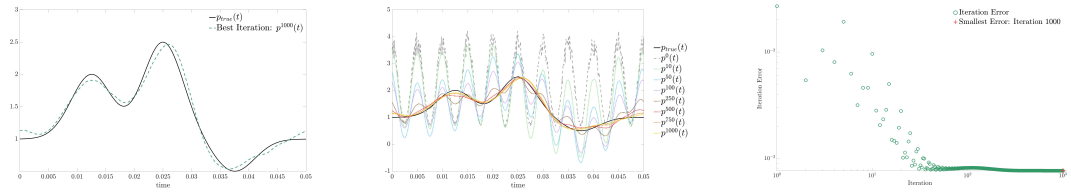


Figure 4.6: Results for Case (4.4.1f). Left: plots of the true p and the numerically recovered p at iteration 1000, corresponding with the smallest residue; Middle: plots of the true p, the noisy initial guess, and various iteration values for p; Right: iteration errors on a log-log scale.

4.8 - 4.10, while the results for Case (4.4.2d) can be found in Figure 4.4. In all four cases, the scheme was able to identify that $p_{true}(t)$ was a function consisting of two bumps, with the left bump (occurring earlier in time) having a smaller amplitude than the right bump (occurring later in time). This indicates the true function $p(t)$ can be recovered with the initial condition chosen from a wide range of functions. The numerical performances are similar in appearance and convergence rate. Cases (4.4.2a) - (4.4.2c) all achieve their best guess in less than 300 iterations. The true p and corresponding p with smallest residue error tend to have the some discrepancy near the final time T. The exception is Case (4.4.2a) in which the initial guess for p at time T is near to $p_{true}(T)$.

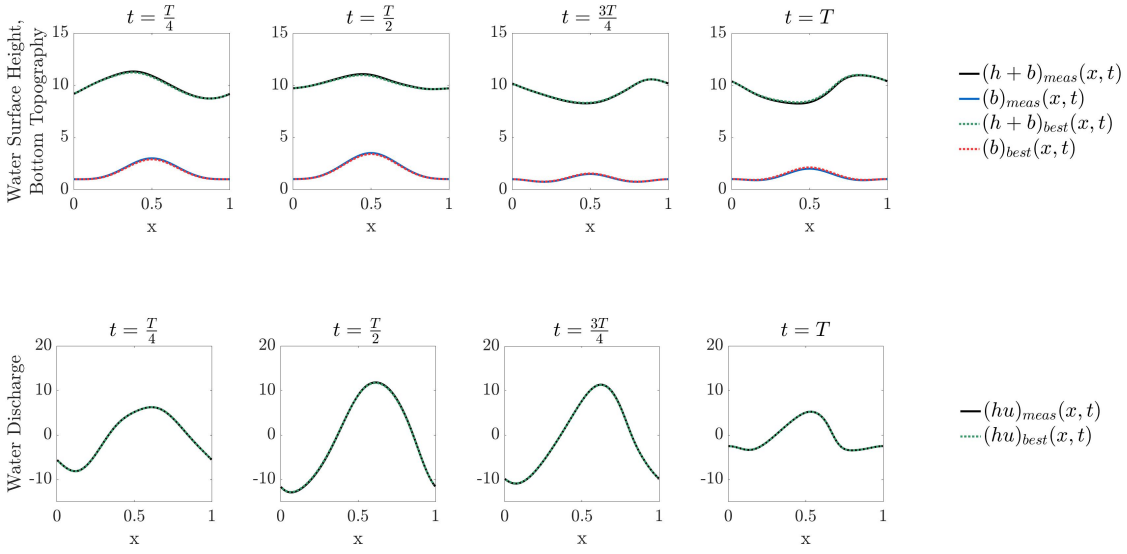


Figure 4.7: Comparison between the measured forward solutions and the results from the best iteration for SWEs Case (4.4.1f) at times $t = \frac{T}{4}, \frac{T}{2}, \frac{3T}{4}$, and T. In the top row the bottom topography function, b, and the water surface heights, $h + b$, are compared. In the bottom row, the water discharge, hu, is compared. The measured data and results from the iterative scheme are well matched for all functions in for each of the selected time snapshots.

4.4.3 Recovering $p(t)$ with Spatial Discontinuities in the Solutions of the Forward Problem

In the previous two subsections, the final stopping time is chosen to ensure that the solutions are smooth over the entire computational region. One well-known fact of hyperbolic conservation laws is that discontinuities may appear even when the initial conditions are smooth. In this subsection, we explore the capability of the proposed inverse algorithms when the solutions contain spatial discontinuities. We use the same initial conditions as in (4.44) and spatial components of the bottom topography functions as in (4.45). We run the forward problem for a longer time so that discontinuities form in the solutions. Here, we set the final time as $T = 0.2$. We consider two choices for $p_{true}(t)$ in this subsection, which

Case	$p^0(t)$
(4.4.2a)	1
(4.4.2b)	$4\sin^2\left(\frac{\pi}{T}t\right)$
(4.4.2c)	$-2\sin^2\left(\frac{\pi}{T}t\right)+2$
(4.4.2d)	$3\cos^2\left(\frac{10\pi}{T}t\right)+0.75$

Table 4.2: The corresponding initial guesses used, p^0 with $T = 0.05$. Multiplicative noise is applied to p^0 in the simulations.

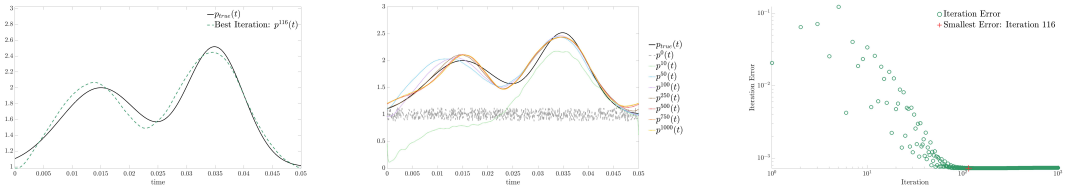


Figure 4.8: Results for Case (4.4.2a). Left: plots of the true p and the p corresponding with the smallest residue error, at iteration 116; Middle: plots of the true p, the noisy initial guess, and various iteration values for p; Right: iteration errors on a log-log scale.

are listed in Table 4.3 along with their corresponding initial guesses.

Case	$p_{true}(t)$	$p^0(t)$
(4.4.3a)	$e^{\beta\left(t-\frac{1}{2}T\right)^2}+1$	1
(4.4.3b)	$e^{4\beta\left(t-\frac{1}{4}T\right)^2}+\frac{3}{2}e^{4\beta\left(t-\frac{1}{2}T\right)^2}-\frac{1}{2}e^{4\beta\left(t-\frac{3}{4}T\right)^2}+1$	1

Table 4.3: The true function for $p(t)$ denoted as p_{true} and the corresponding initial guess used, p^0 with $\beta = -700$. Multiplicative noise is applied to p^0 in the simulations.

To capture the discontinuities well and remove the possible oscillations, a slope limiter is often employed in the DG method. We implement two different slope limiters for generating the measured data, as well as for solving the forward problem in the inverse scheme. The simple minmod limiter [35] is employed along the characteristic direction to generate the measured data. On the other hand, the WENO limiter, introduced by Qiu and Shu in [121] is used for the forward solver within the iterative inverse scheme. This limiter is known to be robust and it is able to capture the sharp transition of the discontinuities.

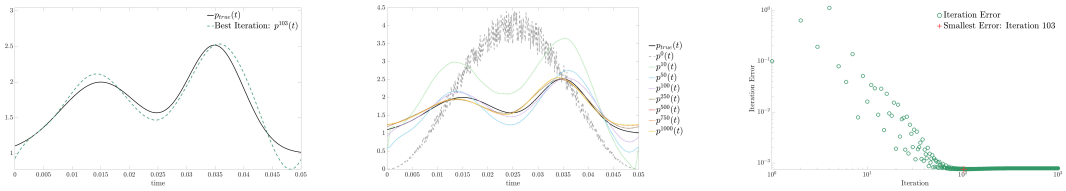


Figure 4.9: Results for Case (4.4.2b). Left: plots of the true p and the p corresponding with the smallest residue error, at iteration 103; Middle: plots of the true p, the noisy initial guess, and various iteration values for p; Right: iteration errors on a log-log scale.

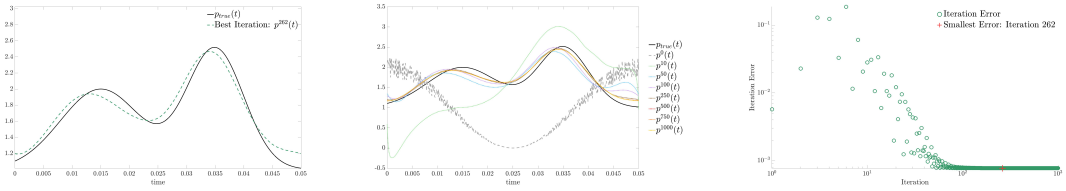


Figure 4.10: Results for Case (4.4.2c). Left: plots of the true p and the p corresponding with the smallest residue error, at iteration 262; Middle: plots of the true p, the noisy initial guess, and various iteration values for p; Right: iteration errors on a log-log scale.

The implementation of two different slope limiters was in an effort to avoid 'inverse crime'.

A learning rate of $\ell = 0.02$ was implemented with the regularization parameters $\gamma_L = 1 \times 10^{-4}$ and $\gamma_H = 1 \times 10^{-6}$. Results for case (4.4.3a) are found in Figures 4.11 and 4.12 while Figures 4.13 and 4.14 contain the results for test (4.4.3b). Note that periodic boundary conditions are employed, therefore the discontinuities can pass the right boundary and re-enter the domain through the left boundary at some time between $3T/4$ and T, which means the measured data include the information of discontinuities. In both cases, we observe that our algorithm can recover the exact function $p_{true}(t)$ well, and the results are comparable with those containing smooth data only. We have included the comparison between the measured data of the water surface height, bottom topography function, water discharge, and the corresponding numerical solutions at the iteration with the smallest residue, at different times $t = \frac{T}{4}, \frac{T}{2}, \frac{3T}{4}$, and T, from which we can observe the numerical

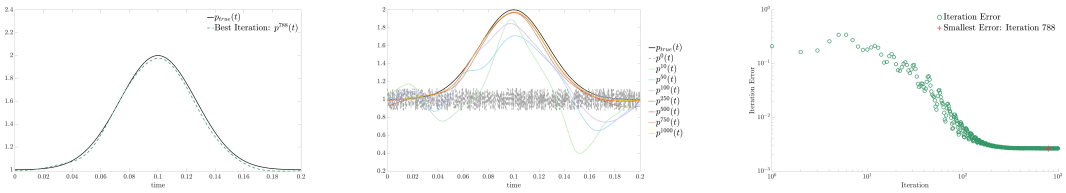


Figure 4.11: Results for Case (4.4.3a). Left: plots of the true p and the p corresponding with the smallest residue error, at iteration 788; Middle: plots of the true p, the noisy initial guess, and various iteration values for p; Right: iteration errors on a log-log scale.

solutions match the measured data well. This elucidates the numerical scheme developed for the inverse problems can recover the true $p(t)$ well even when discontinuities develop in the solutions of the forward problem and hence in the measured data.

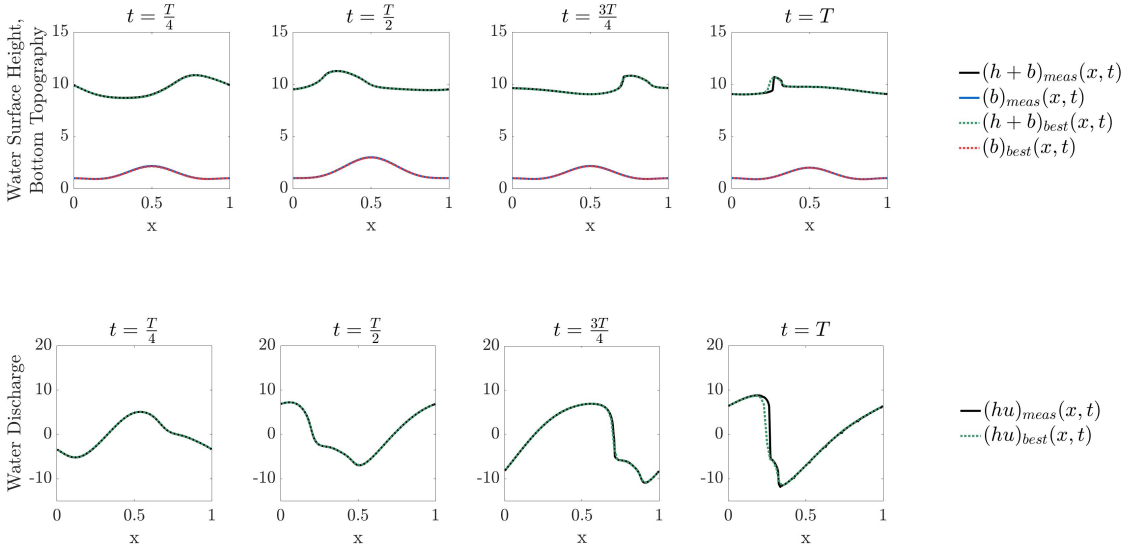


Figure 4.12: Comparison between the measured forward solutions and the numerical results from the iteration with the smallest residue error for Case (4.4.3a). The results shown are for 4 different time snapshots. The bottom topography function, b, water surface height, $h + b$ (top row), and the water discharge, hu (bottom row), are compared.

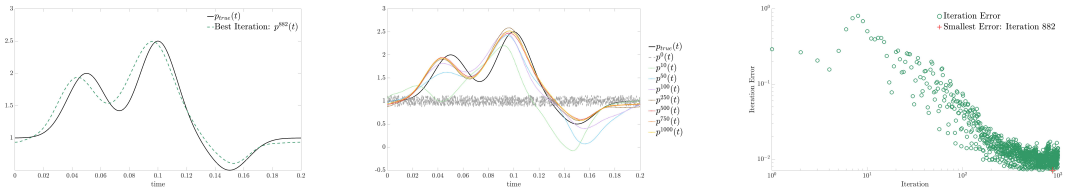


Figure 4.13: Results for Case (4.4.3b). Left: plots of the true p and the p corresponding with the smallest residue error, at iteration 882; Middle: plots of the true p, the noisy initial guess, and various iteration values for p; Right: iteration errors on a log-log scale.

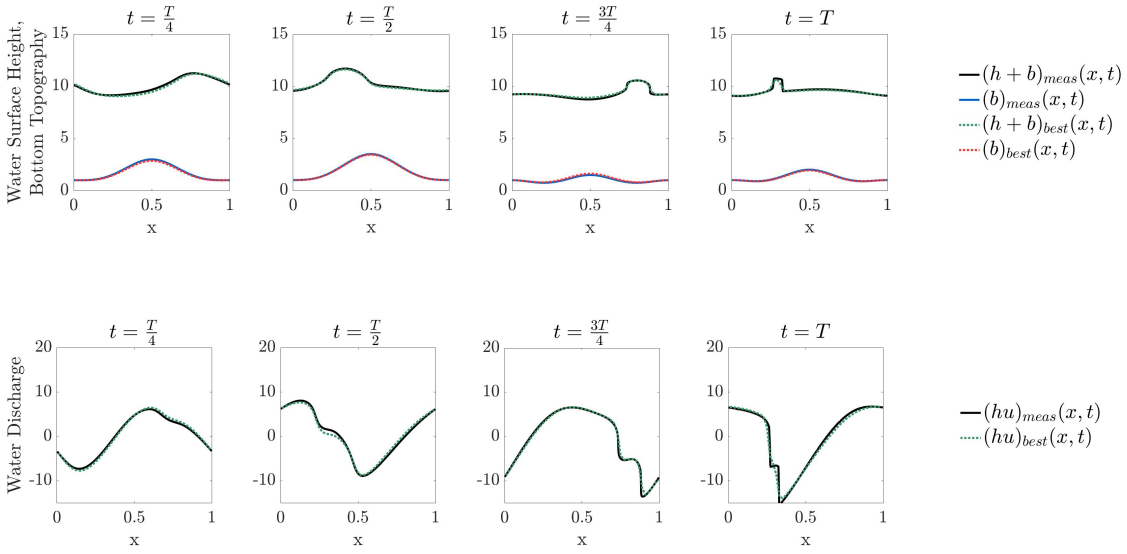


Figure 4.14: Comparison between the measured forward solutions and the numerical results from the iteration with the smallest residue error for Case (4.4.3b). The results shown are for 4 different time snapshots. The bottom topography function, b, water surface height, $h + b$ (top row), and the water discharge, hu (bottom row), are compared.

Chapter 5

Conclusions & Future Work

5.1 Chapter Summaries

In Chapter 2, well-balanced DG methods for the shallow water equations with horizontal temperature gradients, also known as the Ripa model, are designed and tested. We presented two types of well-balanced methods, one for the the simpler still-water equilibrium (2.5) and the other for more complicated moving-water equilibrium (2.8), and showed that the former one is a special case of the latter. We also demonstrated the same framework can be extended to design well-balanced methods for the isobaric steady state (2.6) and the constant water height steady state (2.7), with different definitions of $\boldsymbol{U}_\tau^e(x)$ and $\boldsymbol{U}^{*,\pm}_{\tau,j+\frac{1}{2}}$ following the equilibria to be preserved. The proposed method is an extension of the well-balanced method [150] for the shallow water equations, but with several improvements to simplify the algorithms. To achieve the well-balanced property, special attention was paid to the approximation of the source term and the construction of the numerical fluxes. Numerical examples were given to demonstrate the accuracy, well-balanced prop-

erty, perturbations to steady states, and non-oscillatory behavior near discontinuities. This approach is rather general and can be applied to design well-balanced methods for other hyperbolic balance laws.

In Chapter 3 we constructed and tested DG methods for the one-dimensional arterial blood flow system with the man-at-eternal-rest and living-man equilibria. Well-balanced DG methods are designed to efficiently capture the nearly equilibria flow which are small perturbation of these equilibrium states. We focus on the living-man equilibrium states which are more relevant to the practical problem. To construct well-balanced methods, special attention was paid to the projection of the initial conditions into piecewise polynomial space, the approximation of the source term, and the construction of the numerical fluxes. Extensive numerical examples were given to demonstrate the well-balanced property, accuracy, non-oscillatory behavior at discontinuities, and ability to resolve small perturbations to steady states. DG methods have been shown to be efficient for the hyperbolic balance laws on network, and it would be interesting to test the performance of the proposed methods on the arterial network blood flow simulations, which will be our future work.

Finally, in Chapter 4 we constructed and validated an adjoint-based approach for recovering the bottom topography function in the source term of the one dimensional SWEs, from the noisy measurement data at two boundaries of the domain. One novelty of this work is that the reconstruction of the bottom topography function is accomplished with only boundary data from a single measurement event. The adjoint scheme was determined by a linearization of the forward system, and has been derived for general hyperbolic balance

laws. Another contribution of this work is the inclusion of two regularization terms. These extra regularization terms in the numerical approach aided in convexifying and handling the ill-posedness of problem. The bottom topography function was recovered through an iterative process using a three-operator splitting descent method. Extensive numerical tests were carried out, which demonstrated that a variety of shapes for the true $p(t)$ function could be recovered regardless of the noisy initial guess.

All numerical tests presented in this thesis were commuted in the programming language Matlab. Each code base was written by the author of this thesis. Access to the code for each chapter is available from the author upon request.

5.2 Future Work

The work in Chapters 2 and 3 focused on developing well-balanced methods for various steady states (with zero and non-zero velocity) for two different systems of balance laws. We aim to extend this work by applying our methods to the model on networks, i.e. networks of rivers or arteries. We have begun working on extending the methods developed by [19] on networks of traffic laws to a system of equations from the family of shallow water equations in which the channel width can vary. The extension of this method to a network of arteries could be beneficial for coupling with higher dimensional models. The PDE model itself could be extended to improve its biological relevance.

There are numerous directions in which we aim to extend the inverse project presented in Chapter 4. First, we wish to test the scheme on other hyperbolic balance laws to see how it performs on systems with more equations or more complicated topographical

functions that arise in the source term. The second extension we would like to pursue is the inclusion of discontinuities in time in the numerical tests. The introduction of temporal discontinuities may require a more careful choice of regularization terms. Thus, the third component of future work includes a more robust study of regularization terms. Lastly, we would like to develop an inverse algorithm to recover the general bottom topography $B(x,t)$ and its extension in higher dimensional problems.

Bibliography

[1] E. Audusse, F. Bouchut, M. O. Bristeau, R. Klein, and B. Perthame. A fast and stable well-balanced scheme with hydrostatic reconstruction for shallow water flows. *SIAM Journal of Scientific Computing*, 25(6):2050–2065, 7 2004.

[2] M. K. Banda and M. Herty. Adjoint IMEX-based schemes for control problems governed by hyperbolic conservation laws. *Computational Optimization and Applications*, 51(2):909–930, 3 2012.

[3] A.J.C. Barré de Saint-Venant. Théorie du mouvement non-permanent des eaux, avec application aux crues des rivières et è l'introduction des marées dans leur lit. *C. R. Acad Sci Paris*, 1 1871.

[4] A. Beck. *First-Order Methods in Optimization*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 10 2017.

[5] J. Behrens and F. Dias. New computational methods in tsunami science. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 373(2053):20140382, 10 2015.

[6] J. Behrens and R LeVeque. Modeling and simulating tsunamis with an eye to hazard mitigation. *SIAM News*, 44(4):1–8, 2011.

[7] A. Beljadid, A. Mohammadian, and A. Kurganov. Well-balanced positivity preserving cell-vertex central-upwind scheme for shallow water flows. *Computers and Fluids*, 136:193–206, 9 2016.

[8] A. Bermudez and M. E. Vazquez. Upwind methods for hyperbolic conservation laws with source terms. *Computers and Fluids*, 23:1049–1071, 1994.

[9] A. Bernstein, A. Chertock, and A. Kurganov. Central-upwind scheme for shallow water equations with discontinuous bottom topography. *Bulletin of the Brazilian Mathematical Society*, 47(1):91–103, 3 2016.

[10] C. Berthon and F. Marche. A positive preserving high order VFRoe scheme for shallow water equations: A class of relaxation schemes. *SIAM Journal on Scientific Computing*, 30(5):2587–2612, 2007.

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

[12] A. Bollermann, G. Chen, A. Kurganov, and S. Noelle. A well-balanced reconstruction of Wet/Dry fronts for the shallow water equations. *Journal of Scientific Computing*, 56(2):267–290, 8 2013.

[13] A. Bollermann, S. Noelle, and M. Lukáčová-Medvid'ová. Finite Volume Evolution Galerkin methods for the shallow water equations with dry beds. *Communications in Computational Physics*, 10(2):371–404, 2011.

[14] F. Bouchut and T. Morales. A subsonic-well-balanced reconstruction scheme for shallow water flows. *SIAM Journal on Numerical Analysis*, 48:1733–1758, 2010.

[15] F. Bouchut and V. Zeitlin. A robust well-balanced scheme for multi-layer shallow water equations. *Discrete & Continuous Dynamical Systems - B*, 13:739–758, 2010.

[16] J. Britton and Y. Xing. High Order Still-Water and Moving-Water Equilibria Preserving Discontinuous Galerkin Methods for the Ripa Model. *Journal of Scientific Computing*, 82(2), 2 2020.

[17] J. Britton and Y. Xing. Well-balanced discontinuous galerkin methods for the one-dimensional blood flow through arteries model with man-at-eternal-rest and living-man equilibria. *Computers & Fluids*, 203:104493, 2020.

[18] S. B. Brunnermeier and S. A. Martin. Interoperability costs in the US automotive supply chain. *Supply Chain Management*, 7(2):71–82, 2002.

[19] J. Buli and Y. Xing. A discontinuous Galerkin method for the Aw-Rascle traffic flow model on networks. *Journal of Computational Physics*, submitted.

[20] S. Bunya, E. J. Kubatko, J. J. Westerink, and C. Dawson. A wetting and drying treatment for the Runge-Kutta discontinuous Galerkin solution to the shallow water equations. *Computer Methods in Applied Mechanics and Engineering*, 198(17-20):1548–1562, 4 2009.

[21] R. Bürger, A. Coronel, and M. Sepúlveda. Numerical solution of an inverse problem for a scalar conservation law modelling sedimentation. *Proceedings of Symposia in Applied Mathematics*, pages 445–454, 01 2009.

[22] S. Čanić. Blood flow through compliant vessels after endovascular repair: wall deformations induced by the discontinuous wall properties. *Computing and Visualization in Science*, 4(3):147–155, 2002.

[23] S. Čanić, B. Piccoli, J.-M. Qiu, and T. Ren. Runge–Kutta discontinuous Galerkin method for traffic flow model on networks. *Journal of Scientific Computing*, 63(1):233–255, 2015.

[24] W. Castaings, D. Dartus, M. Honnorat, F. X. Le Dimet, Y. Loukili, and J. Monnier. Automatic Differentiation: A Tool for Variational Data Assimilation and Adjoint Sensitivity Analysis for Flood Modeling. *Lecture Notes in Computational Science and Engineering*, 50:249–262, 2006.

[25] M. J. Castro, J. A. López-García, and C. Paŕes. High order exactly well-balanced numerical methods for shallow water systems. *Journal of Computational Physics*, 246:242–264, 2013.

[26] M. J. Castro, A. P. Milanés, and C. Parés. Well-balanced numerical schemes based on a generalized hydrostatic reconstruction technique. *Mathematical Models and Methods in Applied Sciences*, 17(12):2055–2113, 12 2007.

[27] A. Cauchy. Methode generale pour la resolution des systemes d'equations simultanees. *C.R. Acad. Sci. Paris*, 25:536–538, 1847.

[28] P. Chandrashekar and C. Klingenberg. A second order well-balanced finite volume scheme for Euler equations with gravity. *SIAM Journal on Scientific Computing*, 37:382–402, 2015.

[29] G. Chen and S. Noelle. A new hydrostatic reconstruction scheme motivated by the wet-dry front. *SIAM Journal on Numerical Analysis*, 55:758–784, 2017.

[30] Y. Cheng, A. Chertock, M. Herty, A. Kurganov, and T. Wu. A new approach for designing moving-water equilibria preserving schemes for the shallow water equations. *Journal of Scientific Computing*, 80:538–554, 2019.

[31] Y. Cheng and A. Kurganov. Moving-water equilibria preserving central-upwind schemes for the shallow water equations. *Communications in Mathematical Sciences*, 14(6):1643–1663, 2016.

[32] A. Chertock, A. Kurganov, and Y. Liu. Central-upwind schemes for the system of shallow water equations with horizontal temperature gradients. *Numerical Mathematics*, 127:595–639, 2014.

[33] B. Cockburn, G. E. Karniadakis, and C. W. Shu. The Development of Discontinuous Galerkin Methods. In *Discontinuous Galerkin Methods*, pages 3–50. Springer, Berlin, Heidelberg, 2000.

[34] B. Cockburn and C.-W. Shu. The Runge-Kutta local projection P1-discontinuous-Galerkin finite element method for scalar conservation laws. In *1st National Fluid Dynamics Conference*, Reston, Virigina, 7 1988. American Institute of Aeronautics and Astronautics.

[35] B. Cockburn and C.-W. Shu. TVB Runge-Kutta Local Projection Discontinuous Galerkin Finite Element Method for Conservation Laws II: General Framework. *Mathematics of Computation*, 52(186):411, 4 1989.

[36] B. Cockburn and C. W. Shu. The Runge-Kutta Discontinuous Galerkin Method for Conservation Laws V: Multidimensional Systems. *Journal of Computational Physics*, 141(2):199–224, 4 1998.

[37] S. S. Collis, K. Ghayour, M. Heinkenschloss, M. Ulbrich, and S. Ulbrich. Numerical Solution of Optimal Control Problems Governed by the Compressible Navier-Stokes Equations. In *Optimal Control of Complex Structures*, pages 43–55. Birkhäuser Basel, 2001.

[38] J. Darbon. On convex finite-dimensional variational methods in imaging sciences and Hamilton–Jacobi equations. *SIAM Journal on Imaging Sciences*, 8(4):2268–2293, 10 2015.

[39] J. Darbon and S. Osher. Algorithms for overcoming the curse of dimensionality for certain Hamilton–Jacobi equations arising in control theory and elsewhere. *Research in Mathematical Sciences*, 3(1):1–26, 12 2016.

[40] D. Davis and W. Yin. A Three-Operator Splitting Scheme and its Optimization Applications. *Set-Valued and Variational Analysis*, 25(4):829–858, 12 2017.

[41] O. Delestre and P.-Y. Lagrée. A 'well-balanced' finite volume scheme for blood flow simulation. *International Journal for Numerical Methods in Fluids*, 72(2):177–205, 2013.

[42] M. Delfour and F. Trochu. Discontinuous finite element methods for the approximation of optimal control problems governed by hereditary differential systems. *Distributed Parameter Systems: Modelling and Identification*, pages 256–271, 1978.

[43] P. Dellar. Common hamiltonian structure of the shallow water equations with horizontal temperature gradients and magnetic fields. *Physics of Fluids*, 15:292–297, 2003.

[44] V. Desveaux, M. Zenk, C. Berthon, and C. Klingenberg. Well-balanced schemes to capture non-explicit steady states: Ripa model. *Math. of Comput.*, 85:1571–1602, 2016.

[45] J. C. Dietrich, S. Tanaka, J. J. Westerink, C. N. Dawson, R. A. Luettich, M. Zijlema, L. H. Holthuijsen, J. M. Smith, L. G. Westerink, and H. J. Westerink. Performance of the unstructured-mesh, SWAN+ ADCIRC model in computing hurricane waves and surge. *Journal of Scientific Computing*, 52(2):468–497, 8 2012.

[46] P. R. Eiseman and A. P. Stone. Conservation Laws of Fluid Dynamics–A Survey. *SIAM Review*, 22(1):12–27, 1 1980.

[47] A. Ern, S. Piperno, and K. Djadel. A well-balanced Runge-Kutta discontinuous Galerkin method for the shallow-water equations with flooding and drying. *International Journal for Numerical Methods in Fluids*, 58(1):1–25, 9 2008.

[48] L. Euler. Principia pro motu sanguinis per arterias determinando. *Opera Postuma*, 2:814–823, 1862.

[49] M. Ferlauto. A pseudo-compressibility method for solving inverse problems based on the 3D incompressible Euler equations. *Inverse Problems in Science and Engineering*, 23(5):798–817, 7 2015.

[50] S. Fischer and I. Steinwart. Sobolev Norm Learning Rates for Regularized Least-Squares Algorithm. *arXiv: Machine Learning*, 2 2017.

[51] U. S. Fjordholm, S. Mishra, and E. Tadmor. Well-balanced and energy stable schemes for the shallow water equations with discontinuous topography. *Journal of Computational Physics*, 230(14):5587–5609, 6 2011.

[52] L. Formaggia, J.-F. Gerbeau, F. Nobile, and A. Quarteroni. Numerical treatment of defective boundary conditions for the Navier–Stokes equations. *SIAM Journal on Numerical Analysis*, 40(1):376–401, 2002.

[53] L. Formaggia, J.F. Gerbeau, F. Nobile, and A. Quarteroni. On the coupling of 3D and 1D Navier-Stokes equations for flow problems in compliant vessels. *Computer Methods in Applied Mechanics and Engineering*, 191(6):561–582, 2001.

[54] L. Formaggia, F. Nobile, and A. Quarteroni. A one dimensional model for blood flow: Application to vascular prosthesis. In Ivo Babuška, Philippe G. Ciarlet, and Tetsuhiko Miyoshi, editors, *Mathematical Modeling and Numerical Simulation in Continuum Mechanics*, pages 137–153, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.

[55] L. Formaggia, A. Quarteroni, and A. Veneziani. *Cardiovascular Mathematics: modeling and simulation of the circulatory system.* Springer-Verlag Italia, Milano, 2009.

[56] J. M. Gallardo, C. Parés, and M. Castro. On a well-balanced high-order finite volume scheme for shallow water equations with topography and dry areas. *Journal of Computational Physics*, 227(1):574–601, 11 2007.

[57] T. Gallouët, J. M. Hérard, and N. Seguin. Some approximate Godunov schemes to compute shallow-water equations with topography. *Computers and Fluids*, 32(4):479–513, 5 2003.

[58] M. Garavello and B. Piccoli. *Traffic Flow on Networks: Conservation Laws Model.* American Institute of Mathematical Sciences, 2006.

[59] A. F. Gessese, M. Sellier, E. Van Houten, and G. Smart. Reconstruction of river bed topography from free surface data using a direct numerical approach in one-dimensional shallow water flow. *Inverse Problems*, 27(2):025001, 2 2011.

[60] A.R. Ghigo, O. Delestre, J.-M. Fullana, and P.-Y. Lagrée. Low-shapiro hydrostatic reconstruction technique for blood flow simulation in large arteries with varying geometrical and mechanical properties. *Journal of Computational Physics*, 331:108–136, 2017.

[61] M. Giles and S. Ulbrich. Convergence of linearized and adjoint approximations for discontinuous solutions of conservation laws. Part 2: Adjoint approximations and extensions. *SIAM Journal on Numerical Analysis*, 48(3):905–921, 6 2010.

[62] O. Glass and S. Guerrero. On the uniform controllability of the burgers equation. *SIAM Journal on Control and Optimization*, 46(4):1211–1238, 9 2007.

[63] S. K. Godunov. A difference method for numerical calculation of discontinuous equations of hydrodynamics. *Mat. Sb. (N.S.)*, 47 (89)(3):271–306, 1959.

[64] G. Gomes and R. Horowitz. Optimal freeway ramp metering using the asymmetric cell transmission model. *Transportation Research Part C: Emerging Technologies*, 14(4):244–262, 8 2006.

[65] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016.

[66] S. Gottlieb, C. W. Shu, and E. Tadmor. Strong stability-preserving high-order time discretization methods. *SIAM Review*, 43(1):89–112, 3 2001.

[67] U. Gruber and P. Bartelt. Snow avalanche hazard modelling of large areas using shallow water numerical methods and GIS. *Environmental Modelling and Software*, 22(10):1472–1481, 10 2007.

[68] M. Gugat, M. Dick, and G. Leugering. Gas flow in fan-shaped networks: Classical solutions and feedback stabilization. *SIAM Journal on Control and Optimization*, 49(5):2101–2117, 2011.

[69] X. Han and G. Li. Well-balanced finite difference WENO schemes for the Ripa model. *Computers & Fluids*, 134-135:1–10, 2016.

[70] C. Heining and N. Aksel. Bottom reconstruction in thin-film flow over topography: Steady solution and linear stability. *Physics of Fluids*, 21(8), 2009.

[71] M. Heinkenschloss. Formulation and Analysis of a Sequential Quadratic Programming Method for the Optimal Dirichlet Boundary Control of Navier-Stokes Flow. In *Optimal Control: Theory, Algorithms, and Applications*, pages 178–203. Springer, Boston, MA, 1998.

[72] P. Heinrich. Nonlinear Water Waves Generated by Submarine and Aerial Landslides. *Journal of Waterway, Port, Coastal, and Ocean Engineering*, 118(3):249–266, 5 1992.

[73] G. Hernandez-Duenas. A hybrid method to solve shallow water flows with horizontal density gradients. *Journal of Scientific Computing*, 73:753–782, 2017.

[74] M. Hintermüller. Semismooth Newton Methods and Applications. *Department of Mathematics, Humboldt-University of Berlin*, 2010.

[75] H. Holden, F. S. Priuli, and N. H. Risebro. On an inverse problem for scalar conservation laws. *Inverse Problems*, 30(3):035015, 2 2014.

[76] M. Honnorat, J. Marin, J. Monnier, and X. Lai. Dassflow v1.0: a variational data assimilation software for 2D river flows. Technical report, 2007.

[77] P. Hu, Z. Cao, G. Pender, and G. Tan. Numerical modelling of turbidity currents in the Xiaolangdi reservoir, Yellow River, China. *Journal of Hydrology*, 464-465:41–53, 9 2012.

[78] D. Jacquet, C. C. De Wit, and D. Koenig. Traffic control and monitoring with a macroscopic model in the presence of strong congestion waves. In *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference, CDC-ECC '05*, volume 2005, pages 2164–2169, 2005.

[79] D. Jacquet, M. Krstic, and C. C. De Wit. Optimal control of scalar one-dimensional conservation laws. In *Proceedings of the American Control Conference*, volume 2006, pages 5213–5218, 2006.

[80] A. Jameson. Aerodynamic design via control theory. *Journal of Scientific Computing*, 3(3):233–260, 9 1988.

[81] A. Jameson. Optimum aerodynamic design using CFD and control theory. In *12th Computational Fluid Dynamics Conference*, pages 926–949. American Institute of Aeronautics and Astronautics Inc, AIAA, 1995.

[82] S. Jin. A Steady-State Capturing Method for Hyperbolic Systems with Geometrical Source Terms. In *Transport in Transition Regimes*, pages 177–183. Springer, New York, NY, 2004.

[83] H. Kang and K. Tanuma. Inverse problems for scalar conservation laws. *Inverse Problems*, 21(3):1047, 4 2005.

[84] S. Kindermann, S. Osher, and P. W. Jones. Deblurring and denoising of images by nonlocal functionals. *Multiscale Modeling and Simulation*, 4(4):1091–1115, 2005.

[85] A. Kurganov. Finite-volume schemes for shallow-water equations, 5 2018.

[86] A. Kurganov and D. Levy. Central-upwind schemes for the Saint-Venant system. *Mathematical Modelling and Numerical Analysis*, 36(3):397–425, 2002.

[87] A. Kurganov and G. Petrova. A second-order well-balanced positivity preserving central-upwind scheme for the Saint-Venant system. *Communications in Mathematical Sciences*, 5(1):133–160, 2007.

[88] J. Larson, M. Menickelly, and S. M. Wild. Derivative-free optimization methods. *Acta Numerica*, 28:287–404, 4 2019.

[89] J. M. Lellouche, J. L. Devenon, and I. Dekeyser. Boundary control of Burgers' equation-a numerical approach. *Computers and Mathematics with Applications*, 28(5):33–44, 1994.

[90] P. Lesaint and P. A. Raviart. On a Finite Element Method for Solving the Neutron Transport Equation. *Publications mathématiques et informatique de Rennes*, (S4):1–40, 1974.

[91] R. J. LeVeque. Balancing Source Terms and Flux Gradients in High-Resolution Godunov Methods: The Quasi-Steady Wave-Propagation Algorithm. *Journal of Computational Physics*, 146(1):346–365, 10 1998.

[92] R. J. LeVeque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press, 8 2002.

[93] R.J. LeVeque. *Numerical Methods for Conservation Laws*. Birkhäuser Basel, Basel, 1992.

[94] G. Li, O. Delestre, and L. Yuan. Well-balanced discontinuous galerkin method and finite volume weno scheme based on hydrostatic reconstruction for blood flow model in arteries. *International Journal for Numerical Methods in Fluids*, 86(7):491–508, 2018.

[95] G. Li and Y. Xing. Well-balanced discontinuous Galerkin methods with hydrostatic reconstruction for the euler equations with gravitation. *Journal of Computational Physics*, 352:445–462, 2018.

[96] J. Li and G. Chen. The generalized Riemann problem method for the shallow water equations with bottom topography. *International Journal for Numerical Methods in Engineering*, 65(6):834–862, 2 2006.

[97] S. Li and C. J. Duffy. Fully coupled approach to modeling shallow water flow, sediment transport, and bed evolution in rivers. *Water Resources Research*, 47(3), 3 2011.

[98] M. J. Lighthill. *Waves in fluids*. Cambridge University Press Cambridge [Eng.]; New York, 1978.

[99] J.-L. Lions. *Optimal control of systems governed by partial differential equations*. Springer-Verlag, 1971.

[100] X. Liu, J. Albright, Y. Epshteyn, and A. Kurganov. Well-balanced positivity preserving central-upwind scheme with a novel wet/dry reconstruction on triangular grids for the Saint-Venant system. *Journal of Computational Physics*, 374:213–236, 12 2018.

[101] M. Lukáčová-Medvidóvá, S. Noelle, and M. Kraft. Well-balanced finite volume evolution Galerkin methods for the shallow water equations. *Journal of Computational Physics*, 221(1):122–147, 1 2007.

[102] J. Monnier, F. Couderc, D. Dartus, K. Larnier, R. Madec, and J. P. Vila. Inverse algorithms for 2D shallow water equations in presence of wet dry fronts: Application to flood plain dynamics. *Advances in Water Resources*, 97:11–24, 11 2016.

[103] G.I. Montecinos, J. Lopez-Rios, J.H. Ortega, and R. Lecaros. A numerical procedure and unified formulation for the adjoint approach in hyperbolic PDE-constrained optimal control problems. *IMA Journal of Applied Mathematics*, pages 1–29, 2005.

[104] J. Murillo and P. García-Navarro. A Roe type energy balanced solver for 1D arterial blood flow and transport. *Computers and Fluids*, 117:149–167, 2015.

[105] L.O. Müller, C. Parés, and E.F. Toro. Well-balanced high-order numerical schemes for one-dimensional blood flow in vessels with varying mechanical properties. *Journal of Computational Physics*, 242:53 – 85, 2013.

[106] L.O. Müller and E. F. Toro. Well-balanced high-order solver for blood flow in networks of vessels with variable properties. *International Journal for Numerical Methods in Biomedical Engineering*, 29(12):1388–1411, 2013.

[107] H. Nessyahu and E. Tadmor. Non-oscillatory central differencing for hyperbolic conservation laws. *Journal of Computational Physics*, 87(2):408–463, 4 1990.

[108] M. K. Ng, R. H. Chan, T. F. Chan, and A. M. Yip. Cosine transform preconditioners for high resolution image reconstruction. *Linear Algebra and Its Applications*, 316(1-3):89–104, 9 2000.

[109] S. Noelle, N. Pankratz, G. Puppo, and J. R. Natvig. Well-balanced finite volume schemes of arbitrary order of accuracy for shallow water flows. *Journal of Computational Physics*, 213(2):474–499, 4 2006.

[110] S. Noelle, Y. Xing, and C. W. Shu. High-order well-balanced finite volume WENO schemes for shallow water equation with moving water. *Journal of Computational Physics*, 226(1):29–58, 9 2007.

[111] M. S. Olufsen, C. S. Peskin, W. Y. Kim, E. M. Pedersen, A. Nadim, and J. Larsen. Numerical simulation and experimental validation of blood flow in arteries with structured-tree outflow conditions. *Annals of Biomedical Engineering*, 28(11):1281–1299, 2000.

[112] I. Özgen, J. Zhao, D. Liang, and R. Hinkelmann. Urban flood modeling using shallow water equations with depth-dependent anisotropic porosity. *Journal of Hydrology*, 541:1165–1184, 10 2016.

[113] N. Parikh. Proximal Algorithms. *Foundations and Trends in Optimization*, 1(3):127–239, 2014.

[114] T. J. Pedley. *The Fluid Mechanics of Large Blood Vessels*. Cambridge Monographs on Mechanics. Cambridge University Press, 1980.

[115] B. Perthame and C. W. Shu. On positivity preserving finite volume schemes for Euler equations. *Numerische Mathematik*, 73(1):119–130, 1996.

[116] B. Perthame and C. Simeoni. A kinetic scheme for the Saint-Venant system with a source term. *Calcolo*, 38(4):201–231, 2001.

[117] O. Pironneau. On optimum design in fluid mechanics. *Journal of Fluid Mechanics*, 64(1):97–110, 1974.

[118] G. Pontrelli. A mathematical model of flow in a liquid-filled visco-elastic tube. *Medical & biological engineering & computing*, 40:550–6, 2002.

[119] G. Pontrelli. Nonlinear pulse propagation in blood flow problems. In Angelo Marcello Anile, Vincenzo Capasso, and Antonio Greco, editors, *Progress in Industrial Mathematics at ECMI 2000*, pages 630–635, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.

[120] J. Qiu, B.C. Khoo, and C.-W. Shu. A numerical study for the performance of the Runge-Kutta discontinuous Galerkin method based on different numerical fluxes. *Journal of Computational Physics*, 212(2):540 – 565, 2006.

[121] J. Qiu and C. W. Shu. Runge-Kutta discontinuous Galerkin method using WENO limiters. *SIAM Journal on Scientific Computing*, 26(3):907–929, 7 2005.

[122] A. Quarteroni, M. Tuveri, and A. Veneziani. Computational vascular fluid dynamics: problems, models and methods. *Computing and Visualization in Science*, 2(4):163–197, 2000.

[123] R. L. Raffard, K. Amonlirdviman, J. D. Axelrod, and C. J. Tomlin. An adjoint-based parameter identification algorithm applied to planar cell polarity signaling. *IEEE Transactions on Automatic Control*, 53(SPECIAL ISSUE):109–121, 2008.

[124] W. H. Reed and T. R. Hill. Triangular mesh methods for the neutron transport equation, 1973.

[125] J. Reilly, S. Samaranayake, M. L. Delle Monache, W. Krichene, P. Goatin, and A. M. Bayen. Adjoint-Based Optimization on a Network of Discretized Scalar Conservation Laws with Applications to Coordinated Ramp Metering. *Journal of Optimization Theory and Applications*, 167(2):733–760, 11 2015.

[126] M. Ricchiuto. An explicit residual based approach for shallow water flows. *Journal of Computational Physics*, 280:306–344, 1 2015.

[127] M. Ricchiuto and A. Bollermann. Stabilized residual distribution for shallow water simulations. *Journal of Computational Physics*, 228(4):1071–1115, 3 2009.

[128] P. Ripa. Conservation laws for primitive equations models with inhomogeneous layers. *Geophysical and Astrophysical Fluid Dynamics*, 70:85–111, 1993.

[129] P. Ripa. On improving a one-layer ocean model with thermodynamics. *Journal Fluid Mechanics*, 303:16–201, 1995.

[130] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1-4):259–268, 11 1992.

[131] G. Russo. Central schemes for conservation laws with application to shallow water equations. In *Trends and Applications of Mathematics to Mechanics*, pages 225–246. Springer Milan, 9 2006.

[132] G. Russo and A. Khe. High order well balanced schemes for systems of balance laws. In *Hyperbolic problems: theory, numerics and applications*, pages 919–928. Proceedings of Symposia in Applied Mathematics 67, Part 2, American Mathematics Society, Providence, RI, 2009.

[133] G. Russo and A. Khe. High order well-balanced schemes based on numerical reconstruction of the equilibrium variables. pages 230–241. World Scientific Pub Co Pte Lt, 4 2010.

[134] C. Sanchez-Linares, T. Morales de Luna, and M. J. Castro. A HLLC scheme for Ripa model. *Applied Mathematics and Computation*, 272:369–384, 2016.

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

[136] S.J. Sherwin, V. Franke, J. Peiró, and K. Parker. One-dimensional modelling of a vascular network in space-time variables. *Journal of Engineering Mathematics*, 47(3):217–250, 2003.

[137] Y. Shi, P. V. Lawford, and R. Hose. Review of Zero-D and 1-D models of blood flow in the cardiovascular system. *BioMedical Engineering OnLine*, 10(1):33, 2011.

[138] C. W. Shu. TVB Uniformly High-Order Schemes for Conservation Laws. *Mathematics of Computation*, 49(179):105, 7 1987.

[139] J. F.A. Sleath. Sediment transport by waves and currents. *Journal of Geophysical Research*, 100(C6):10977–10986, 6 1995.

[140] L. Song, J. Zhou, J. Guo, Q. Zou, and Y. Liu. A robust well-balanced finite volume model for shallow water flows with wetting and drying over irregular terrain. *Advances in Water Resources*, 34(7):915–932, 7 2011.

[141] W. Sun and Y.-X. Yuan. *Optimization Theory and Methods: Nonlinear Programming*. Springer Science+Business Media, LLC, 2006.

[142] H. Tang, T. Tang, and K. Xu. A gas-kinetic scheme for shallow-water equations with source terms. *Zeitschrift fur Angewandte Mathematik und Physik*, 55(3):365–382, 2004.

[143] R. Touma and C. Klingenberg. Well-balanced central finite volume methods for the Ripa system. *Applied Numerical Mathematics*, 97:42–68, 2015.

[144] R. Vazquez, J. M. Coron, M. Krstic, and G. Bastin. Local exponential H^2 stabilization of a 2×2 quasilinear hyperbolic system using backstepping. In *Proceedings of the IEEE Conference on Decision and Control*, pages 1329–1334, 2011.

[145] M.E. Vazquez-Cendon. Improved treatment of source terms in upwind schemes for the shallow water equations in channels with irregular geometry. *Journal Computational Physics*, 148:497–526, 1999.

[146] Z. Wang, G. Li, and O. Delestre. Well-balanced finite difference weighted essentially non-oscillatory schemes for the blood flow model. *International Journal for Numerical Methods in Fluids*, 82(9):607–622, 2016.

[147] L. C. Wellford and J. T. Oden. Discontinuous finite-element approximations for the analysis of shock waves in nonlinearly elastic materials. *Journal of Computational Physics*, 19(2):179–210, 10 1975.

[148] A. Wirgin. The inverse crime. 1 2004.

[149] N. Xiao, J. Alastruey, and C. Alberto Figueroa. A systematic comparison between 1-d and 3-d hemodynamics in compliant arterial models. *International Journal for Numerical Methods in Biomedical Engineering*, 30(2):204–231, 2014.

[150] Y. Xing. Exactly well-balanced discontinuous Galerkin methods for the shallow water equations with moving water equilibrium. *Journal of Computational Physics*, 257:536–553, 2014.

[151] Y. Xing. Numerical Methods for the Nonlinear Shallow Water Equations. In *Handbook of Numerical Analysis*, volume 18, pages 361–384. Elsevier B.V., 2017.

[152] Y. Xing and C.-W. Shu. High order finite difference WENO schemes with the exact conservation property for the shallow water equations. *Journal of Computational Physics*, 208:206–227, 2005.

[153] Y. Xing and C.-W. Shu. A new approach of high order well-balanced finite volume WENO schemes and discontinuous Galerkin methods for a class of hyperbolic systems with source terms. *Communications in Computational Physics*, 1:100–134, 2006.

[154] Y. Xing and C-W. Shu. High order well-balanced finite volume WENO schemes and discontinuous Galerkin methods for a class of hyperbolic systems with source terms. *Journal of Computational Physics*, 214:567–598, 2006.

[155] Y. Xing and C.-W. Shu. A new approach of high order well-balanced finite volume WENO schemes and discontinuous Galerkin methods for a class of hyperbolic systems with source terms. *Commun. Comput. Phys.*, 1:100–134, 2006.

[156] Y. Xing and C.-W. Shu. High order well-balanced WENO scheme for the gas dynamics equations under gravitational fields. *Journal of Scientific Computing*, 54:645–662, 2013.

[157] Y. Xing and C. W. Shu. High order well-balanced WENO scheme for the gas dynamics equations under gravitational fields. *Journal of Scientific Computing*, 54(2-3):645–662, 2 2013.

[158] Y. Xing and C.-W. Shu. A survey of high order schemes for the shallow water equations. *Journal of Mathematical Study*, 47:221–249, 2014.

[159] Y. Xing, C.-W. Shu, and S. Noelle. On the advantage of well-balanced schemes for moving-water equilibria of the shallow water equations. *Journal of Scientific Computing*, 48:339–349, 2011.

[160] Y. Xing, X. Zhang, and C-W. Shu. Positivity-preserving high order well-balanced discontinuous Galerkin methods for the shallow water equations. *Advances in Water Resources*, 33:1476–1493, 2010.

[161] W. Yin, S. Osher, D. Goldfarb, and J. Darbon. Bregman iterative algorithms for ℓ_1-minimization with applications to compressed sensing. *SIAM Journal on Imaging Sciences*, 1(1):143–168, 2008.

[162] T. Young. Hydraulic investigations, subservient to an intended croonian lecture on the motion of the blood. *Philosophical Transactions of the Royal Society of London*, 98:164–186, 1808.

[163] T. Young. XIII. Hydraulic investigations, subservient to an intended Croonian Lecture on the motion of the blood. *Philosophical Transactions of the Royal Society of London*, 98:164–186, 12 1808.

[164] X. Zhang and C. W. Shu. On maximum-principle-satisfying high order schemes for scalar conservation laws. *Journal of Computational Physics*, 229(9):3091–3120, 5 2010.

[165] X. Zhang and C. W. Shu. On positivity-preserving high order discontinuous Galerkin schemes for compressible Euler equations on rectangular meshes. *Journal of Computational Physics*, 229(23):8918–8934, 11 2010.

[166] A. K. Ziliaskopoulos. Linear programming model for the single destination System Optimum Dynamic Traffic Assignment problem. *Transportation Science*, 34(1):37–49, 2 2000.