Doctoral Program
Computational Mathematics
Numerical Analysis and Symbolic Computation

On the Parameter Estimation Problem of Magnetic Resonance Advection Imaging

Simon Hubmer   Andreas Neubauer   Ronny Ramlau

Henning U. Voss

DK-Report No. 2016-05   12 2016

A–4040 LINZ, ALTENBERGERSTRASSE 69, AUSTRIA

On the Parameter Estimation Problem of Magnetic Resonance Advection Imaging

Simon Hubmer[*], Andreas Neubauer[†], Ronny Ramlau[‡§], Henning U. Voss[¶]

Abstract

We present a reconstruction method for estimating the pulse-wave velocity in the brain from dynamic MRI data. The method is based on solving an inverse problem involving an advection equation. A space-time discretization is used and the resulting largescale inverse problem is solved using an accelerated Landweber type gradient method incorporating sparsity constraints and utilizing a wavelet embedding. Numerical example problems and a real-world data test show a significant improvement over the results obtained by the previously used method.

Keywords. Brain Imaging, MRI, Cerebral Hemodynamics, MR Angiography, Cerebrovascular Disease, Mathematical Modelling, Space-Time Discretization, Regularization, Inverse Problems, Sparsity, Landweber Iteration, Wavelets, Nesterov Acceleration Strategy, Parameter Identification

1 Introduction

Magnetic Resonance Advection Imaging (MRAI) is a recently developed method to map the pulsatory signal component of dynamic echo planar imaging (EPI) data of the brain, such as acquired in functional and resting state functional Magnetic Resonance Imaging (MRI) experiments [34]. Its underlying mathematical model is specifically based on the wave equation for unidirectional waves, and as such is an advection equation. MRAI derives its information from the local spatiotemporal properties of the dynamic MRI data sets [35] and does not require an external pulse reference signal. It has been shown that MRAI maps depict the location of major arteries as well as some venous structure.

[*]Johannes Kepler University Linz, Doctoral Program Computational Mathematics, Altenbergerstraße 69, A-4040 Linz, Austria (simon.hubmer@dk-compmath.jku.at), Corresponding author

[†]Johannes Kepler University Linz, Institute of Industrial Mathematics, Altenbergerstraße 69, A-4040 Linz, Austria (neubauer@indmath.uni-linz.ac.at)

[‡]Johannes Kepler University Linz, Institute of Industrial Mathematics, Altenbergerstraße 69, A-4040 Linz, Austria (ronny.ramlau@jku.at)

[§]Johann Radon Institute Linz, Altenbergerstraße 69, A-4040 Linz, Austria (ronny.ramlau@ricam.oeaw.ac.at)

[¶]Weill Cornell Medical College, Department of Radiology, 516 E 72nd Street New York, NY 10021 (hev2006@med.cornell.edu)

In addition, colour direction maps allow for visualization of the orientation direction of blood vessels.

It has been suggested that MRAI may potentially serve as a biomarker for the health of the cerebrovascular system. The reason is that MRAI is designed to reflect the spatiotemporal properties of travelling waves, and pulse wave velocities (PWV) are a main indicator for the physical properties of blood vessels. By means of the well-known Moens-Korteweg equation, PWVs are related to vessel diameter, wall thickness, and wall stiffness. In particular wall thickness and stiffness are key parameters that change in vascular disease and with age.

However, although MRAI is designed to reflect the spatiotemporal properties of travelling waves, it has been found that PWVs are difficult to measure with the previously proposed modelling approach [34], which is based on local multiple regression of finite difference estimators of the differential operators in the unidirectional wave equation. It would be desirable to improve estimation techniques in order to overcome these limitations. It would also be desirable to have an optimized method to be able to estimate requirements on the data in order to obtain meaningful, i.e., quantitative values for the pulse wave velocities or at least representative values that are affected by differing pulse wave velocities throughout the cerebrovascular system.

In [34], a multiple regression approach was used in order to estimate the PWV. In this contribution, we consider the problem from the viewpoint of parameter estimation in PDEs. More precisely, we will define an operator F which maps a velocity v onto data y via

$$F(v) = y \,,$$

and then solve this equation for v using standard techniques in inverse problem theory, as described for example in [10]. The underlying equation connecting the velocity vector field $v(x, y, z)$ to the dynamic MRI signal $\rho(x, y, z, t)$ will be the advection equation

$$\frac{\partial}{\partial t}\rho(x, y, z, t) + v(x, y, z) \cdot \nabla \rho(x, y, z, t) = 0 \,. \tag{1.1}$$

The subsequent paper is structured as follows. In the next section, we will give some medical background, introduce the reader to dynamic MRI and give some specifics of our problem. In Section 3 we will define the mathematical model used throughout this paper and in Section 4 we will discretize the model and introduce an inverse problem for the PWV. Section 5 is concerned with the solution of the inverse problem and in Section 6 several numerical experiments with simulated and real-world data are presented. Finally, Section 7 gives some conclusions and an outlook.

2   Background

2.1   Pulse wave velocity

Cardiovascular pulse waves provide a natural physical perturbation to vascular dynamics, and their effects have been utilized in clinical diagnostics for a long time. For

example, the PWV in major arteries can be measured directly and contains information about arterial compliance [18], defined as the ratio of blood volume change to blood pressure change. Arterial compliance is an important determinant of the state of the cerebrovascular system. With respect to the brain, aortic stiffness has been associated with cerebral small-vessel disease in hypertensive patients [14] and cognitive decline [26]. In addition, emerging concepts such as pulse wave encephalopathy would profit from diagnostic imaging methods of cerebral vasculature [2, 13].

The PWV in arteries, or Moens-Korteweg velocity [19], follows from the Moens-Korteweg formula [17, 23],

$$v = \sqrt{\frac{Eh}{\rho_B d}}. \tag{2.1}$$

The PWV depends on the three parameters vascular diameter d, wall thickness h, and the vessel wall's Young's modulus or distensibility, E, if the reasonable approximation of constant density of the blood, ρ_B, is made. Equation 2.1 models vessels as elastic tubes with isotropic walls [40]. Pressure gradients, which would be required to determine blood flow, do not appear in the Moens-Korteweg formula. In other words, PWVs can be modelled independently from blood flow velocity and in fact can be one to two orders of magnitude faster [19].

For a blood vessel along the direction of v, Equation 1.1 describes the pulsatile component of the blood flow velocity along that direction, with v being the PWV. Importantly, the same advection equation would also hold for blood pressure waves [8], which, via the Windkessel model [29], are a function of the integrated net flow into the vessel reservoir [1]. Since the local blood volume is related to local blood pressure via the compliance C as $dV = C\,dp$, the same advection equation applies to the pulsatile component of the blood volume as well. For the rest of this paper, $\rho(x, y, z, t)$ should be understood as the MRI signal variability attributable to volume change, and we also interpret the three-dimensional velocity vector v as a PWV.

Though $\rho(x, y, z, t)$ depends on the blood flow velocity via the relationship between pulsatile flow, pressure, and volume, the constant component of the flow, or the average blood flow, decouples from Equation 1.1, as it does not cause any spatial or temporal signal change. It could cause signal variability, however, if blood is not assumed to have homogeneous properties in the model domain Ω, which may be caused by variations in oxygenation or temperature, or any other property that might affect the particular MRI contrast. Here we assume that the MRI signal is not affected by any of these properties.

2.2   EPI data

There are only few in-vivo options for imaging vascular dynamics in the human brain. Arterial spin labelling (ASL) is the most advanced method and has high spatial resolution [37]. It provides quantitative blood flow values in the capillary bed, assuming steady flow. However, arterial compliance depends on the pulsatile component of the

Figure 2.1: Example image of a clinical MRI scanner.

flow. It can be imaged with specific ASL pulse sequences [36,38]. Pulsatile flow components can also be imaged over the whole brain with 4D phase contrast angiography [11].

In this contribution we are aiming at deriving PWV related quantities from echo-planar imaging [31] (EPI) data. EPI is the method of choice for functional MRI [25]. Here we are less interested in the functional aspects of EPI, but in the fact that EPI can also yield fast dynamic data over the whole brain. As all MRI imaging signal intensity is proportional to the total amount of resonant spins within a voxel, it reflects the local proton distribution, which shows pulsatile information around vessels. This has been demonstrated with phase coherence maps before [33,34], as well as with statistical parametric mapping [4,32].

In general, the MR signal $S(\mathbf{k})$ in physical $\mathbf{k}$ space is proportional to the inverse Fourier transform of the spin density in a voxel at position $\mathbf{r}$, $\rho(\mathbf{r})$, i.e.,

$$S(\mathbf{k}) \propto \int_{-\infty}^{+\infty} \rho(\mathbf{r}) e^{2\pi i \mathbf{k} \cdot \mathbf{r}} d\mathbf{r} \,, \tag{2.2}$$

where $\mathbf{k}$ is the $\mathbf{k}$-vector. The image $S(\mathbf{r})$ is the magnitude of the Fourier inversion of this expression. The vectors $\mathbf{r}$ and $\mathbf{k}$ are, depending on the acquisition scheme, two or three-dimensional. (See, for example [21].) In EPI, images are acquired very rapidly to allow for whole brain coverage within seconds. A typical EPI data set consists of three-dimensional volumes acquired repeatedly in time with a repetition time TR. Each volume contains either the whole brain or always the same part of it. Volumes are acquired slice-by-slice, i.e., the vector $\mathbf{r}$ is two-dimensional. The typical spatial resolution or voxel volume depends on the field-of-view, in-plane or slice matrix size, and slice thickness [7]. The field of view typically is 20 to 24 cm in order to include the whole head in axial slices. Slices are either acquired sequentially or interleaved. In sequential acquisition, first slice 1, then slice 2, etc., are acquired, where slice 1 is adjacent to slice 2 and so on. In interleaved acquisition, for example first all odd and then all even slices are acquired. Therefore, care has to be taken to assign the correct acquisition

time to each slice, and the model that we are proposing can take slice acquisition order into account. It is described for sequential, specifically, ascending slices, but can easily be adapted to other acquisition schemes. It might be worth mentioning that we are not pursuing a "slice-time correction" [30], which is often performed in the analysis of functional MRI data and consists of an interpolation of intensity values to an evenly sampled time grid. Such a procedure is always of approximative nature and would not sufficiently take into account the fast dynamics of for example travelling pulse waves.

3   Mathematical Model

Our goal is to estimate velocities of travelling waves in blood vessels from spatiotemporal MRI data. As a first approximation, we will neglect any frequency-dependence of the velocity, or dispersion, as well as reflected pulses travelling against the main blood flow direction. The latter assumption means that the back flow amplitude is considered to be much smaller than the forward flow amplitude.

Under those assumptions, the authors of [34]considered the following local model, defined on small subdomains Ω_S of the model domain Ω, e.g. $3 \times 3 \times 3$ voxels in size. On each subdomain, the dynamic MRI signal $\rho(x, y, z, t)$ is assumed to fulfill the advection equation

$$\frac{\partial}{\partial t}\rho(x, y, z, t) + \bar{v} \cdot \nabla \rho(x, y, z, t) = 0 \,, \tag{3.1}$$

where ∇ is the gradient with respect to the space variables (x, y, z) and $\bar{v} = \bar{v}(\Omega_S)$ is a velocity assumed to be constant on each subdomain Ω_S. Using finite difference approximations of the derivatives of ρ, the authors of [34]used a multiple regression approach to get estimates for the local velocities $\bar{v}$. Although yielding maps of velocity estimates that reflect main cerebral arteries, those estimates were not quantitative. Furthermore, the local regression matrices used there were ill-conditioned for many of the data points and additionally, the finite difference operators in z-direction used to derive those matrices did not take into account the limited data due to the slice-time acquisition procedure and therefore lead to crude approximations of the z-derivatives.

In this paper, we use an approach which is global in nature, retains the underlying advection equation and gets rid of the numerical instabilities of the regression approach. Following the physical arguments of [35], one can see that ρ is in essence assumed to be a conserved quantity, for which there holds the following continuity equation

$$\frac{\partial}{\partial t}\rho(x, y, z, t) + \text{div}\left(\rho(x, y, z, t)\, v(x, y, z)\right) = 0 \,, \tag{3.2}$$

where $v = v(x, y, z)$ is a constant-in-time velocity field now defined on the entire model domain Ω. Assuming v to be divergence-free, i.e., $\text{div}\, v = 0$, which is reasonable since we consider a basically incompressible carrier medium (blood), the product rule yields

$$\frac{\partial}{\partial t}\rho(x, y, z, t) + v(x, y, z) \cdot \nabla \rho(x, y, z, t) = 0 \,, \tag{3.3}$$

5

which is again an advection equation, now defined on the entire model domain Ω. Given measurements of $\rho(x, y, z, t)$, we want to recover the global but now space dependent velocity vector field v satisfying the above equation. This is an ill-posed problem, one reason being that derivatives of ρ are taken in (3.3), which, as the data ρ is subject to measurement errors, is an ill-posed procedure in itself.

Note that v is assumed to be independent of the time t. This assumption stems from the fact that the pulse-wave velocity is primarily dependent on time independent quantities such as vessel wall property parameters, see (2.1).

Note now that we are trying to reconstruct the vector valued quantity v from one single scalar equation. Even worse, assuming that v is a solution of (3.3), every $v + h$, where h satisfies $h \cdot \nabla \rho = 0$, is a solution as well. However, following again the physical arguments of [34], velocities h satisfying $h \cdot \nabla \rho = 0$ are of no interest to us and are in fact not detectable by our algorithm.

We could now consider the inverse problem in the continuous setting, first defining a nonlinear operator mapping between suitable function spaces, then choosing a solution method and finally discretizing. This approach turns out to be highly complicated, as the solution theory of advection equations with non-Lipschitz velocity vector fields is quite involved, see e.g., [6]. Most problematic is the fact that the Lax-Milgram framework commonly used for PDEs is no longer applicable in that case, resulting in solution concepts which are hard to handle.

Hence, we will use a first-discretize-then-regularize approach, which simplifies the subsequent computations significantly. However, we will use one fact from the classical theory, see [6], namely that v should be at least an H^1, or locally H^1 vector field.

4 Discretization

Motivated by the above considerations, we assume that as the pulse wave travels through the brain, the dynamic MRI signal $\rho = \rho(x, y, z, t)$ fulfills the advection equation, i.e.,

$$\frac{\partial}{\partial t} \rho(x, y, z, t) + \nabla \rho(x, y, z, t) \cdot v(x, y, z) = 0, \qquad (x, y, z) \in \Omega, t \in [0, T]. \qquad (4.1)$$

where $v = (v_1, v_2, v_3)$ is a velocity vector field assumed to be independent of time t and which is additionally assumed to be divergence-free, i.e., satisfies $\operatorname{div} v = 0$.

It is the aim to estimate v from measurements of ρ. We assume (for simplicity) that the brain domain Ω is a cuboid and that pointwise measurements are available at certain points. Unfortunately (see Section 2.2), the time coordinate is linked to the z-coordinate, i.e., measurements are available only at points

$$(x_i, y_j, z_k, t_{k,l}), \qquad 0 \le i \le I, 0 \le j \le J, 0 \le k \le K, 0 \le l \le L, \qquad (4.2)$$

where

$$x_i := x_0 + i\Delta x, \quad y_j := y_0 + j\Delta y, \quad z_k := z_0 + k\Delta z, \quad t_{k,l} := (k + (K+1)l)\Delta t.$$

6

This corresponds to ascending slice acquisition and means that in each time step, only one z-slice can be measured and that after a full cycle, the measurement process restarts.

In a next step, the equation (4.1) is discretized according to the data, which leads to a space-time discretization, which we will see below. The derivative with respect to t is approximated by a backwards differential quotient, the derivatives with respect to x, y, z by central quotients in the interior and by forward or backward quotients at the boundary. We then get the following discretized system of equations:

$$\frac{\rho_{i,j,k,l} - \rho_{i,j,k,l-1}}{(K+1)\Delta t} + D_{x_i}\rho_{i,j,k,l}\, v_{1,i,j,k} + D_{y_j}\rho_{i,j,k,l}\, v_{2,i,j,k} + D_{z_k}\rho_{i,j,k,l}\, v_{3,i,j,k} = 0\,, \qquad (4.3)$$

$0 \le i \le I$, $0 \le j \le J$, $0 \le k \le K$, and $1 \le l \le L$. Here

$$\rho_{i,j,k,l} = \rho(x_i, y_j, z_k, t_{k,l}) \quad \text{and} \quad v_{s,i,j,k} = v_s(x_i, y_j, z_k)\,, \ s = 1, 2, 3\,.$$

We still have to define the differential quotients $D_{x_i}, D_{y_j}, D_{z_k}$. One has to be very careful with D_{z_k}, since ρ does not exist at neighbouring z values at the same time steps. Therefore, the appropriate values are then obtained by interpolation or even extrapolation if $l = L$:

$$D_{x_i}\rho_{i,j,k,l} \ := \ \begin{cases} \dfrac{\rho_{i+1,j,k,l} - \rho_{i-1,j,k,l}}{2\Delta x}\,, & 1 \le i \le I-1 \\[2mm] \dfrac{\rho_{1,j,k,l} - \rho_{0,j,k,l}}{\Delta x}\,, & i = 0 \\[2mm] \dfrac{\rho_{I,j,k,l} - \rho_{I-1,j,k,l}}{\Delta x}\,, & i = I \end{cases} \qquad (4.4)$$

$$D_{y_j}\rho_{i,j,k,l} \ := \ \begin{cases} \dfrac{\rho_{i,j+1,k,l} - \rho_{i,j-1,k,l}}{2\Delta y}\,, & 1 \le j \le J-1 \\[2mm] \dfrac{\rho_{i,1,k,l} - \rho_{i,0,k,l}}{\Delta y}\,, & j = 0 \\[2mm] \dfrac{\rho_{i,J,k,l} - \rho_{i,J-1,k,l}}{\Delta y}\,, & j = J \end{cases} \qquad (4.5)$$

$$D_{z_k}\rho_{i,j,k,l} := \begin{cases} \dfrac{(1-r)(\rho_{i,j,k+1,l} - \rho_{i,j,k-1,l+1}) + r(\rho_{i,j,k+1,l-1} - \rho_{i,j,k-1,l})}{2\Delta z}, \\ \qquad\qquad\qquad 1 \le k \le K-1, 1 \le l < L \\[2mm] \dfrac{(1-r)\rho_{i,j,k+1,L} - (1+r)\rho_{i,j,k-1,L} + r(\rho_{i,j,k+1,L-1} + \rho_{i,j,k-1,L-1})}{2\Delta z}, \\ \qquad\qquad\qquad 1 \le k \le K-1, l = L \\[2mm] \dfrac{(1-r)\rho_{i,j,1,l} + r\rho_{i,j,1,l-1} - \rho_{i,j,0,l}}{\Delta z}, \\ \qquad\qquad\qquad k = 0, 1 \le l \le L \\[2mm] \dfrac{\rho_{i,j,K,l} - (1-r)\rho_{i,j,K-1,l+1} - r\rho_{i,j,K-1,l}}{\Delta z}, \\ \qquad\qquad\qquad k = K, 1 \le l < L \\[2mm] \dfrac{\rho_{i,j,K,L} - (1+r)\rho_{i,j,K-1,L} + r\rho_{i,j,K-1,L-1}}{\Delta z}, \\ \qquad\qquad\qquad k = K, l = L \end{cases} \tag{4.6}$$

$$r := \frac{1}{K+1}$$

We want to write the equations (4.3) in matrix-vector form. For this, we first collect all $\rho_{i,j,k,l}$ values ($l > 1$) in the vector $\vec{\rho}$ and all $\rho_{i,j,k,l}$ values ($l = 0$) in the vector $\vec{\rho}_0$, where we use the lexicographic ordering with respect to (i,j,k,l) to sort the values inside $\vec{\rho}$ and $\vec{\rho}_0$. The vector $\vec{\rho}$ then has length $m := (I+1)(J+1)(K+1)L$ and the vector $\vec{\rho}_0$ has length $n := (I+1)(J+1)(K+1)$. If we define the indices

$$\begin{aligned} \mathrm{ind}^m_{i,j,k,l} &:= i(J+1)(K+1)L + j(K+1)L + kL + l\,, \\ \mathrm{ind}^n_{i,j,k} &:= i(J+1)(K+1) + j(K+1) + k + 1\,, \end{aligned} \tag{4.7}$$

then the relationship between $\rho_{i,j,k,l}$ and $\vec{\rho}$ and $\vec{\rho}_0$ can be written precisely via

$$\begin{aligned} \rho_{i,j,k,l} &= (\vec{\rho})_{\mathrm{ind}^m_{i,j,k,l}}\,, \\ \rho_{i,j,k,0} &= (\vec{\rho}_0)_{\mathrm{ind}^n_{i,j,k}}\,. \end{aligned} \tag{4.8}$$

Next, we collect all $v_{s,i,j,k}$ values in a vector $\vec{v}$ of length $3n$, using again a lexicographic ordering but now with respect to (s,i,j,k), which leads to the relation

$$v_{s,i,k,l} = (\vec{v})_{(s-1)n+\mathrm{ind}^n_{i,j,k}}\,. \tag{4.9}$$

We want to write (4.3) in the following matrix-vector form:

$$A(\vec{v})\vec{\rho} = b(\vec{v}, \vec{\rho}_0)\,, \tag{4.10}$$

where $A(\vec{v})$ is an $m \times m$ matrix and $b(\vec{v}, \vec{\rho}_0)$ is a vector of length m. This is possible, since $\rho_{i,j,k,l}$ appears only linearly in (4.3). In order to assemble the system matrix $A(\vec{v})$,

note first that (4.3) naturally divides into four parts, each part corresponding to the differential quotient with respect to one of the variables t, x, y or z. Hence, the system matrix $A(\vec{v})$ naturally splits up into four parts, i.e.,

$$A(\vec{v}) := A^t + A^x(\vec{v}) + A^y(\vec{v}) + A^z(\vec{v}), \tag{4.11}$$

where A^t, $A^x(\vec{v})$, $A^y(\vec{v})$ and $A^z(\vec{v})$ are $m \times m$ matrices corresponding to the differential quotients. They can be assembled by looping over all possible values of (i,j,k,l) and setting suitable values at the positions implicitly defined by the difference quotients. For example, for the matrix $A^x(\vec{v})$, the assembly procedure looks as follows:

- Create an all-zero $m \times m$ matrix A.

- For $i = 0, \ldots, I$, $k = 0, \ldots, K$, $j = 0, \ldots, J$ and $l = 1, \ldots, L$ do:

  Set $\text{ind} := \text{ind}^m_{i,j,k,l}$ and $\text{step} := (J+1)(K+1)L$.
  If (i = 0) {
  $\quad A_{\text{ind},\text{ind}+\text{step}} = \vec{v}_{\text{ind}}/\Delta x,$
  $\quad A_{\text{ind},\text{ind}} = -\vec{v}_{\text{ind}}/\Delta x.$
  }
  Elseif (i = I) {
  $\quad A_{\text{ind},\text{ind}} = \vec{v}_{\text{ind}}/\Delta x,$
  $\quad A_{\text{ind},\text{ind}-\text{step}} = -\vec{v}_{\text{ind}}/\Delta x.$
  }
  Else {
  $\quad A_{\text{ind},\text{ind}+\text{step}} = \vec{v}_{ind}/(2\Delta x),$
  $\quad A_{\text{ind},\text{ind}-\text{step}} = -\vec{v}_{ind}/(2\Delta x).$
  }

- Set $A^x(\vec{v}) := A$.

For the matrices A^t, $A^y(\vec{v})$ and $A^z(\vec{v})$, the assembly procedure looks similar, with obvious modifications due to the respective definitions of D_{y_j}, D_{z_k} and the backwards time difference quotient in (4.3).

From the assembly procedure above, one can see that $A^x(\vec{v})$ is a sparse matrix, with only three diagonals being non-zero. Similarly, also the other three matrices are sparse, with A^t having only 2 non-zero diagonals, $A^y(\vec{v})$ having 3 non-zero diagonals and $A^z(\vec{v})$ having 6 non-zero diagonals. Hence, the system matrix $A(\vec{v})$ is sparse as well, with (note that all four matrices share the non-zero main diagonal) only 11 non-zero diagonals.

As for the right-hand side in (4.10), one could again loop over all indices (i,j,k,l) to assemble it, or alternatively use the closed formula

$$b(\vec{v}, \vec{\rho}_0)_{\text{ind}^m_{i,j,k,l}} := \begin{cases} \frac{1}{(K+1)\Delta t}(\vec{\rho}_0)_{\text{ind}^n_{i,j,k}} - \frac{r}{2\Delta z}(\vec{\rho}_0)_{\text{ind}^n_{i,j,k+1}} (\vec{v})_{2n+\text{ind}^n_{i,j,k}} & l = 1, 1 \leq k < K, \\ \frac{1}{(K+1)\Delta t}(\vec{\rho}_0)_{\text{ind}^n_{i,j,k}} - \frac{r}{\Delta z}(\vec{\rho}_0)_{\text{ind}^n_{i,j,k+1}} (\vec{v})_{2n+\text{ind}^n_{i,j,k}} & l = 1, 1 \leq k < K, \\ 0 & \text{else}. \end{cases} \tag{4.12}$$

9

The forward problem consists in calculating $\vec{\rho}$ for given $\vec{v}$ and given initial data $\vec{\rho}_0$, via solving (4.10). Let us denote this solution by $\rho(\vec{v}, \vec{\rho}_0)$.

Note that in order to guarantee unique solvability of (3.3), one usually prescribes boundary conditions on $\partial\Omega$. However, since for our problem sufficient boundary data are not available, we used forward and backward differential quotients in the definition of D_{x_i} D_{y_j} and D_{z_k} at the boundary. It can easily be seen that this amounts to linear extrapolation of ρ and is also the reason why (4.10) turns out to be solvable.

4.1   The Inverse Problem

Let us now turn to the inverse problem. It consist in calculating the velocity $\vec{v}$ and the initial data $\vec{\rho}_0$ for given measurements of $\vec{\rho}$ and $\vec{\rho}_0$. Introducing the nonlinear operator

$$F(\vec{v}, \vec{\rho}_0) := (\rho(\vec{v}, \vec{\rho}_0), \vec{\rho}_0) \in \mathbb{R}^{m+n}, \tag{4.13}$$

our inverse problem can be written in the standard form

$$F(\vec{v}, \vec{\rho}_0) = (\vec{\rho}, \vec{\rho}_0). \tag{4.14}$$

The additional equation $\rho_0 = \rho_0$ in (4.14) seems to be superfluous at first. Note however, that as a result of measurement errors, we are not really given $\vec{\rho}$ and $\vec{\rho}_0$, but only noisy data $\vec{\rho}^\delta$ and $\vec{\rho}_0^\delta$ and hence, including this equation becomes necessary.

In order to solve (4.14), we will need the derivative and its adjoint of F. For this, we consider F as an operator from $\mathcal{X}$ to $\mathcal{Y}$, where

$$\begin{aligned} \mathcal{X} &= \{(\vec{v}, \vec{\rho}_0) \,|\, \vec{v} \in (\mathbb{R}^n)^3, \vec{\rho}_0 \in \mathbb{R}^n\}, \\ \mathcal{Y} &= \{(\vec{\rho}, \vec{\rho}_0) \,|\, \vec{\rho} \in \mathbb{R}^m, \vec{\rho}_0 \in \mathbb{R}^n\}. \end{aligned} \tag{4.15}$$

We equip $\mathcal{X}$ and $\mathcal{Y}$ with the inner products

$$\begin{aligned} \langle\, (\vec{v}, \vec{\rho}_0), (\vec{x}, \vec{w}_0) \,\rangle_\mathcal{X} &:= \vec{v}^T H \vec{x} + \vec{\rho}_0^T \vec{w}_0, \\ \langle\, (\vec{\rho}, \vec{\rho}_0), (\vec{w}, \vec{w}_0) \,\rangle_\mathcal{Y} &:= \vec{\rho}^T \vec{w} + \vec{\rho}_0^T \vec{w}_0, \end{aligned} \tag{4.16}$$

where H is a positive definite $3n \times 3n$ matrix chosen such that the inner product is an approximation of the H^1-inner product of functions v, see also Section 4.3.

Before we proceed with the derivation of the Frechet derivative and its adjoint, we introduce the following notation: Whenever we have an arbitrary Frechet-differentiable function G between suitable spaces A and B and we are given $x \in A$ and $\Delta x \in A$, then we denote by $G'(x)\Delta x$ the Frechet derivative of G at x in the direction of Δx. This notation will be used multiple times in the following:

Lemma 4.1. *Let $F : \mathcal{X} \to \mathcal{Y}$ be given as in (4.13) and let $(\vec{v}, \vec{\rho}_0) \in \mathbb{R}^{3n+n}$ and $(\Delta\vec{v}, \Delta\vec{\rho}_0) \in \mathbb{R}^{3n+n}$. Then for the Frechet derivative of F there holds*

$$F'(\vec{v}, \vec{\rho}_0)(\Delta\vec{v}, \Delta\vec{\rho}_0) = (\rho'(\vec{v}, \vec{\rho}_0)(\Delta\vec{v}, \Delta\vec{\rho}_0), \Delta\vec{\rho}_0), \tag{4.17}$$

where $\rho'(\vec{v}, \vec{\rho}_0)(\Delta\vec{v}, \Delta\vec{\rho}_0)$, the Frechet derivative of ρ, is given as the solution of

$$A(\vec{v})[\rho'(\vec{v}, \vec{\rho}_0)(\Delta\vec{v}, \Delta\vec{\rho}_0)] = -(A'(\vec{v})\Delta\vec{v})\rho(\vec{v}, \vec{\rho}_0) + b'(\vec{v}, \vec{\rho}_0)(\Delta\vec{v}, \Delta\vec{\rho}_0)\,, \qquad (4.18)$$

where A' and b' are the Frechet derivatives of A and b, respectively, and therefore $A'(\vec{v})\Delta\vec{v} \in \mathbb{R}^{m\times m}$ and $b'(\vec{v}, \vec{\rho}_0) \in \mathbb{R}^{m\times(3n+n)}$.

Proof. First, note that (4.17) follows immediately from the definition of the Frechet derivative. Now, from equation (4.10), we know that

$$A(\vec{v})\rho(\vec{v}, \vec{\rho}_0) = b(\vec{v}, \vec{\rho}_0)\,. \qquad (4.19)$$

Applying the Frechet-derivative at the point $(\vec{v}, \vec{\rho}_0)$ in the direction of $(\Delta\vec{v}, \Delta\vec{\rho}_0)$ to this equation and using the chain rule yields

$$(A'(v)\Delta\vec{v})\rho(\vec{v}, \vec{\rho}_0) + A(\vec{v})[\rho'(\vec{v}, \vec{\rho}_0)(\Delta\vec{v}, \Delta\vec{\rho}_0)] = b'(\vec{v}, \vec{\rho}_0)(\Delta\vec{v}, \Delta\vec{\rho}_0)\,, \qquad (4.20)$$

from which the statement of the lemma now immediately follows. $\qquad\square$

It follows from (4.17)) and (4.18) that in order to calculate $F'(\vec{v}, \vec{\rho}_0)(\Delta\vec{v}, \Delta\vec{\rho}_0)$, a linear system of equations involving the vectors $(A'(v)\Delta\vec{v})\rho(\vec{v}, \vec{\rho}_0)$ and $b'(\vec{v}, \vec{\rho}_0)(\Delta\vec{v}, \Delta\vec{\rho}_0)$ needs to be solved. It is possible to calculate those vectors without assembling the matrices $(A'(v)\Delta\vec{v})$ and $b'(\vec{v}, \vec{\rho}_0)$. However, as we will see in the lemma below, the assembly of three specific matrices will be inevitable for calculating the adjoint of the derivative of F, and those matrices can then also be used to compute the two required vectors.

To arrive at these matrices, note first that it follows from the assembly procedure described above that $A^x(\vec{v})$, $A^y(\vec{v})$ and $A^z(\vec{v})$ depend only linearly on $\vec{v}$. Together with (4.11) and the definition of the Frechet derivative, it follows that

$$A'(\vec{v})\Delta\vec{v} = A^x(\Delta\vec{v}) + A^y(\Delta\vec{v}) + A^z(\Delta\vec{v})\,, \qquad (4.21)$$

and hence $A'(\vec{v})\Delta\vec{v}$ is not only linear in $\Delta\vec{v}$ but also independent of $\vec{v}$. As a result, it is possible to find a matrix $D_A(\rho) \in \mathbb{R}^{m\times 3n}$ such that

$$(A'(\vec{v})\Delta\vec{v})\rho(\vec{v}, \vec{\rho}_0) = D_A(\rho(\vec{v}, \vec{\rho}_0))\Delta\vec{v}\,. \qquad (4.22)$$

As for the other two matrices, note that once one has assembled $b'(\vec{w}, \vec{\rho}_0)$, which can be calculated easily using (4.12), this matrix can be split up into two sub-matrices, i.e.,

$$b'(\vec{w}, \vec{\rho}_0) = \left(b'_{\Delta\vec{v}}(\vec{w}, \vec{\rho}_0) \quad | \quad b'_{\Delta\vec{\rho}_0}(\vec{w}, \vec{\rho}_0)\right)\,, \qquad (4.23)$$

where $b'_{\Delta\vec{v}}(\vec{w}, \vec{\rho}_0) \in \mathbb{R}^{m\times 3n}$ and $b'_{\Delta\vec{\rho}_0}(\vec{w}, \vec{\rho}_0) \in \mathbb{R}^{m\times n}$ are the derivatives corresponding to $\Delta\vec{v}$ and $\Delta\vec{\rho}_0$, respectively.

Thanks to the special structure of $A(\vec{v})$ and $b(\vec{v}, \vec{\rho}_0)$, when following the above derivation steps in detail, one finds out that most of the elements of the matrices

$$D_A(\rho) \qquad \text{and} \qquad b'(\vec{v}, \vec{\rho}_0) \qquad (4.24)$$

are zero, with at most three non-zero elements in each row in both cases.

Using the above derivations, we can now prove the following

Lemma 4.2. *Let $F : \mathcal{X} \to \mathcal{Y}$ be given as in (4.13) and let $(\vec{v}, \vec{\rho}_0) \in \mathbb{R}^{3n+n}$ and $(\vec{w}, \vec{w}_0) \in \mathbb{R}^{m+n}$. Then for the adjoint of the Frechet derivative of F there holds*

$$F'(\vec{v}, \vec{\rho}_0)^*(\vec{w}, \vec{w}_0) := \begin{pmatrix} H^{-1} \left(-D_A(\rho(\vec{v}, \vec{\rho}_0))^T + b'_{\Delta\vec{v}}(\vec{v}, \vec{\rho}_0)^T \right) A(\vec{v})^{-T}\vec{w} \\ b'_{\Delta\vec{\rho}_0}(\vec{v}, \vec{\rho}_0)^T A(\vec{v})^{-T}\vec{w} + \vec{w}_0 \end{pmatrix} . \tag{4.25}$$

Proof. To compute the adjoint, consider first

$$\langle\, F'(\vec{v}, \vec{\rho}_0)(\Delta\vec{v}, \Delta\vec{\rho}_0), (\vec{w}, \vec{w}_0) \,\rangle_{\mathcal{Y}} =$$
$$= \langle\, (-A(\vec{v})^{-1}D_A(\rho(\vec{v}, \vec{\rho}_0))\Delta v + A(\vec{v})^{-1}b'(\vec{v}, \vec{\rho}_0)(\Delta\vec{v}, \Delta\vec{\rho}_0), \Delta\vec{\rho}_0), (\vec{w}, \vec{w}_0) \,\rangle_{\mathcal{Y}}$$
$$= \left(-A(\vec{v})^{-1}D_A(\rho(\vec{v}, \vec{\rho}_0))\Delta v\right)^T \vec{w} + \left(A(\vec{v})^{-1}b'(\vec{v}, \vec{\rho}_0)(\Delta\vec{v}, \Delta\vec{\rho}_0)\right)^T \vec{w} + \Delta\vec{\rho}_0^T \vec{w}_0$$
$$= -\Delta\vec{v}^T D_A(\rho(\vec{v}, \vec{\rho}_0))^T A(\vec{v})^{-T}\vec{w} + (\Delta\vec{v}, \Delta\vec{\rho}_0)^T b'(\vec{v}, \vec{\rho}_0)^T A(\vec{v})^{-T}\vec{w} + \Delta\vec{\rho}_0^T \vec{w}_0$$

Splitting up $b'(\vec{w}, \vec{\rho}_0)$ as in (4.23), we get

$$\langle\, F'(\vec{v}, \vec{\rho}_0)(\Delta\vec{v}, \Delta\vec{\rho}_0), (\vec{w}, \vec{w}_0) \,\rangle_{\mathcal{Y}} = \Delta\vec{\rho}_0^T b'_{\Delta\vec{\rho}_0}(\vec{v}, \vec{\rho}_0)^T A(\vec{v})^{-T}\vec{w} + \Delta\vec{\rho}_0^T \vec{w}_0$$
$$+ \Delta\vec{v}^T \left(-D_A(\rho(\vec{v}, \vec{\rho}_0))^T + b'_{\Delta\vec{v}}(\vec{v}, \vec{\rho}_0)^T\right) A(\vec{v})^{-T}\vec{w}$$

and hence, using the definition of the inner product in $\mathcal{X}$, the statement follows. $\square$

4.2 Incorporating the Divergence-Free Condition

Up to now, the divergence-free condition $\operatorname{div} v = 0$ on the velocity field to be reconstructed did not enter the reconstruction method. However, it is a modelling assumption and has to be taken care of.

One possible way to do so would be to incorporate the condition into the space $\mathcal{X}$, i.e., allowing only divergence-free vector fields in $\mathcal{X}$. This approach essentially, except at the boundary, implies $v_3 = v_3(v_1, v_2)$. This changes the derivative and its adjoint of F in a computationally unfavourable way and hence we avoid this approach.

Instead, we enforce the divergence-free condition in a weak way, by changing F to

$$F(\vec{v}, \vec{\rho}_0) := (\rho(\vec{v}, \vec{\rho}_0), \vec{\rho}_0, D\vec{v}) , \tag{4.26}$$

where D is a matrix representing the divergence-free condition. The operator F now maps from $\mathcal{X}$ to $\mathcal{Y}$ with $\mathcal{X}$ as before and

$$\mathcal{Y} = \{(\vec{\rho}, \vec{\rho}_0, \vec{w}) \,|\, \vec{\rho} \in \mathbb{R}^m \,, \vec{\rho}_0 \in \mathbb{R}^n \,, \vec{w} \in \mathbb{R}^n\} , \tag{4.27}$$

where we use the following inner product

$$\langle\, (\vec{\rho}, \vec{\rho}_0, \vec{v}_d), (\vec{w}, \vec{w}_0, \vec{w}_d) \,\rangle_{\mathcal{Y}} = \vec{\rho}^T\vec{w} + \vec{\rho}_0^T\vec{w}_0 + \vec{v}_d^T\vec{w}_d . \tag{4.28}$$

The resulting nonlinear inverse problem now reads as

$$F(\vec{v}, \vec{\rho}_0) = (\rho, \rho_0, \vec{0}) . \tag{4.29}$$

12

An analogous calculation as before yields that the Frechet derivative of F is given by

$$F'(\vec{v}, \vec{\rho}_0)(\Delta \vec{v}, \Delta \vec{\rho}_0) = (\rho'(\vec{v}, \vec{\rho}_0), \Delta \vec{\rho}_0, D\Delta \vec{v}), \tag{4.30}$$

and that the adjoint is given by

$$F'(\vec{v}, \vec{\rho}_0)^*(\vec{w}, \vec{w}_0, \vec{w}_d) = \begin{pmatrix} H^{-1} \left[\left(-D_A(\rho(\vec{v}, \vec{\rho}_0))^T + b'_{\Delta \vec{v}}(\vec{v}, \vec{\rho}_0)^T \right) A(\vec{v})^{-T} \vec{w} + D^T \vec{w}_d \right] \\ b'_{\Delta \vec{\rho}_0}(\vec{v}, \vec{\rho}_0)^T A(\vec{v})^{-T} \vec{w} + \vec{w}_0 \end{pmatrix}.$$

As for the choice of the matrix D, notice that since, due to the Divergence Theorem,

$$\operatorname{div} v(x, y, z) = \lim_{|V| \to 0} \frac{1}{|V|} \int_{\partial V} v \; dS(x, y, z), \tag{4.31}$$

we may approximate the divergence in $\Omega_{i,j,k} := [x_{i-1}, x_i] \times [y_{j-1}, y_j] \times [z_{k-1}, z_k]$ by the integral above with $V = \Omega_{i,j,k}$. Assuming that each component of v is piecewise linear (tri-linear), divergence-free then means

$$D_1 v_{1,i,j,k} + D_2 v_{2,i,j,k} + D_3 v_{3,i,j,k} = 0, \tag{4.32}$$

where $1 \le i \le I$, $1 \le j \le J$, $1 \le k \le K$, and

$$
\begin{aligned}
D_1 v_{1,i,j,k} \;\; &:= \;\; \frac{1}{4\Delta x} (v_{1,i,j-1,k-1} - v_{1,i-1,j-1,k-1} + v_{1,i,j-1,k} - v_{1,i-1,j-1,k} \tag{4.33} \\
&\qquad\qquad + v_{1,i,j,k-1} - v_{1,i-1,j,k-1} + v_{1,i,j,k} - v_{1,i-1,j,k}), \\
D_2 v_{2,i,j,k} \;\; &:= \;\; \frac{1}{4\Delta y} (v_{2,i-1,j,k-1} - v_{2,i-1,j-1,k-1} + v_{2,i-1,j,k} - v_{2,i-1,j-1,k} \tag{4.34} \\
&\qquad\qquad + v_{2,i,j,k-1} - v_{2,i,j-1,k-1} + v_{2,i,j,k} - v_{2,i,j-1,k}), \\
D_3 v_{3,i,j,k} \;\; &:= \;\; \frac{1}{4\Delta z} (v_{3,i-1,j-1,k} - v_{3,i-1,j-1,k-1} + v_{3,i-1,j,k} - v_{3,i-1,j,k-1} \tag{4.35} \\
&\qquad\qquad + v_{3,i,j-1,k} - v_{3,i,j-1,k-1} + v_{3,i,j,k} - v_{3,i,j,k-1}).
\end{aligned}
$$

The (sparse!) matrix D is now built such that $D\vec{v} = 0$ is equivalent to (4.32).

Whenever we speak of the *weak divergence-free option* in subsequent chapters, we mean that we use F defined as in (4.26). As it will turn out in our numerical tests below, using this option has a significant effect on the reconstructed solutions.

4.3 Choosing the matrix H

We now turn to the choice of the matrix H in the inner product of $\mathcal{X}$. From the theory of transport equations (see e.g. [6]), we know that v should be at least an H^1 velocity field. Assuming as above that each component of v is piecewise linear and can hence be written in the form

$$v(x, y, z) = \sum_{i,j,k} \begin{pmatrix} v_{1,i,j,k} \\ v_{2,i,j,k} \\ v_{3,i,j,k} \end{pmatrix} \psi_{i,j,k}(x, y, z), \tag{4.36}$$

where the $\psi_{i,j,k}$ are the 3D hat functions commonly used to form a basis in H^1-FEM, we find that the optimal choice of H in this case would be,

$$H = c_s \operatorname{diag}(\tilde{H}, \tilde{H}, \tilde{H}), \quad \tilde{H} = [\langle \psi_{i,j,k}, \psi_{l,m,n} \rangle_{H^1}]. \tag{4.37}$$

where c_s is a suitable scaling constant. As can easily be seen, $\tilde{H}$ is the FEM system matrix of the equation $-\Delta u + u = f$. However, in the computation of the adjoint of F we need to apply H^{-1}, or equivalently three times $\tilde{H}^{-1}$, which amounts to solving three perturbed Laplace equations in each iteration step. This is way too costly and hence we need to find a suitable alternative for inverting the matrix H.

One possibility is to approximate $\tilde{H}$ by its diagonal part, which leads to a diagonal matrix H that is easy to assemble and to invert, i.e.,

$$H = c_s \operatorname{diag}(\tilde{H}, \tilde{H}, \tilde{H}), \quad \tilde{H} = \operatorname{diag}(\langle \psi_{i,j,k}, \psi_{i,j,k} \rangle_{H^1}). \tag{4.38}$$

The scaling constant c_s is chosen such that the two terms on the right hand side of (4.16) are balanced and that the H^1 and the L^2 norm approximations of constant vectors coincide, which leads to the choice

$$c_s = 3n \left(\sum_{i=1}^{3n} H_{ii} \right)^{-1}. \tag{4.39}$$

Another, more sophisticated possibility is to use an orthogonal system $\psi_{i,j,k}$, e.g., wavelets, since then the matrix $\tilde{H}$ becomes diagonal. One can see that applying $\tilde{H}^{-1}$ coincides with applying the operator i_1^*, where $i_1 : H^1 \to L^2$ is the embedding operator. For a given wavelet system $\{\phi, \psi\}$, every function $f \in L^2$ can be expanded as

$$f = \sum_{k \in \mathbb{Z}} \langle f, \phi_{0k} \rangle \phi_{0k} + \sum_{j=0}^{\infty} \sum_{k \in \mathbb{Z}} \langle f, \psi_{jk} \rangle \psi_{jk}, \tag{4.40}$$

where $\phi_{0k} = \phi(t - k)$ and $\psi_{jk}(t) = 2^{j/2}\psi(2^j t - k)$. If the wavelet system is sufficiently smooth, then for every Sobolev space, the H^s inner product of two functions f and g is equivalent to

$$\langle f, g \rangle_{H^s} = \sum_{k \in \mathbb{Z}} \langle f, \phi_{0k} \rangle \langle g, \phi_{0k} \rangle + \sum_{j=0}^{\infty} 2^{2js} \sum_{k \in \mathbb{Z}} \langle f, \psi_{jk} \rangle \langle g, \phi_{jk} \rangle. \tag{4.41}$$

Following [27], we see that the adjoint i_s^* of the embedding $i_s : H^s \to L^2$ is given by

$$i_s^* g = \sum_{k \in \mathbb{Z}} \langle g, \phi_{0k} \rangle \phi_{0k} + \sum_{j=0}^{\infty} \sum_{k \in \mathbb{Z}} \frac{\langle g, \psi_{jk} \rangle}{2^{2js}} \psi_{jk}. \tag{4.42}$$

Using this, we can, instead of applying $\tilde{H}^{-1}$ to the components of $\vec{v}$, compute their discrete wavelet transforms, weight the resulting coefficients according to (4.42) and

then apply the inverse discrete wavelet transforms. The computation of $\vec{v}^T H \vec{w}$ in (4.16) is then replaced by using a scaled version of (4.41), using again the discrete wavelet transform. Thus, whenever we speak of using the *wavelet embedding option* in subsequent chapter, we mean that this procedure is being used.

Note that for the results presented below, due to the low spatial resolution, using the wavelet embedding option with $s = 1$ in (4.41) and (4.42) leads to an undesirably high amount of smoothing and subsequently to mediocre results. Using a smaller s and hence less smoothing yields much better results and therefore, the choice $s = 0.1$ was used in all computations below. As for the choice of wavelets, Daubechies 3 wavelets (see [5]) were used in all cases.

Both approximations of H^{-1}, using only the diagonal entries of H and via the wavelet embedding are very fast (diagonal matrix inversion and $\mathcal{O}(n)$ wavelet decomposition). The use of wavelets has the additional advantage that it yields a very good approximation of the application of H^{-1}, as compared to using the diagonal approximation, which in essence only amounts to a scaling of the steps in the iterative solution method introduced below.

5   Regularization Approach

The stable solution of ill posed (nonlinear) equations $F(x) = y$ requires the use of regularization methods, see [10] for an overview of methods. Tikhonov regularization is probably the best known method. For nonlinear operators F, Tikhonov regularization results in the minimization of a functional consisting of a data fit term and a penalty term. Besides this, there exists a wide array of gradient based iterative methods such as *Landweber iteration* or the *iteratively regularized Gauß-Newton method* (see [16]). Even though Gauß-Newton and other sophisticated methods are considered to be much faster than Landweber iteration, they face a serious drawback when confronted with largescale problems, since in every iteration step, a system of equations with a full system matrix has to be solved. Hence, since in our MRAI problems we often deal with more than one million unknowns, we resort to the Landweber iteration, which is given by (see [10])

$$x_{k+1}^\delta = x_k^\delta + \omega\, F'(x_k^\delta)^*(y^\delta - F(x_k^\delta))\,, \qquad x_0^\delta = x_0\,, \tag{5.1}$$

where x_0 is a suitable initial guess and ω is a scaling parameter which has to satisfy

$$\omega\, \|F'(x)\| < 1\,, \tag{5.2}$$

for all x in a sufficiently large neighbourhood around the initial guess x_0. The superscript δ in (5.1) indicates that we are only given noisy data y^δ instead of y. The most common assumption on the data is that

$$\|y - y^\delta\| \le \delta\,. \tag{5.3}$$

Since we are dealing with an ill-posed problem, a suitable stopping rule is needed in order to achieve reasonable approximations of the solution. Typically, one uses the

discrepancy principle, i.e., the iteration is stopped after k_* steps, where k_* is the smallest index fulfilling

$$\left\| y^\delta - F(x_{k_*}^\delta) \right\| \leq \tau\delta \leq \left\| y^\delta - F(x_k^\delta) \right\| , \quad 0 \leq k \leq k_* , \tag{5.4}$$

where τ is an appropriately chosen positive number.

Determining an appropriate ω in (5.1) is very important but difficult. Hence we use an iteration dependent scaling parameter $\omega^\delta(z)$ defined by:

$$\omega^\delta(z) := \frac{\left\| s^\delta(z) \right\|^2}{\left\| F'(z)s^\delta(z) \right\|^2} , \qquad s^\delta(z) := F'(z)^*(y^\delta - F(z)) , \tag{5.5}$$

which leads to

$$x_{k+1}^\delta = x_k^\delta + \omega^\delta(x_k^\delta)\, F'(x_k^\delta)^*(y^\delta - F(x_k^\delta)) , \qquad x_0^\delta = x_0 , \tag{5.6}$$

the so-called *steepest descent* method. For details, see for example [16, Chapter 3.4].

As is well known, both the Landweber iteration and the steepest descent method converge rather slowly. Hence, in order to reduce the number of required iterations, we follow the idea of Nesterov's acceleration scheme [24] and introduce an intermediate step into our iteration, which now reads as follows:

$$\begin{aligned} z_k^\delta &= x_k^\delta + \tfrac{k-1}{k+2}\left(x_k^\delta - x_{k-1}^\delta \right) , \\ x_{k+1}^\delta &= z_k^\delta + \omega^\delta(z_k^\delta)\, F'(z_k^\delta)^*(y^\delta - F(z_k^\delta)) , \\ x_0^\delta &= x_{-1}^\delta = x_0 . \end{aligned} \tag{5.7}$$

A similar intermediate step was also used in the highly successful FISTA algorithm for linear inverse problems [3], where additionally the convergence rate $\mathcal{O}(k^{-2})$ was proven; a big improvement over the classical rate $\mathcal{O}(k^{-1})$. Although the authors are not aware of any publication proving the convergence rate $\mathcal{O}(k^{-2})$ for our method (5.7), numerical tests showed that using (5.7) instead of (5.6) leads to a significant decrease in iteration number, even for our nonlinear problem (4.14). This is remarkable, since the intermediate steps z_k^δ are simply linear combinations of previous iterations and are therefore very cheap to compute.

5.1   Implementation Details

The implementation of (5.7) seems very straightforward at first, since we have explicit expressions for F' and its adjoint available. However, even though we will be dealing with a rather coarse space discretization, since we are essentially using a *space-time approach* with three space dimensions, the problem becomes largescale, with around 3.3 million unknowns for one of the real-world data sets considered below. This causes severe numerical difficulties.

Please note that for the calculation of one iteration step it is necessary to solve three large sparse linear systems of equations, one for calculating F, one for F' and one

for ω^δ. Since, due to the size of the problem, this can no longer be done directly, the iterative solver biCGstab with an incomplete LU factorization preconditioner was used.

The implementation of the method was done in MATLAB R2015b. Since some built-in MATLAB functions were too slow for our purposes, we had to rely on *fsparse.m*, a function for creating sparse matrices, see [9].

5.2 Enforcing Sparse Solutions

The relevant blood vessels in which the pulse waves travel constitute only a minor part of the brain. Hence, the velocity vector field v which we seek to reconstruct should take nonzero values in those blood vessels only.

In mathematical terms this means that v should be compactly supported and should have a sparse representation in the basis $\psi_{i,j,k}$. The reconstruction algorithm should take this into account, which leads directly to the concept of sparsity regularization.

Following [28] and [20], we seek to compute $(\vec{v}, \vec{\rho}_0)$ as a minimizer of the functional

$$\frac{1}{2} \left\| F(\vec{v}, \vec{\rho}_0) - (\vec{\rho}^\delta, \vec{\rho}_0^\delta) \right\|_{\mathcal{Y}}^2 + \alpha \left(\sum_{l=1}^{3} \sum_{i,j,k} \omega_{i,j,k}^{\vec{v}_l} |v_{l,i,j,k}|^p + \sum_{i,j,k} \omega_{i,j,k}^{\vec{\rho}_0} |\rho_{i,j,k,0}|^p \right) ,$$

where $\omega_{i,j,k}^{\vec{v}_l}$ and $\omega_{i,j,k}^{\vec{\rho}_0}$ are positive weights bounded away from zero, α is a regularization parameter and $p \in [1, 2]$. The choice $p = 1$ yields sparse minimizers, while $1 < p < 2$ is suspected to promote sparsity.

For further use below, and for $1 \leq p < \infty$ and $\tau > 0$, we define the *real valued shrinkage function* $S_{\tau,p} : \mathbb{R} \to \mathbb{R}$ via

$$S_{\tau,p}(x) = \begin{cases} \text{sgn}(x) \max(|x| - \tau, 0) & p = 1 , \\ G_{\tau,p}^{-1}(x) & p \in (1, 2] , \end{cases} \tag{5.8}$$

where

$$G_{\tau,p}(x) = x + \tau \text{sgn}(x) |x|^{p-1} . \tag{5.9}$$

For a vector $\vec{x} = \{x_k\}_{k \in \Lambda}$ and weights $\omega = \{\omega_k\}_{k \in \Lambda}$ we define the shrinkage function $S_{\omega,p}$ via

$$S_{\omega,p}(\vec{x}) := [S_{\omega_k,p}(x_k)]_{k \in \Lambda} . \tag{5.10}$$

Following [20], a possible method for solving nonlinear inverse problems $F(x) = y^\delta$ involving sparsity constraints is given by the so-called *iterated soft shrinkage algorithm*, which reads as

$$x_{k+1}^\delta = S_{\omega_k \alpha \omega, p} \left(x_k^\delta + \omega_k F'(x_k^\delta)^* (y^\delta - F(x_k^\delta)) \right) , \quad x_0^\delta = x_0 . \tag{5.11}$$

For our problem we combine this approach with the accelerated gradient method (5.7) to arrive, after collecting the $\omega^{\vec{v}_l}$ and $\omega^{\vec{\rho}_0}$ into a single sequence ω^s, at the following iterative scheme:

$$(\vec{w}_k^\delta, \vec{w}_{0,k}^\delta) = (\vec{v}_k^\delta, \vec{\rho}_{0,k}^\delta) + \frac{k-1}{k+2} \left((\vec{v}_k^\delta, \vec{\rho}_{0,k}^\delta) - (\vec{v}_{k-1}^\delta, \vec{\rho}_{0,k-1}^\delta) \right) ,$$

$$\omega_k^\delta = \omega^\delta((\vec{w}_k^\delta, \vec{w}_{0,k}^\delta)) ,$$

$$(\vec{v}_{k+1}^\delta, \vec{\rho}_{0,k+1}^\delta) = S_{\omega_k^\delta \alpha \omega^s, p} \left((\vec{w}_k^\delta, \vec{w}_{0,k}^\delta) + \omega_k^\delta F'(\vec{w}_k^\delta, \vec{w}_{0,k}^\delta)^* ((\vec{\rho}^\delta, \vec{\rho}_0^\delta) - F(\vec{w}_k^\delta, \vec{w}_{0,k}^\delta)) \right) .$$

Note that for this algorithm, values for α, ω^s and p need to be specified. Moreover, if $p > 1$, $4n$ nonlinear equations need to be solved for approximating $G_{\tau,p}^{-1}$. This is very costly, hence we use $p = 1$ only. Furthermore, since we want to weight all $v_{l,i,j,k}$ and $\rho_{i,j,k}$ equally, ω^s was set to 1 for the numerical examples presented below.

Obviously, the above algorithm can also, with minor modifications, be applied to the case when F contains the divergence-free part as introduced in Section 4.2.

6   Results

In this section, we present several results obtained by using the method described above on simulated and real-world data. We compare different choices of parameters and the effects of the sparsity and the weak divergence-free option as well as the different approximations of H described in Section 4.3.

For all examples, a maximum intensity projection (MIP) over the z-axis of the norm of the velocity vector field was calculated. Afterwards, a colour direction MIP was created by assigning a RGB value to every pixel of the MIP. This was done by first considering, for every pixel of the MIP, that voxel whose velocity norm value was responsible for the entry of the MIP at that pixel. The absolute values of the $\vec{v}_1$, $\vec{v}_2$ and $\vec{v}_3$ values of that voxel were then taken as the red, green and blue values of the RGB triplet at that pixel, respectively. This means that a red pixel in the colour direction MIP indicates movement along the x-axis, a green pixel along the y-axis and a blue pixel along the z-axis. Finally, all RGB values were divided by the maximum absolute RGB value of the colour direction MIP and the resulting map was divided by a factor of 0.6 in order to enhance colours.

6.1   Simulated Data

In this subsection, we test our algorithm on simulated data. For this purpose, a phantom of size $40 \times 30 \times 30$ was created, featuring several blood vessels, i.e. pipes, of variable thickness and orientation. A projection of this phantom over the z-axis can be seen in Figure 6.1, which not only shows the vessels themselves but also the norm of the velocity vector field (left figure) and the a colour direction MIP of the velocity (right figure) moving through the vessels.

Looking at the colour direction MIP in Figure 6.1, the three red horizontal vessels on the bottom of the phantom move along the x-axis and have a thickness of 1, 2 and 3 voxels, respectively. The three blue vessels above them move along the z-axis, and hence only a small part of them can be seen in the picture. Note that one of those vessels has a plus-shaped cross-section, which is also the case for the bottommost of the three red blood vessels. The three orange-red blood vessels move diagonally across the x-y-plane and have a z-thickness of 1 voxel each. Both the orange and the red vessels lie in the middle of the z-plane, while the blue vessels extend over the entire range of the z-axis.

As for the simulation of the data $\rho_{i,j,k,l}$, consider first the case of a signal $\rho_0(x, y, z)$
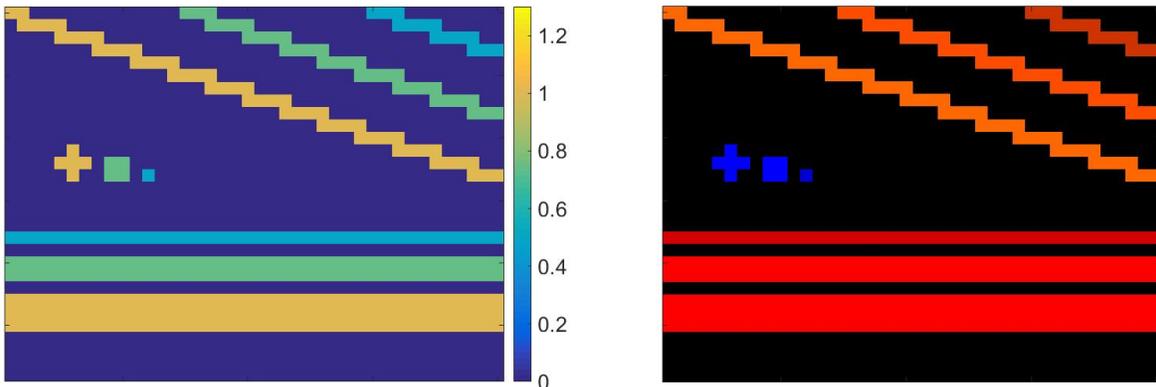
Figure 6.1: Simulation phantom: Magnitude of the norm of the velocity vector field (left figure) and colour direction MIP of the velocity (right figure).

transported via a constant velocity field $\bar{v} = (\bar{v}_1, \bar{v}_2, \bar{v}_3)$. It can be easily seen that in this case

$$\rho(x, y, z, t) = \rho_0(x - \bar{v}_1 t, y - \bar{v}_2 t, z - \bar{v}_3 t), \tag{6.1}$$

solves the advection equation with initial guess ρ_0. If in each vessel we prescribe a constant velocity vector field pointing in one of the two directions of the vessel, then for a given initial signal ρ_0 we can calculate the solution of the advection equation in that vessel via (6.1). Summing up those solutions for all the different vessels and sampling at the correct space-time points then gives us the data $\rho_{i,j,k,l}$. Adding a random data error of fixed magnitude, e.g., 1%, we arrive at the final data used in the simulation.

In our simulation, for a vessel with given velocity $\bar{v}$, we used the initial signal

$$\rho_0(x, y, z) = \sin\left(\frac{6\pi}{\|\bar{v}\|_2}\left(\frac{\bar{v}_1}{I\Delta x}x + \frac{\bar{v}_2}{J\Delta y}y + \frac{\bar{v}_3}{K\Delta z}z\right)\right).$$

To make the simulation procedure a bit clearer, consider the bottommost vessel in Figure 6.1. Prescribing for example the velocity $\bar{v} = (c, 0, 0)$ in that vessel leads to

$$\rho(x, y, z, t) = \sin\left(\frac{6\pi(x - c\,t)}{I\Delta x}\right), \tag{6.2}$$

and the data $\rho_{i,j,k,l}$, for those (i, j, k) for which (x_i, y_j, z_k) belongs to the vessel under consideration, is then defined via

$$\rho_{i,j,k,l} = \rho(x_i, y_j, z_k, t_{k,l}) = \sin\left(\frac{6\pi(x_i - c\,t_{k,l})}{I\Delta x}\right). \tag{6.3}$$

We apply the same procedure to all the remaining vessels and set $\rho_{i,j,k,l}$ to 0 whenever (i, j, k) does not correspond to any vessel. Finally, a randomly generated data error of magnitude delta is added.

Note that the velocity vector field v underlying this data simulation is constant in each vessel and hence locally, but not globally, in H^1. Even though we have derived our

solution method from the assumption of a globally H^1 velocity vector field, using this simulation makes sense, since for real MRI data we also expect nonzero velocities to occur *inside* blood vessels only, which renders the velocity vector field to be reconstructed only locally H^1 as well.

For the results presented below, we have chosen $\Delta x = \Delta y = \Delta z = 1\,\text{mm}$ and, as mentioned above, $I = 39$, $J = 29$, $K = 29$ for the space discretization, as well as $L = 4$. As for the time discretization Δt, notice that our forward solver belongs to the class of BTCS (backward in time, central in space) finite difference methods, which are implicit methods requiring no restriction on the time stepsize Δt to achieve stability. However, in order to get good accuracy of the forward solver, Δt should be chosen small enough. Denoting with ΔT the duration of a full measurement circle, i.e., $\Delta T = (K + 1)\Delta t$, it turned out that a suitable bound is given by the CFL-type condition

$$\Delta T \, \|v\|_2 \lesssim \frac{\Delta x}{10} \,. \tag{6.4}$$

Using a ΔT significantly greater than this bound was found to lead to large errors in the reconstructed velocity (see Figure 6.5). Hence, for our tests below, we used $\Delta T = 0.1\,\text{s}$, which by (6.4) allows for velocities with a maximum norm of $1\,\text{mm/s}$.
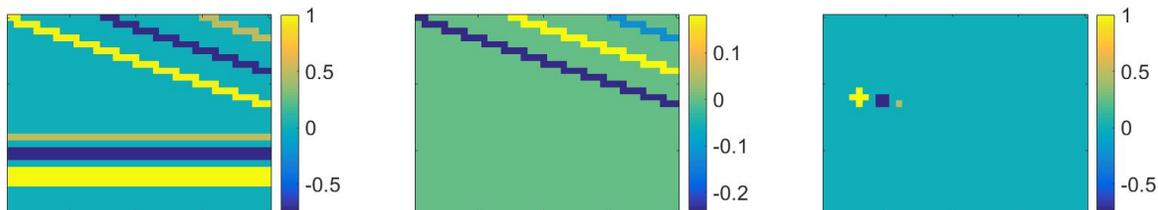

Figure 6.2: Magnitudes of the velocity vector field components. Left: First component. Middle: Second component. Right: Third component.

For our simulations, we have used velocities $\bar{v}$ with three different magnitudes $\|\bar{v}\|_2$, which can be seen in Figure 6.1. The orientation of these velocity vector fields is depicted in more detail in Figure 6.2, which shows the values of the three velocity components, revealing also the different orientations of the simulated pulse waves.

For all tests below, a random data error of magnitude δ was added and the iteration was stopped using the discrepancy principle (5.4) together with the choice of $\tau = 1$. Furthermore, if not noted otherwise, the matrix H introduced in Section 4.3 is used instead of the wavelet embedding described in the same chapter.

As a first test, we use our method without any special options, i.e., neither using the weak divergence-free option, nor the sparsity or the wavelet embedding option. The resulting approximation, achieved after 90 iterations, can be seen in Figure 6.3. The structure of the vessels can clearly be identified and also the reconstructed velocity is partially correct. However, the sinusoidal structure of the initial signal ρ_0 is visibly transferred to the reconstructed velocity.

Figure 6.4 shows the results of the second test: the weak-divergence free option was included in the reconstruction algorithm, which now stopped after 126 iterations. The
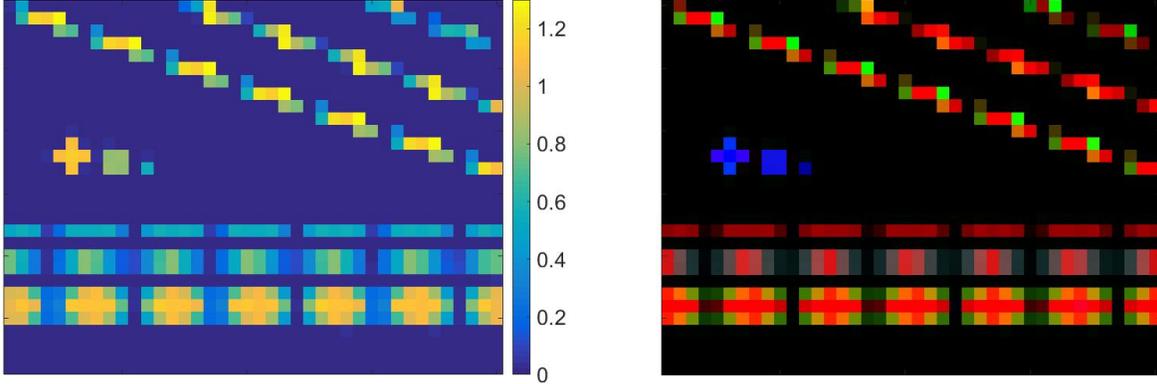
Figure 6.3: Result of the algorithm applied to the test problem ($\delta = 1\%$), using no additional options. Velocity norm MIP (left) and colour direction MIP (right).
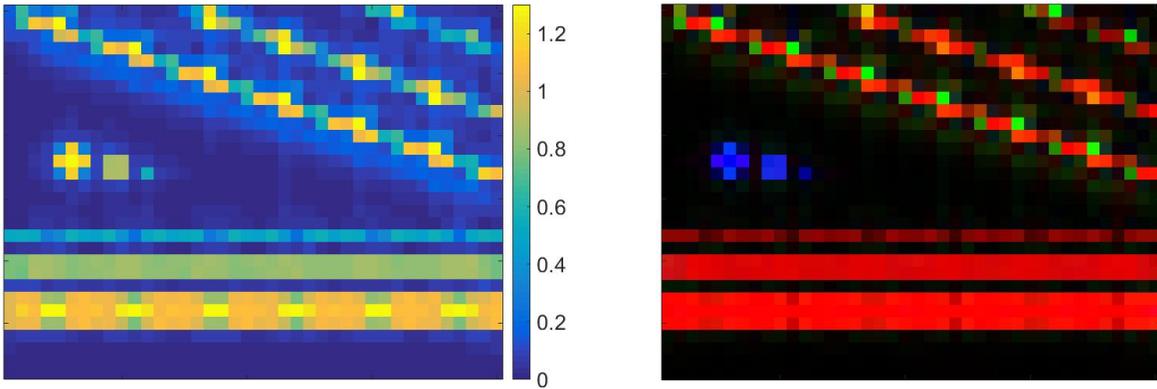

Figure 6.4: Result of the algorithm applied to the test problem ($\delta = 1\%$), using the weak divergence-free option. Velocity norm MIP (left) and colour direction MIP (right).

reconstructed velocity is much smoother than before, now resembling the true solution much more closely. However, as could be expected, using the weak divergence-free calculation option leads to a smoothing of the solution, clearly visible in the reconstruction around the vessels.

Next, we want to see what happens if (6.4) is no longer satisfied. For this, we use the same problem setup as before, but now multiply the velocities in all the vessels with the factor 10, such that the norm of the velocity vector has a maximum value of $\|v\|_2 = 10$. The results of the algorithm applied to this problem can be seen in Figure 6.5, the iteration having stopped after 194 iterations due to a detected increase of the residual. One can clearly see that the velocities are strongly underestimated, although the location of the vessels, the directions of the pulse waves and also some small qualitative differences in the norm of the velocity vector field are still detectable. Note that one could think of ignoring the increase in residual and continuing the iteration. This, however, does not lead to improved results, the algorithm simply not being able to improve the reconstruction any further.

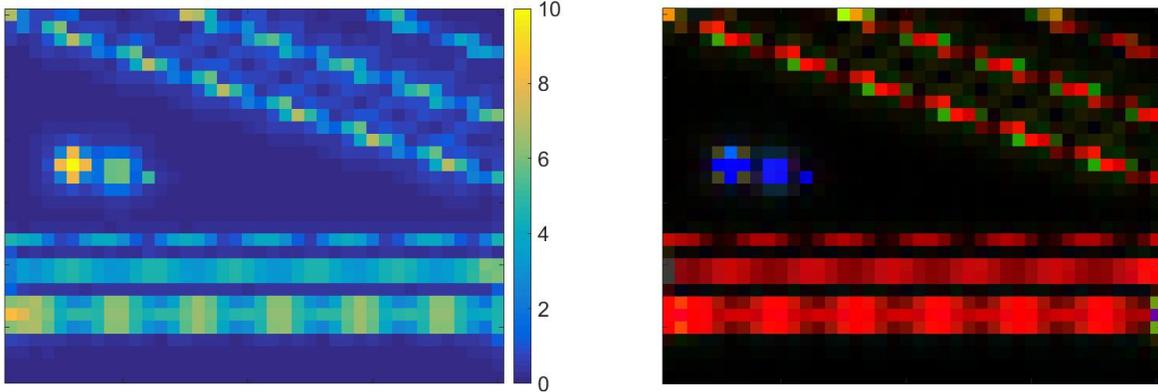For the next and all the subsequent tests, we return again to the original simulated

Figure 6.5: Result of the algorithm applied to the modified problem ($\delta = 1\%$), where all involved velocities were multiplied by the factor 10, using the weak divergence-free option. Velocity norm MIP (left) and colour direction MIP (right).
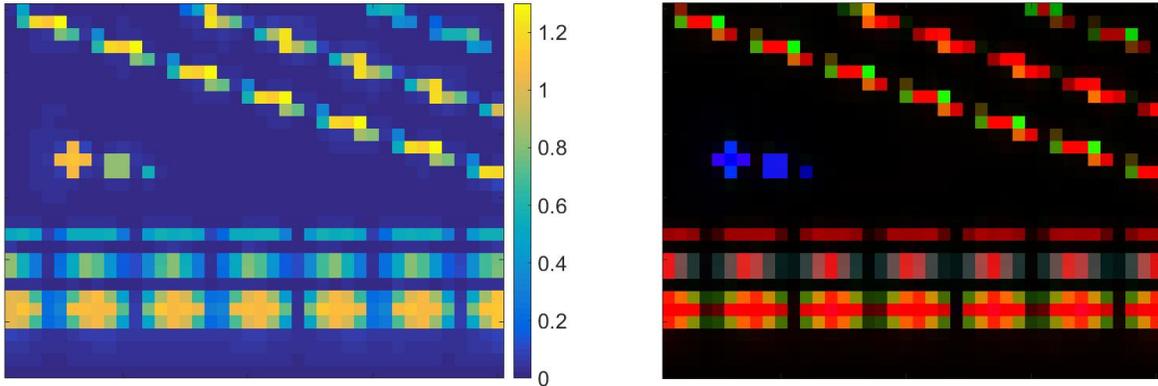

Figure 6.6: Result of the algorithm applied to the test problem ($\delta = 1\%$), using the wavelet embedding option. Velocity norm MIP (left) and colour direction MIP (right).

data, this time applying the algorithm together with the wavelet embedding option. The stopping criterion was met after 121 iterations and the results can be seen in Figure 6.6. A comparison with Figure 6.3 shows that using this option mainly leads to a smoothing of the reconstructed velocity, comparable to but not as strong as using the weak divergence-free option.

The results of combining the weak-divergence free and the wavelet embedding options can be seen in Figure 6.7. This time, the iteration terminated after 162 iterations and once again one can see the strong smoothing effects of the two calculation options. As we will later see in Table 6.1, of all the combinations of different reconstruction options, this one yields the third best result.

Lastly, we want to present some results of using the sparsity option, together with either the divergence-free or the wavelet embedding option. For this, we need to choose an α, see (5.2). A good choice turns out to be $\alpha = 10^{-3}$, which was used for computing all presented results. Figures 6.8, 6.9, 6.10 and 6.11 show the results of the different
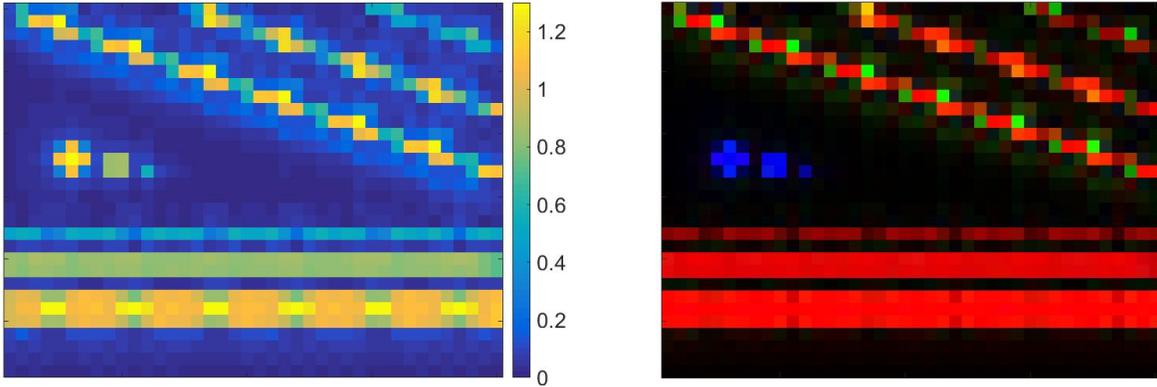
Figure 6.7: Result of the algorithm applied to the test problem ($\delta = 1\%$), using the weak divergence-free and the wavelet embedding option. Velocity norm MIP (left) and colour direction MIP (right).
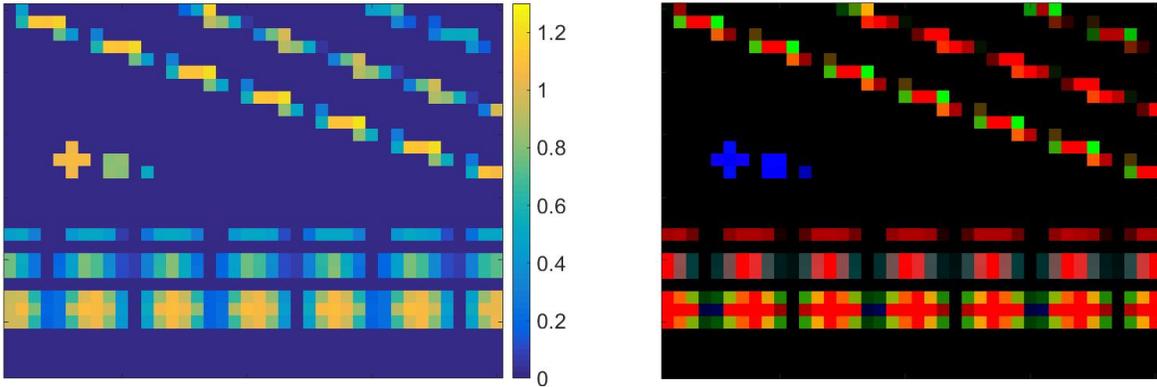

Figure 6.8: Result of the algorithm applied to the test problem ($\delta = 1\%$), using the sparsity option with $\alpha = 10^{-3}$. Velocity norm MIP (left) and colour direction MIP (right).

combinations, the iteration stopping after 71, 100, 99 and 138 steps, respectively. Note that all iterations involving the sparsity option were terminated before satisfying the discrepancy principle, since the residual stopped decreasing monotonously but rather started to oscillate around a certain value.

Comparing Figure 6.6 and Figure 6.9, we see that the edges are now more sharply reconstructed, although the result itself does not look much better then when using no additional options at all. This also holds true for only using the sparsity option alone, see Figure 6.8. However, Figure 6.10 strongly shows the advantages of combining the divergence-free and the sparsity options. The initial signal ρ_0 only slightly affects the reconstructed solution and the sparsity option removes some of the smearing introduced by the divergence-free option, producing very nice results. Even better results are obtained when combining all three calculation options, which can be seen Figure 6.11, most notably at the plus shaped vessel in the centre of the figure and at the diagonal vessels, especially at the middle one of the three, which, despite the crude discretization
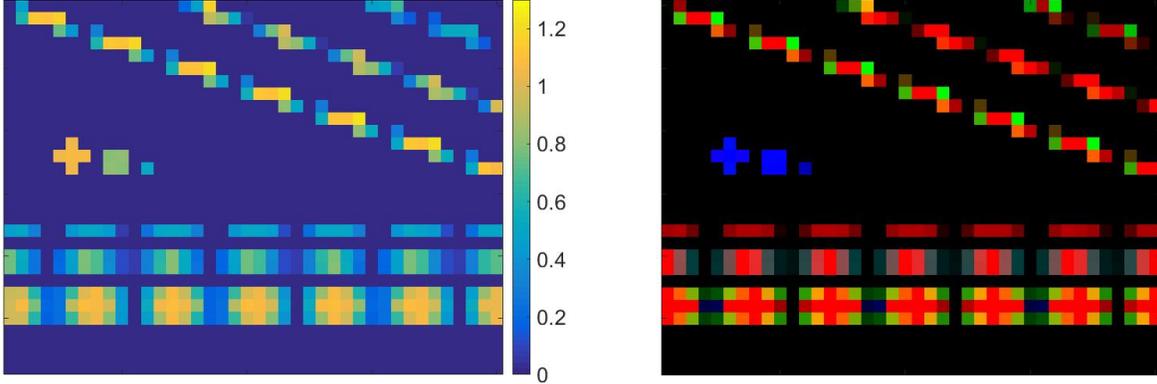
Figure 6.9: Result of the algorithm applied to the test problem ($\delta = 1\%$), using the wavelet embedding and the sparsity option with $\alpha = 10^{-3}$. Velocity norm MIP (left) and colour direction MIP (right).
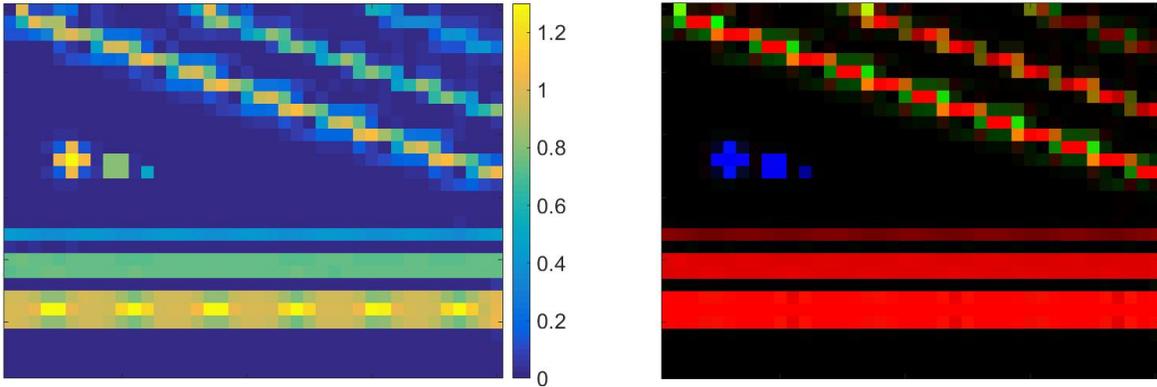

Figure 6.10: Result of the algorithm applied to the test problem ($\delta = 1\%$), using the divergence-free and the sparsity option with $\alpha = 10^{-3}$. Velocity norm MIP and colour direction MIP.

and the smoothing introduced by the divergence-free and the wavelet embedding option, is reconstructed rather nicely.

To conclude this section, in Table 6.1 we have collected some important information about the results presented above. The first three columns contain information about the used calculation options, the fourth column contains the iteration index k_* at which the algorithm was terminated and in the fifth column, the error between approximated and true solution (denoted by $(\vec{v}^\dagger, \vec{\rho}_0^\dagger)$) is given. Here we have used the standard Euclidean ℓ_2-norm for measuring the error, in order to allow for a fair comparison between those results which were achieved using the wavelet embedding option and the ones not using it. Once again it can be seen that the best results are obtained using all three calculation options in the reconstruction algorithm.
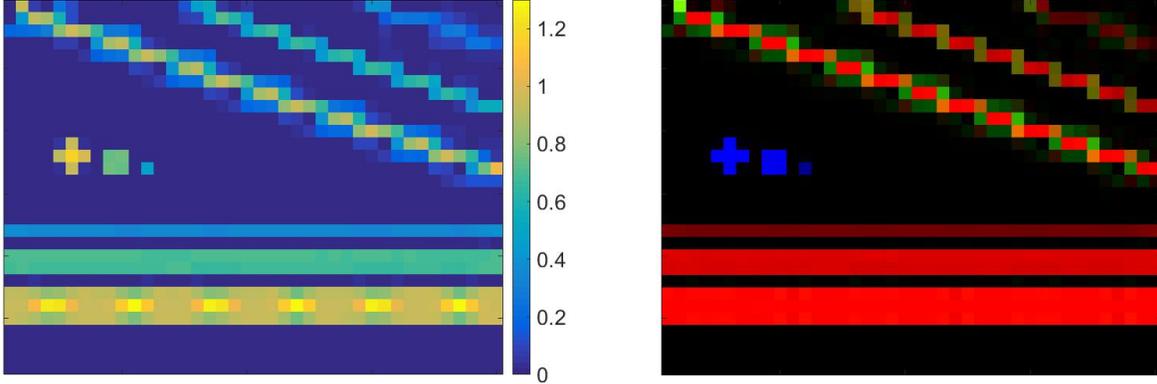
Figure 6.11: Result of the algorithm applied to the test problem ($\delta = 1\%$), using the divergence-free, the wavelet embedding and the sparsity option with $\alpha = 10^{-3}$. Velocity norm MIP and colour direction MIP.

	div-free	wavelets	sparsity	k_*	$\left\| (\vec{v}_{k_*}^{\delta}, \vec{\rho}_{0,k_*}^{\delta}) - (\vec{v}^{\dagger}, \vec{\rho}_0^{\dagger}) \right\|_{\ell_2}$
Figure 6.3	no	no	no	90	16.9658
Figure 6.4	yes	no	no	126	9.2904
Figure 6.6	no	yes	no	121	16.4324
Figure 6.7	yes	yes	no	162	8.8878
Figure 6.8	no	no	yes	71	17.179
Figure 6.9	no	yes	yes	100	16.7834
Figure 6.10	yes	no	yes	99	5.6577
Figure 6.11	yes	yes	yes	138	5.5245

Table 6.1: Comparison of the results of the reconstruction algorithm applied to the test problem ($\delta = 1\%$), achieved using combinations of the different computation options.

6.2 Natural Stimulation Data Set

In this subsection, we test the applicability of our algorithm to real-world data sets. For this, we use a publicly available natural stimulation dynamic EPI data set obtained on a 7.0 T MRI scanner [12]. Subjects were listening to an audio version of a movie. The data set includes eight 15 minutes long segments for each subject, of which the first 20 seconds of the second one were used for analysis. The transversal slices covered most of the frontal and occipital cortex and the regions in between. Data was sampled with a pulse repetition time (TR) of 2 s and an isotropic spatial resolution of 1.4 mm. The data set also contains time-of-flight angiography images of about the same coverage as the EPI data, as well as pulse oximetry data.

We want to apply our algorithm to different subjects of this data set, for which Voss et al. have already tried to reconstruct the pulse wave velocity using their multiple regression approach [34]. In order to do so, we need to adapt our stopping rule, since the discrepancy principle defined in (5.4) relies essentially on knowledge of δ, which, as

is usually the case in real-world situations, is not given explicitly. However, an estimate of the data error can be made by looking at the background voxels, i.e., those voxels which are known to lie outside the brain and which therefore should have value 0 if no noise were present. The corresponding calculations suggest that for our data set, the relative data error is approximately 2 - 3 %. This estimate, combined with the discrepancy principle and a check for monotonous decrease of the residual suggests to stop the iteration after 15 - 25 iterations.

The upper two figures of Figure 6.12 show the results of applying our algorithm with the above described changes to subject 16 of the real world data set. Here we used the divergence free, the wavelet embedding and the sparsity option, this time with $\alpha = 10^4$ and the computation was stopped after 20 iterations. As before, the upper left image shows the velocity norm MIP over the z-axis and the upper right image the colour direction MIP of the reconstructed velocity $\vec{v}$. One can clearly see the location of the major blood vessels and arteries, as well as their orientation, even though the expected PWV is severely underestimated (see below). The norm of the velocity vector field has maximum 0.1495, mean $\mu = 0.0011$ and standard deviation $\sigma = 0.0034$. In order to generate the figure, a slight scaling was introduced, cutting all values in the MIP which are above $\mu + 5\sigma$. The colour direction MIP was slightly brightened in order to enhance visibility.

For comparison of our proposed and the previously used method, the lower two figures of Figure 6.12 depict the results of the multiple regression approach, applied to the same subject 16 of the data set, using a 0.05 Hz cut-off filtering preprocessing step and all 15 minutes of the second segment of the data set. Again, a $\mu + 5\sigma$ scaling and a slight brightening of the colour direction MIP were used for better visibility. One can see that while the previously used algorithm mainly yields estimates of the PWV in the main arteries and blood vessels, our new algorithm is able to resolve finer structures as well, using only a fraction of the data. Note that the previously used algorithm applied to 20 seconds of data would yield a result hardly distinguishable from white noise.

For certain subjects of the data set, the regression approach of [34] yields very unsatisfactory results. This appears to be due to the heart rate of the subjects having an unfavourable frequency and the data error being higher in those cases. The results of the regression approach and our proposed algorithm applied to one of those subjects, subject 2 of the data set, can be seen in Figure 6.13. The differences are quite obvious and can be attributed to two main reasons. Firstly, our algorithm works with much less data then the regression approach and secondly, by stopping the iteration after a certain amount of steps, we get a regularizing effect. Consequently, the effects of data error can partly be compensated and therefore, better reconstructions are obtained.

Please also note, that the calculated velocities using the real-world data set, which are around 10^{-5} m/s, differ by orders of magnitude from the expected pulse wave velocity, which can exceed 10 m/s. One reason for this is the high amount of noise in the data, which can only be partially controlled using appropriate filters. Another reason is the low spatiotemporal, in particular the low temporal resolution of the MRI data; Condition 6.4 is far from being fulfilled and hence the algorithm, after a certain amount of iterations, is no longer able to improve the approximation, which leads to
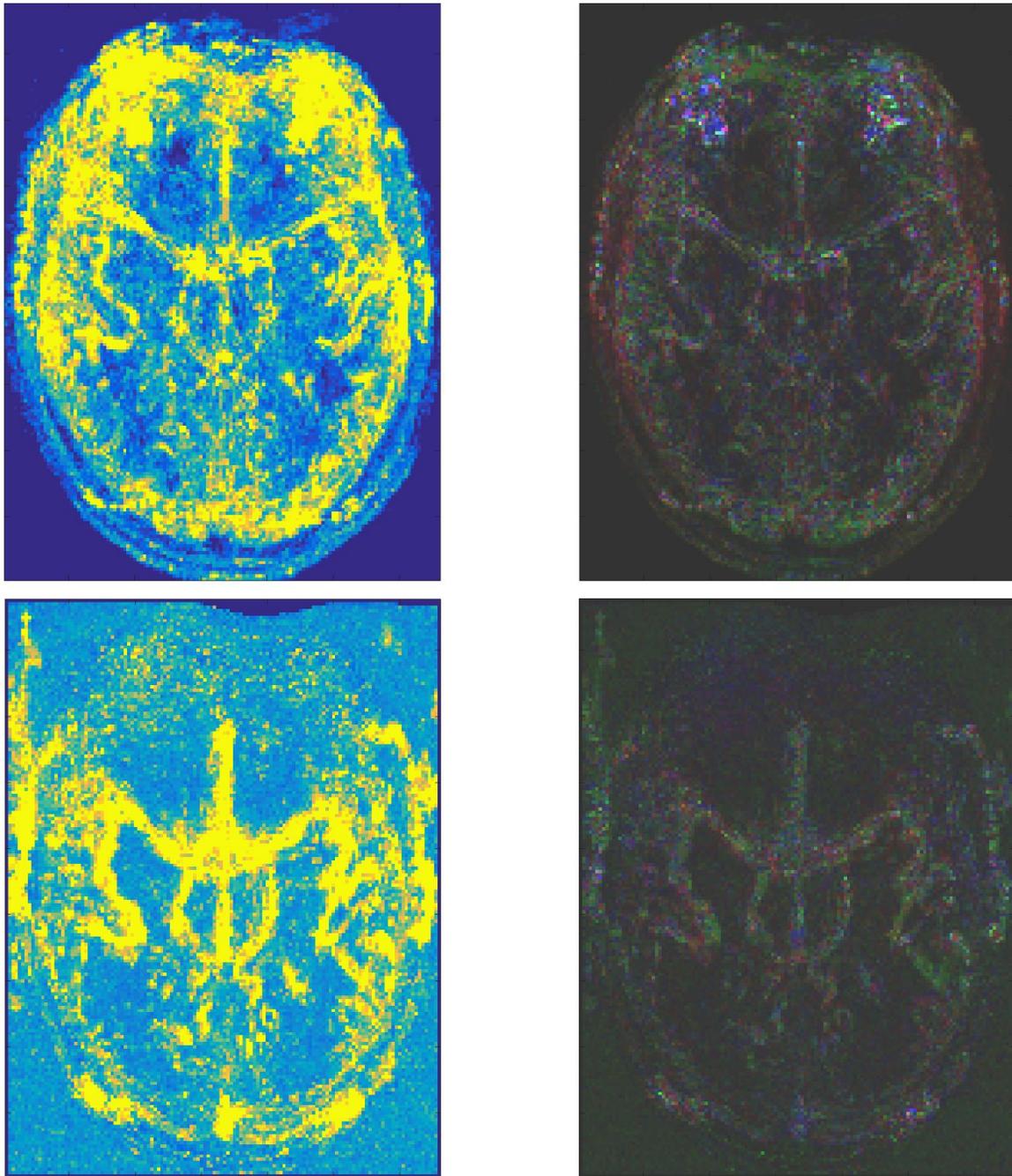
Figure 6.12: Results of our proposed algorithm (upper two figures, 20 seconds of data) and the regression-based algorithm (lower two figures, 15 minutes of data), applied to subject 16 of the data set. Velocity norm MIPs (left) and colour direction MIPs (right).

underestimated velocities. The same problem has already been observed in [33], where the velocities were severely underestimated as well. This phenomenon, although most clearly understandable from the point of view of the finite difference approximation and
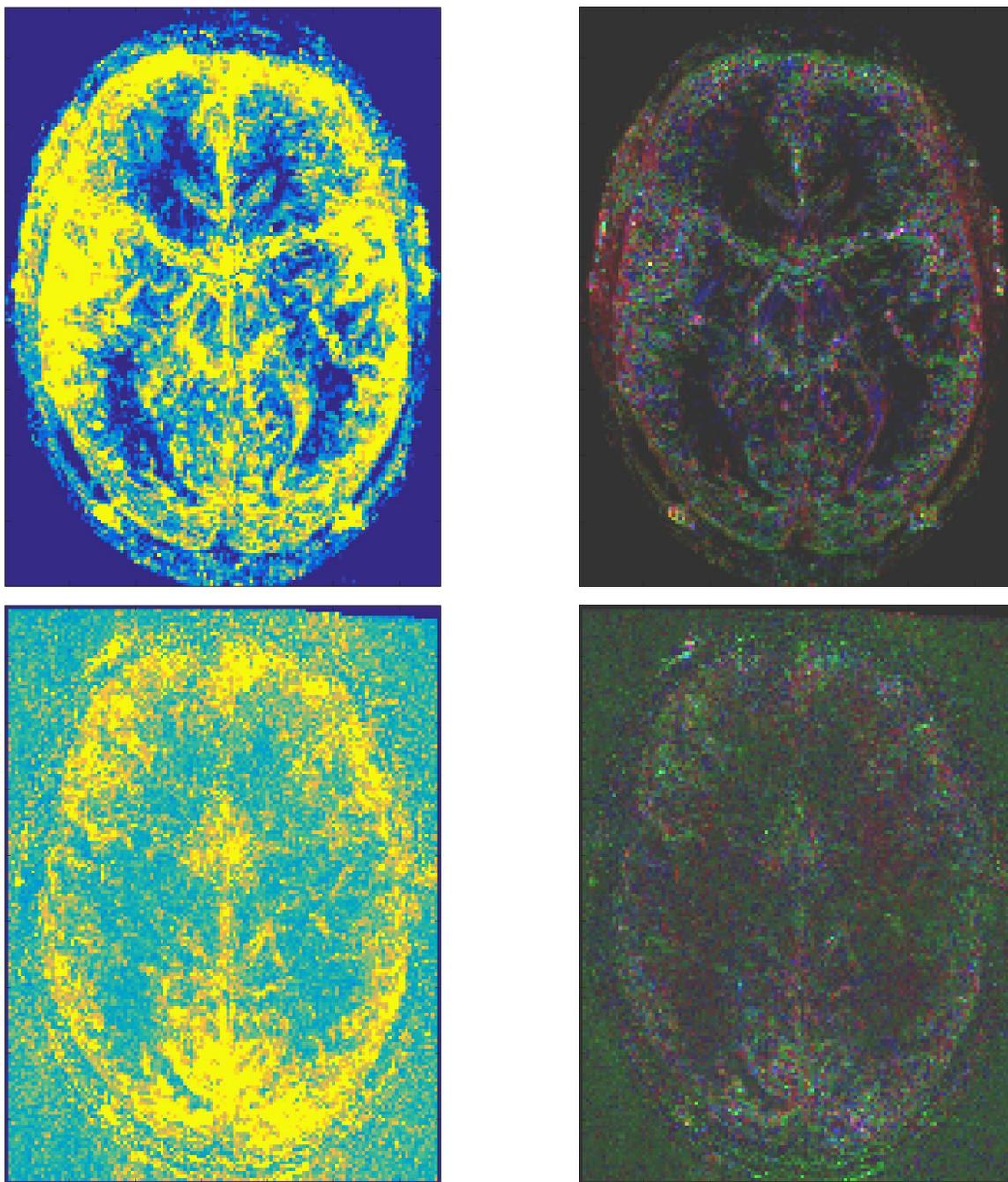
Figure 6.13: Results of our proposed algorithm (upper two figures, 20 seconds of data) and the regression-based algorithm (lower two figures, 15 minutes of data), applied to subject 2 of the data set. Velocity norm MIPs (left) and colour direction MIPs (right).

condition (6.4), is also quite likely to appear, in one or another form, when using other discretization techniques as well.

7 Conclusion and Outlook

The advection model was intended to model travelling pulse waves, but there might be other travelling disturbances along blood vessels or nerves in the brain. For example, it has been observed that endothelially mediated vasodilation related to functional brain activation travels along small blood vessels [15]; although on spatial scales much smaller than used here, high-resolution dynamic MRI data that would be potentially able to resolve this phenomenon with MRAI already exists [39]. Further, due to its sensitivity to pulsatile components of the signal and due to dramatic advances in dynamic MRI data acquisition [22], MRAI might have future potential to contribute to the modelling of the cerebrovascular system and to serve as a biomarker for cerebrovascular disease. It should also be noted that the methods described herein are quite general and could in principle be applied to spatiotemporal dynamics across a wide range of dynamic imaging applications in medicine and other fields (with an adaptation of the PDE model to the specific situation).

Concerning our proposed velocity estimation algorithm, the numerical simulation results of Section 6.1 clearly demonstrate the good reconstruction abilities of our method, especially when used with a suitable combination of the currently available options (weak divergence-free, sparsity, wavelets). This points to an advantage over the regression approach of Voss et al., namely the high flexibility of our approach. Considering the parameter estimation problem of MRAI in the framework of inverse problems, a vast array of techniques becomes available, leading to improved results. While the regression based method is more or less inflexible, our proposed approach can easily be adapted to include different or newly developed reconstruction options.

Another advantage of our algorithm is its ability to produce appealing results with only a small amount of data. Where the multiple regression approach requires at least a couple of minutes of measurements, our algorithm, as we have seen in Section 6.2, can produce nice qualitative results from only a couple of seconds of measurements. This might prove advantageous in practice, where long scan times often need to be avoided.

Although working on numerical phantom simulations, when applied to the real-world data sets, both the regression based approach and our proposed reconstruction method produce qualitative results only. As mentioned above, the most important reasons for this is are the high amount of data error and the low spatiotemporal resolution of the data when compared to the expected magnitude of the PWV. Hence, in order to achieve better results, MRI data with higher resolutions and less noise need to be used in our algorithm. One possible way towards this would be to use advanced imaging methods such as multiband EPI [22], where a whole stack of slices is acquired in a time that normally allows only for the acquisition of a single slice, which leads to a much higher spatiotemporal resolution.

As the way towards quantitative results leads through higher resolution data sets, in the future our proposed algorithm will need to be modified in order to be capable to deal with the resulting huge data sets. This quite naturally leads into the realms of modern, highly efficient space-time methods, high-performance and parallel computing - three areas which the authors are planning to combine and apply to MRAI in future works in

order to improve it even further, with a hopefully much faster CG-based reconstruction method already under development.

8 Support

References

[1] R. Aaslid. *Transcranial Doppler Sonography*. Springer-Verlag, Wien; New York, 1986.

[2] G. A. Bateman. Pulse-wave encephalopathy: A comparative study of the hydrodynamics of leukoaraiosis and normal-pressure hydrocephalus. *Neuroradiology*, 44(9):740–8, 2002.

[3] A. Beck and M. Teboulle. A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems. *SIAM J. Imaging Sci.*, 2(1):183–202, 2009.

[4] M. S. Dagli, J. E. Ingeholm, and J. V. Haxby. Localization of cardiac-induced signal change in fMRI. *NeuroImage*, 9(4):407–415, 1999.

[5] I. Daubechies. *Ten Lectures on Wavelets*. Society for Industrial and Applied Mathematics, 1992.

[6] R. J. DiPerna and P. L. Lions. Ordinary differential equations, transport theory and Sobolev spaces. *Invent. Math.*, 98(3):511–547, 1989.

[7] W. A. Edelstein, G. H. Glover, C. J. Hardy, and R. W. Redington. The intrinsic signal-to-noise ratio in NMR imaging. *Magnetic Resonance in Medicine*, 3(4):604–618, 1986.

[8] P. Elter. *Methoden und Systeme zur nichtinvasiven, kontinuierlichen und belastungsfreien Blutdruckmessung*. Thesis, 2001.

[9] S. Engblom and D. Lukarski. FSPARSE. `http://user.it.uu.se/~stefane/freeware.html`, 2014.

[10] H. W. Engl, M. Hanke, and A. Neubauer. *Regularization of inverse problems*. Dordrecht: Kluwer Academic Publishers, 1996.

[11] A. Frydrychowicz, C. J. Francois, and P. A. Turski. Four-dimensional phase contrast magnetic resonance angiography: Potential clinical applications. *European Journal of Radiology*, 80(1):24–35, 2011.

[12] M. Hanke, F. J. Baumgartner, P. Ibe, F. R. Kaule, S. Pollmann, O. Speck, W. Zinke, and J. Stadler. A high-resolution 7-Tesla fMRI dataset from complex natural stimulation with an audio movie. *Scientific Data*, 1:140003:1–18, 2014.

[13] M. C. Henry Feugeas, G. De Marco, I. I. Peretti, S. Godon-Hardy, D. Fredy, and E. S. Claeys. Age-related cerebral white matter changes and pulse-wave encephalopathy: observations with three-dimensional MRI. *Magnetic Resonance Imaging*, 23(9):929–37, 2005.

[14] L. H. G. Henskens, A. A. Kroon, R. J. van Oostenbrugge, E. H. B. M. Gronenschild, M. M. J. J. Fuss-Lejeune, P. A. M. Hofman, J. Lodder, and P. W. de Leeuw. Increased aortic pulse wave velocity is associated with silent cerebral small-vessel disease in hypertensive patients. *Hypertension*, 52(6):1120–U72, 2008.

[15] E. M. C. Hillman. Coupling mechanism and significance of the BOLD Signal: A status report. *Annual Review of Neuroscience*, 37:161–181, 2014.

[16] B. Kaltenbacher, A. Neubauer, and O. Scherzer. *Iterative regularization methods for nonlinear ill-posed problems.* Berlin: de Gruyter, 2008.

[17] D. J. Korteweg. Über die Fortpflanzungsgeschwindigkeit des Schalles in Elastischen Röhren. *Annalen der Physik*, 241(12):525–542, 1878.

[18] S. Laurent, J. Cockcroft, L. Van Bortel, P. Boutouyrie, C. Giannattasio, D. Hayoz, B. Pannier, C. Vlachopoulos, I. Wilkinson, H. Struijker-Boudier, and European Network Non-invasive. Expert consensus document on arterial stiffness: Methodological issues and clinical applications. *European Heart Journal*, 27(21):2588–2605, 2006.

[19] J. K. J. Li. *Dynamics of the Vascular System.* Series on Bioengineering and Biomedical Engineering. World Scientific, River Edge, N.J., 2004.

[20] D. A. Lorenz, P. Maass, and P. Q. Muoi. Gradient descent for Tikhonov functionals with sparsity constraints: theory and numerical comparison of step size rules. *Electron. Trans. Numer. Anal.*, 39:437–463, 2012.

[21] D. W. McRobbie, E. A. Moore, and M. J. Graves. *MRI from Picture to Proton.* University Printing House, Cambridge University Press, Cambridge; New York, 3rd edition, 2016.

[22] S. Moeller, E. Yacoub, C. A. Olman, E. Auerbach, J. Strupp, N. Harel, and K. Ugurbil. Multiband multislice GE-EPI at 7 Tesla, with 16-fold acceleration using partial parallel imaging with application to high spatial and temporal whole-brain fMRI. *Magnetic Resonance in Medicine*, 63(5):1144–1153, 2010.

[23] A. I. Moens. *Over de voortplantingssnelheid van den pols [On the speed of propagation of the pulse].* Thesis, 1877.

[24] Yurii Nesterov. A method of solving a convex programming problem with convergence rate $O(1/k^2)$. *Soviet Mathematics Doklady*, 27(2):372–376, 1983.

[25] S. Ogawa, T. M. Lee, A. R. Kay, and D. W. Tank. Brain magnetic-resonance-imaging with contrast dependent on blood oxygenation. *Proceedings of the National Academy of Sciences of the United States of America*, 87(24):9868–9872, 1990.

[26] S. W. Rabkin. Arterial stiffness: Detection and consequences in cognitive impairment and dementia of the elderly. *Journal of Alzheimers Disease*, 32(3):541–549, 2012.

[27] R. Ramlau. Regularization properties of Tikhonov regularization with sparsity constraints. *Electron. Trans. Numer. Anal.*, 30:54–74, 2008.

[28] R. Ramlau and G. Teschke. A Tikhonov-based projection iteration for nonlinear Ill-posed problems with sparsity constraints. *Numer. Math.*, 104(2):177–203, 2006.

[29] K. Sagawa, R. K. Lie, and J. Schaefer. Translation of Otto Frank's paper "Die Grundform des Arteriellen Pulses" Zeitschrift für Biologie 37: 483-526 (1899). *Jounral of Molecular and Cellular Cardiology*, 22(3):253–4, 1990.

[30] R. Sladky, K. J. Friston, J. Troestl, R. Cunnington, E. Moser, and C. Windischberger. Slice-time effects and their correction in functional MRI. *NeuroImage*, 58:588–594, 2011.

[31] M. K. Stehling, R. Turner, and P. Mansfield. Echo-planar imaging - magnetic-resonance-imaging in a fraction of a second. *Science*, 254(5028):43–50, 1991.

[32] Y. J. Tong, L. M. Hocke, and B. D. Frederick. Short Repetition Time Multiband Echo-Planar Imaging with Simultaneous Pulse Recording Allows Dynamic Imaging of the Cardiac Pulsation Signal. *Magnetic Resonance in Medicine*, 72(5):1268–1276, 2014.

[33] H. U. Voss, J. P. Dyke, K. Tabelow, N. D. Schiff, and D. J. Ballon. Mapping cerebrovascular dynamics with magnetic resonance advection imaging (MRAI): modeling challenges and estimation bias. *Meeting of the Society for Neuroscience*, 2015.

[34] H. U. Voss, J. P. Dyke, K. Tabelow, N. D. Schiff, and D. J. Ballon. Magnetic resonance advection imaging (MRAI) of cerebrovascular pulse dynamics. *Journal of Cerebral Blood Flow and Metabolism*, in press.

[35] H. U. Voss and N. D. Schiff. Searching for conservation laws in brain dynamics-BOLD flux and source imaging. *Entropy*, 16(7):3689–3709, 2014.

[36] E. A. H. Warnert, K. Murphy, J. E. Hall, and R. G. Wise. Noninvasive assessment of arterial compliance of human cerebral arteries with short inversion time arterial spin labeling. *Journal of Cerebral Blood Flow and Metabolism*, 35(3):461–468, 2015.

[37] E. C. Wong, R. B. Buxton, and L. R. Frank. Quantitative perfusion imaging using arterial spin labeling. *Neuroimaging Clinics of North America*, 9(2):333–342, 1999.

[38] L. Yan, C. Y. Liu, R. X. Smith, M. Jog, M.l Langham, K. Krasileva, Y. Chen, J. M. Ringman, and D. J. J. Wang. Assessing intracranial vascular compliance using dynamic arterial spin labeling. *NeuroImage*, 124, Part A:433–441, 2016.

[39] X. Yu, Y. He, M. Wang, H. Merkle, S. J. Dodd, A. C. Silva, and A. P. Koretsky. Sensory and optogenetically driven single-vessel fMRI. *Nature Methods*, 13(4):337–40, 2016.

[40] M. Zamir. *The Physics of Pulsatile Flow*. Biological physics series. AIP Press; Springer, New York, 2000.

Technical Reports of the Doctoral Program

"Computational Mathematics"

2016

2016-01 P. Gangl, U. Langer: *A Local Mesh Modification Strategy for Interface Problems with Application to Shape and Topology Optimization* November 2016. Eds.: B. Jüttler, R. Ramlau

2016-02 C. Hofer: *Parallelization of Continuous and Discontinuous Galerkin Dual-Primal Isogeometric Tearing and Interconnecting Methods* November 2016. Eds.: U. Langer, W. Zulehner

2016-03 C. Hofer: *Analysis of Discontinuous Galerkin Dual-Primal Isogeometric Tearing and Interconnecting Methods* November 2016. Eds.: U. Langer, B. Jüttler

2016-04 A. Seiler, B. Jüttler: *Adaptive Numerical Quadrature for the Isogeometric Discontinuous Galerkin Method* November 2016. Eds.: U. Langer, J. Schicho

2016-05 S. Hubmer, A. Neubauer, R. Ramlau, H. U. Voss: *On the Parameter Estimation Problem of Magnetic Resonance Advection Imaging* December 2016. Eds.: B. Jüttler, U. Langer

2015

2015-01 G. Grasegger, A. Lastra, J. Rafael Sendra, F. Winkler: *A Solution Method for Autonomous First-Order Algebraic Partial Differential Equations in Several Variables* January 2015. Eds.: U. Langer, J. Schicho

2015-02 P. Gangl, U. Langer, A. Laurain, H. Meftahi, K. Sturm: *Shape Optimization of an Electric Motor Subject to Nonlinear Magnetostatics* January 2015. Eds.: B. Jüttler, R. Ramlau

2015-03 C. Fürst, G. Landsmann: *Computation of Dimension in Filtered Free Modules by Gröbner Reduction* May 2015. Eds.: P. Paule, F. Winkler

2015-04 A. Mantzaflaris, H. Rahkooy, Z. Zafeirakopoulos: *Efficient Computation of Multiplicity and Directional Multiplicity of an Isolated Point* July 2015. Eds.: B. Buchberger, J. Schicho

2015-05 P. Gangl, S. Amstutz, U. Langer: *Topology Optimization of Electric Motor Using Topological Derivative for Nonlinear Magnetostatics* July 2015. Eds.: B. Jüttler, R. Ramlau

2015-06 A. Maletzky: *Exploring Reduction Ring Theory in Theorema* August 2015. Eds.: B. Buchberger, W. Schreiner

Doctoral Program

"Computational Mathematics"