Data Assimilation for Full 4D PC-MRI Measurements: Physics-Based Denoising and Interpolation

N. H. L. C. de Hoon,[1] A.C. Jalba,[2] E.S. Farag,[3] P. van Ooij,[3] A.J. Nederveen,[3] E. Eisemann[1] and A. Vilanova[1,2]

[1]Department of Intelligent Systems, Delft University of Technology, Delft, The Netherlands
nielsdehoon@gmail.com, e.eisemann@tudelft.nl, A.Vilanova@tue.nl
[2]Department of Mathematics and Computer Science, Eindhoven University of Technology, Eindhoven, The Netherlands
A.C.Jalba@tue.nl
[3]Department of Radiology and Nuclear Medicine, University of Amsterdam, Amsterdam, The Netherlands
e.s.farag@amc.uva.nl, pim.vanooij@gmail.com, a.j.nederveen@amc.uva.nl

Abstract

Phase-Contrast Magnetic Resonance Imaging (PC-MRI) surpasses all other imaging methods in quality and completeness for measuring time-varying volumetric blood flows and has shown potential to improve both diagnosis and risk assessment of cardiovascular diseases. However, like any measurement of physical phenomena, the data are prone to noise, artefacts and has a limited resolution. Therefore, PC-MRI data itself do not fulfil physics fluid laws making it difficult to distinguish important flow features. For data analysis, physically plausible and high-resolution data are required. Computational fluid dynamics provides high-resolution physically plausible flows. However, the flow is inherently coupled to the underlying anatomy and boundary conditions, which are difficult or sometimes even impossible to adequately model with current techniques. We present a novel methodology using data assimilation techniques for PC-MRI noise and artefact removal, generating physically plausible flow close to the measured data. It also allows us to increase the spatial and temporal resolution. To avoid sensitivity to the anatomical model, we consider and update the full 3D velocity field. We demonstrate our approach using phantom data with various amounts of induced noise and show that we can improve the data while preserving important flow features, without the need of a highly detailed model of the anatomy.

Keywords: Flow Visualization, Visualization, Medical Imaging, Visualization, Natural Phenomena, Modelling

ACM CCS: • Computing methodologies → Data assimilation; Physical simulation; • Applied computing → Imaging; Life and medical sciences

1. Introduction

Cardiovascular diseases (CVDs) are globally the main cause of mortality and morbidity [MBG*15]. Blood flow plays a decisive role in their occurrence and progression [HBB*10, MFK*12]. Therefore, acquiring knowledge of the blood flow is of key importance. Although various techniques exist to measure the blood flow of a patient, phase-contrast enhanced magnetic-resonance imaging (PC-MRI) provides the most detailed and complete information. PC-MRI enables measuring time-resolved volumetric vector fields that represent the blood flow in three dimensions over time [MFK*12]. This flow imaging information is complex, and often visually analysed using flow visualization techniques such as integral lines. The

most important aspect of PC-MRI data is that it is *patient specific*, that is, the measured flow describes the actual flow in the patient. However, the presence of noise and artefacts due to limitations of the scanning technique [MFK*12] make the data not physically plausible, and, therefore, the direct visualization of the PC-MRI data generates visualizations that are often inaccurate and difficult to analye. For instance, partial volume effects can occur due to the limitations of the measuring resolution, which are especially noticeable near the vessel wall. For many blood-flow analyses and visualizations, the flow is assumed to be *divergence-free* [LKTW12, OUT*15] and this feature is relevant for the diagnosis. The assumption is that the only sources and sinks of flow should be at the beginning and end of the vessel. Another physical property is
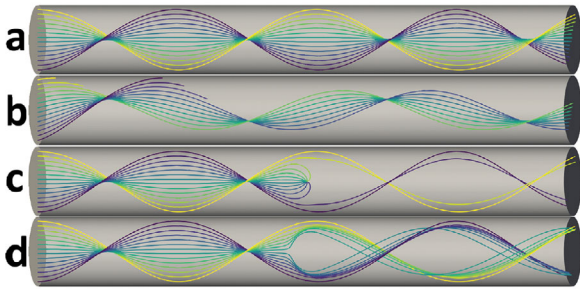
Figure 1: *Inconsistencies in the vector-field data can have a big impact on flow visualization. Respectively, from top to bottom, (a) shows the original data, in image (b) the vector field has a bias in one direction causing the flow to leave the domain. Images (c) and (d) suffer from a single source of divergence, more precisely (c) depicts a sink and (d) a source. Note that the number of parallel integration lines decreases in all three scenarios compared to the baseline as shown by image (a).*

that no flow should leave the blood vessel, that is, the vessel does not leak. Both divergence and flow that leaves the segmentation can occur when using PC-MRI and have an impact on flow visualizations, for example, using integration-based flow visualization. Figure 1 illustrates these effects on the visualization of these measurement artefacts. Here streamlines are shown for the same information with a specific artefact, that is, flow leaving the segmentation and divergence in the form of a sink and source. Each of these artefacts change the flow visualization substantially, and as such, the flow could be interpreted incorrectly. Moreover, for quantitative analysis, these artefacts must be taken into account with care. For example, for quantitative particle tracing, where particles are emitted and counted to determine, for example, the percentage of blood volume that is ejected by the heart in one heart cycle and, thus, leaves the heart adequately. In this case, if particles are leaving the domain or get stuck in a sink, such percentages will be negatively affected and therefore less reliable. Furthermore, the spatial and temporal resolutions that can be achieved with PC-MRI are limited [MFK*12], which has a clear impact on the analyses and visualization, when determining, for example, the positions of vortex cores.

Computational Fluid Dynamics (CFD), on the other hand, can be used to obtain high-resolution, physically correct flow data. However, most of these simulation models make assumptions and require a precise modelling of the anatomy [MNvTK*15]. In some cases, it can be very challenging, or even impossible to model the actual anatomy of the patient with current techniques, for example, when certain details are small in comparison to the imaging resolution, which is the case, for instance, for the heart valves [TBE*16].

Data assimilation can be used to estimate, interpolate or extrapolate the true flow velocity field from the PC-MRI data. Data assimilation combines theory – in this case a physics-based model – with real-world measurements. Data assimilation is a common approach in many fields, such as geoscience, where it is used, for instance, in numerical weather prediction [DUS*11, LBC*15,

PBC*16]. Goals for data assimilation include determining good initial conditions for the model, interpolation of sparse observation data and an improved estimation of the true state of the system. The concept of data assimilation was also previously applied to PC-MRI data. However, these methods are focusing on the improvement of in–out flow boundary conditions of the simulation models. The results remain highly dependent on the model of the anatomy, such as the vessel wall segmentation. Furthermore, flow patterns that are introduced due to anatomical features missing in the segmentation cannot always be reconstructed by these methods. This limits its applicability, despite producing physically plausible flow. Furthermore, these methods also suffer from high computational costs that are usually connected to using finite element models.

In this paper, we present a data-assimilation methodology that uses the full 3D PC-MRI flow data. It updates the full 3D PC-MRI flow data to be physically plausible, with limited sensitivity to the accuracy of the given anatomical model and boundary conditions. We aim at obtaining flow data that fulfils the following goals:

- *Patient-specific*: as close as possible to the original measured data when the measured data are reliable.
- *Physically-plausible*: divergence-free and no flow leaking through the vessel walls.
- *High-resolution*: enough data points to convey useful analysis.

To this end, we define an optimization process that minimizes the difference between the measured data and a model based on the physical flow properties. A quasi-Newton method [Noc80, LN89] is used to find the velocity field produced by the model that best corresponds with the measured data. To obtain the gradients necessary for the quasi-Newton method, we apply automatic differentiation [Hog14] on the model code. Furthermore, the proposed methodology can also be used to increase the resolution, both spatially and temporally, in accordance with the Navier–Stokes equations and the measured data.

Our solution also allows including the concept of uncertainty. That is, local areas with reliable measurements have more influence on the resulting flow field, while the flow model has more influence in the areas where the measurements are less reliable.

In this paper, we firstly discuss related research that was conducted. Subsequently, we describe the type of data that were used in this paper. In the following sections, we present our methodology, and finally we evaluate and show several examples as results. We conclude the paper and provide potential future work.

2. Related work

Our method is related to different research fields that we will discuss in this section. Inspired by optimal control methods, we will use similar techniques to steer the flow towards the measurement. Furthermore, our goal is in some aspects similar to PC-MRI denoising and data assimilation techniques in general.

2.1. Simulation control

In computer graphics, simulated fluid control is an active research field. The main objective is to provide artists with tools and methods to control the fluid simulation to achieve a desired animation of the fluid. For instance, by allowing the artist to use reference images of the flow density [GITH14], control forces [HK04, PM17], control particles [TKPR06] or, more recently, data-driven techniques [CT17]. For animation, the appearance of the fluid is most important to the artist, that is, the fluid surface in the case of liquids and densities in the case of smoke. Therefore, these control methods try to control the fluid surface or density. The method by McNamara *et al.* [MTPS04] uses iterative minimization to match a given target density or surface. A variational approach was introduced by Nielsen *et al.* [NC10], in which the simulation follows the low frequency information present in the target flow. In our work, we use a similar approach, such that the resulting flow best matches the PC-MRI measured velocity field. However, as we do not focus on the visual representation of the flow directly, but rather on the underlying velocity field to produce more reliable visualizations, the existing methods cannot readily be applied to our scenario.

A PC-MRI data coupling approach was presented by De Hoon *et al.* [HJEV16]. The method tries to match a target velocity field $\vec{u}_m$ by correcting a simulated field $\vec{u}_s$ using the difference $\vec{u}_d$ between the target and simulated velocity field. This difference velocity field is then simulated and added as a correction to compute the final velocity field $\vec{u}_{\text{new}} = \vec{u}_s + \text{sim}(\vec{u}_d)$. The term $\vec{u}_{\text{new}}$ is computed for all phases, and it is used to temporally interpolate the data by simulating forward and backward between successive phases. The intermediate fields are obtained by weighted summation of the two simulations. However, the produced velocity field could still substantially deviate from the target field because the simulation step is not guaranteed to be close to $\vec{u}_m$.

2.2. PC-MRI denoising

Multiple approaches exist to denoise PC-MRI data [SNGP93, Buo94, FA03, LKTW12, BGWK13, BLV*15, OUT*15, KBvP*16, SKP18]. Some methods include the finite difference method [SNGP93], which projects the data onto a divergence-free vector field. Another method uses RBF [BGWK13] to minimize the divergence approaching the measured data using a set of convolution and divergence-free radial basis functions. Similarly, Ong *et al.* [OUT*15] proposed the use of divergence-free wavelets (DFW). A comparison by Sereno *et al.* [SKP18], however, found that some divergence remained present in the data for the methods above. They reduce its divergence, but might deviate from the measured data more than necessary, even if some methods actively try to minimize the difference between the outcome and the noisy input data, for example, Bostan *et al.* [BLV*15]. These techniques, on the contrary, mainly focus on making the data divergence free, and often do not consider the dependency on the modelled anatomy, are not compliant with Navier–Stokes equations nor allow for interpolation. Furthermore, some of these methods rely on the Helmholtz–Hodge decomposition to make the data divergence free. This decomposition assumes that the input vector field is sufficiently smooth [BNPB13], which is not the case for most measured data.

2.3. Data assimilation

We want to use a physics-based model and measured data to obtain physically plausible patient-specific data. One relatively common approach is to use simulation models guided by PC-MRI data, that is, the boundary conditions are derived from measured PC-MRI data [LEVDGS19]. The amount of research using data assimilation is limited. Most techniques that exist in the medical domain (i.e. PC-MRI and ultrasound) use data assimilation to match simulation data with measured data. Many of these approaches focus on finding the optimal boundary conditions to match a target velocity field, for example, a measured velocity field [DPV12, FNE*18, IAWW18, GCM*18]. They often consider the whole measurement to find the 2D in/out-flow conditions that best match the measured data. A disadvantage of this approach is that it is strongly dependent on the accuracy of the modelled anatomy. That is, updating the 2D in/out-flow conditions cannot generate flow patterns if the anatomical segmentation does not have the full details. The approaches by Rispoli *et al.* [RNNC15] and Fathi *et al.* [FBB*18], on the other hand, use regularization to compute a least squares solution to match model and measured data based on the actual full 3D flow in the patient. However, they do not consider the local quality of the measured data nor temporal interpolation. Moreover, the goal is to achieve improvement of the model rather than improving the measured data itself. Furthermore, as the optimization is initialized with a velocity that is zero it is more likely that the optimization gets stuck in a local minimum.

Our data assimilation method combines a model with observations of a real world system in order to obtain a better estimate of the system. Our method is based on the commonly used data assimilation 3DVAR method. This method minimizes the weighted squared difference between the observation and model. To the best of our knowledge, we are the first to also consider local quality of the PC-MRI data to steer the local importance of the measurement versus the model in the data assimilation process.

3. Data

In this section, we introduce the type of data sets used in this paper and the voxel-wise uncertainty measurement that will be used by our data assimilation method. Although 4D PC-MRI is not yet applied in a clinical setting, research indicates that it provides interesting insights in the blood flow [HH08, KYM*93, MDM*05, HH08, MFK*12, BDC14, VFdH*18]. Typically patient data have a spatial and temporal resolution in the order of 1.0–3.0 mm and 20–50 ms, respectively. Especially the temporal resolution is coarse as healthy blood flow can reach speeds of 200 cm/s. Higher resolutions can be acquired at the cost of higher noise levels or by longer (impractical) scanning time [MFK*12].

Figure 2 shows a data set of the aorta of a healthy volunteer at the peak systole phase – the moment when the speed is highest in the aorta; the PC-MRI volumes were cropped to match the region of interest. The figure shows one of the three velocity components, the magnitude signal and the Signal-to-Noise Ratio (SNR) of the data. More details on the acquisition of PC-MRI data can be found, for example, in the work by Markl *et al.* [MFK*12] and Gasteiger [Gas14].
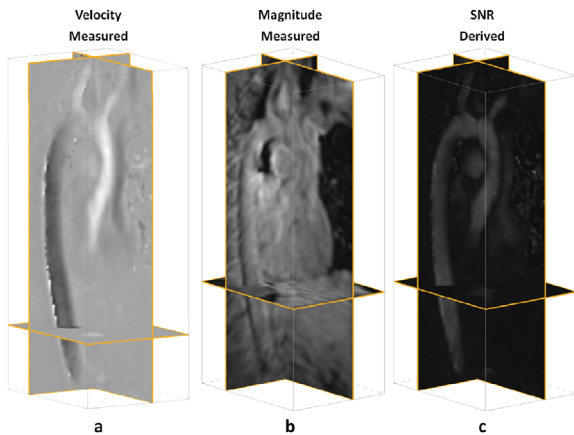
Figure 2: *Respectively, one of the three velocity components (inferior to superior) (a), the magnitude (b) and the corresponding signal to noise ratio (c), computed using the method by Friman* et al. *[FHH*10, FHH*11]. All at peak systole of a healthy volunteer.*
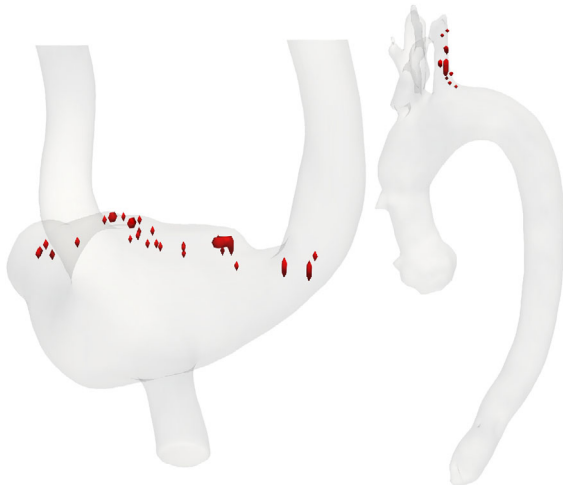

Figure 3: *An overview of the voxels that were marked as invalid during pre-processing.*

For the data sets we consider segmentations that are generated based on the temporal maximum intensity projection (tMIP) of the data. The tMIP represents the maximum velocity magnitude a voxel takes through all phases. The segmentation is then generated using marching cubes and smoothed/corrected using Blender. This segmentation does not require additional imaging of the patient, however, it is a coarse approximation. As our method does not rely on the accuracy of the mesh, such an approximation suffices, in contrast to CFD models which require a more accurate mesh to provide results similar to the measured flow.

Due to noise and artefacts present in measured data, every voxel in the data set can have a different reliability ϵ. For an estimate of the voxel-wise SNR, and thus reliability, caused by PC-MRI measurement noise, we employ the method presented by Friman

et al. [FHH*10, FHH*11]. This provides us with an initial value for ϵ at each voxel. We then pre-process the data using the method proposed by Yang *et al.* [YBK*96] to remove severe artefacts. More specifically, we mark all voxels as invalid if any vector component deviates more than 25% from the weighted average of its surrounding voxels. By using a deviation of 25%, the number of voxels that are marked invalid is small for our data sets, as indicated by Figure 3, if any voxels were marked as invalid at all. The resulting gaps are then filled with the average of the surrounding valid voxels, and the reliability ϵ of the corrected voxels is set to zero. Indicating that these voxel are not trustworthy. Furthermore, we also mark all voxels that are likely to be affected by partial volume effects with reliability zero. We update the SNR based on the distance to the segmentation boundaries. For example, we set the certainty to zero for all voxels that are crossed by the segmentation. This leads to a measurement reliability ϵ per voxel, where the value of ϵ is larger when the voxel is more trustworthy. Note that voxels with a zero reliability are only updated by the model considering the surrounding data, meaning that the measured data stored in such a voxel have no impact on the optimization process, yet the resulting value is based on the surrounding, more reliable voxels. Therefore, the overall influence of these voxels on the resulting flow is negligible when the amount of voxels with zero reliability is low.

4. Data assimilation for PC-MRI data

In this section, we present our minimization framework. To model the behaviour of fluid the Navier–Stokes equations for incompressible fluids should be solved as these describe the physical behaviour of a fluid:

$$\frac{\partial \vec{u}}{\partial t} = -\vec{u} \cdot \nabla \vec{u} - \frac{1}{\rho} \nabla p$$

$$\nabla \cdot \vec{u} = 0. \tag{1}$$

These equations consist in an advection term, a pressure term and a divergence condition. The divergence condition, $\nabla \cdot \vec{u} = 0$, is required to ensure the fluid is incompressible, meaning that no divergence is present in the fluid. Where the advection term, $-\vec{u} \cdot \nabla \vec{u}$, describes how the velocity is transport by the velocity field itself through time. The pressure term, $\frac{1}{\rho} \nabla p$, describes the amount of force applied by the fluid onto itself, hence, differences in pressure cause the fluid to flow. As the advection term considers the transport of fluid over time, this term is initially omitted. Note that for denoising PC-MRI data the advection term is commonly ommited [SNGP93, BGWK13, OUT*15, SKP18]. Moreover, in this work we assume blood to be inviscid, that is, we do not consider fluid viscosity. Blood is a non-Newtonian fluid in general, however, often it is assumed to be inviscid in high-velocity blood flow simulations. In special situations, for example, when the flow stops, the non-Newtonian properties should be considered. In our work, we focus on the inviscid models which is a common assumption regarding blood simulation [SKP18] and simplifies the assimilation of the model with measured data. Moreover, for temporal interpolation our method requires simulating fluid with a negative timestep and only inviscid flow is shown to be time-reversible [DOW08].

The basis of our method is a function $P(\vec{v}) : S \subset \mathbb{R}^3 \rightarrow \mathbb{R}^3$, with S a subset in $\mathbb{R}^3$, which maps a velocity field $\vec{v}$ to another velocity field $\vec{y}$. For denoising, P can be a divergence-free filter performing a pressure projection, which solves the pressure and divergence terms of Equation (1).

To make a given velocity field divergence-free (i.e. $\nabla \cdot \vec{u} = 0$), we use Chorin's projection method. This method is based on the Helmholtz–Hodge decomposition, which states that a smooth velocity field can be decomposed into a curl-free component and a divergence-free vector component. Given a velocity field, $\vec{u}$, for example, the measured data PC-MRI vector field, Chorin's projection method computes the curl-free component of the velocity field and subtracts it from the original velocity field to yield the divergence-free field. That is, first we compute the scalar pressure field p by solving the following Poisson equation

$$\nabla^2 p = \frac{\rho}{\Delta t} \nabla \cdot \vec{u}, \qquad (2)$$

where ρ is the fluid density and Δt the time step. As the fluid must remain incompressible and thus have no divergence, a correction of $\frac{1}{\rho} \nabla p$ is applied to enforce the incompressibility. In our implementation, we solve this Poisson equation using the pre-conditioned conjugate gradient algorithm with an incomplete Cholesky factorization as pre-conditioner. Once the pressure is known, the new velocity field $\vec{u}_n$ is computed using

$$\vec{u}_n = \vec{u} - \frac{\Delta t}{\rho} \nabla p. \qquad (3)$$

Note that if the given velocity field is non-smooth, the pressure solve may fail to correct for its non-zero divergence, hence, a sufficiently smooth input velocity field is necessary.

To handle irregular solid–fluid boundaries, for example, segmentations that do not align with the underlying grid, we use the variational solid boundaries approach by Batty et al. [BBB07]. Here, the segmentation is represented as the zero level of a signed-distance field, that is, for every voxel the smallest (signed) distance to the segmentation is stored. The gradient of such a field represents the direction to the nearest segmentation location. The flow between fluid cells is determined by a normalized weight, which is computed based on the level set and the portion of the cell face between the cells that lays within the segmentation. The pressure solve uses these weights to constrain the flow to stay within the segmentation. This means that the velocity component in the direction of the vessel wall is zero (i.e. Neumann-type boundary condition). Therefore, by construction, this condition ensures no flow will cross the vessel walls.

4.1. Minimization

Using the pressure solve function P described previously, we can find physically plausible velocity fields. However, these fields are not necessarily as close as possible to the measured data. Therefore, our goal is to find a suitable input for the function P such that the field produced by P is as close as possible to the measured field $\vec{m}$. To achieve this, we use a least-square optimization with constraints. To
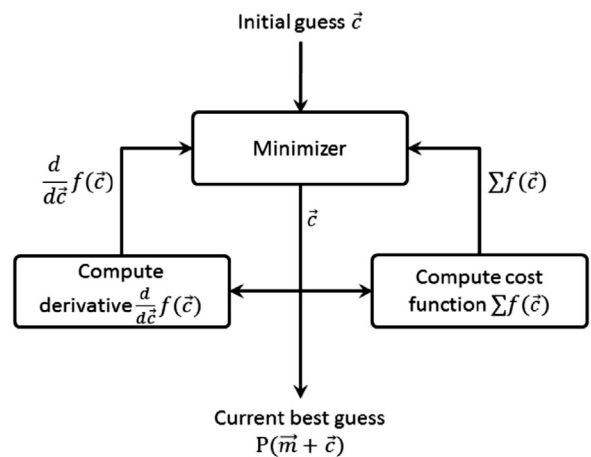

Figure 4: *A schematic overview of our framework for a given function cost function f.*

ensure similarity to the measurements, we use the following squared least error functional,

$$\underset{\vec{u}}{\text{argmin}} \left(\alpha \cdot \epsilon \cdot \left\| P(\vec{u}) - \vec{m} \right\|^2 \right), \qquad (4)$$

where $\vec{u}$ is the field that we solve for, $\alpha \geq 0$ is a constant weight and ϵ provides the local reliability for every voxel as described in Section 3. We are actually interested in $P(\vec{u})$, as discussed above. Note that this cost function corresponds to the 3DVAR cost function with a perfect model assumption. Furthermore, it is similar to the cost function used by Bostan et al. [BLV*15] without their regularization. A schematic overview of the minimization process is shown in Figure 4.

The function P could be replaced by the full Navier–Stokes equations that include the temporal evolution of the velocity field. However, as the temporal resolution of the PC-MRI data is low, we omit the temporal element of the velocity evolution that is part of the Navier–Stokes equations. Moreover, in our preliminary experiments using a function that included the temporal component and a steady flow assumption, similar to the approach by Rispoli et al. [RNNC15], led to results that deviated more from the measured data.

To constrain the minimization, we rewrite Equation (4) as a control problem, comparable to McNamara et al. [MTPS04]. That is, we substitute $\vec{u}$ by $\vec{m} + \vec{c}$, so that the goal is to find the control vector field $\vec{c}$ that corrects the measured field $\vec{m}$, it is as small as possible, and yields $P(\vec{m} + \vec{c})$ that is close to $\vec{m}$. The corresponding minimization is then

$$\underset{\vec{c}}{\text{argmin}} \left(\begin{array}{c} \alpha \cdot \epsilon \cdot \left\| P(\vec{m} + \vec{c}) - \vec{m} \right\|^2 + \\ \gamma \cdot \epsilon \cdot \left\| \vec{c} \right\|^2 \end{array} \right), \qquad (5)$$

where the second term, weighted by $\gamma \geq 0$, is a Tikhonov-style regularization term that penalizes too much control, so that changes to the measured data are kept to a minimum. Currently, our approach does not specifically avoid potential local minima. However, as the

minimization is initialized close to the measured data, when the minimization finds a local minimum the result should be relatively close to the measured, target, data.

Applying Helmholtz–Hodge decomposition assumes that the input vector field is sufficiently smooth [BNPB13], which may not necessarily be true for the measured data, due to the presence of artefacts and noise. Therefore, it is possible that not all divergence can be removed by the pressure solve. Furthermore, the minimization and the pressure solve P have conflicting goals: the minimization tries to keep the solution close to the measured data, while the pressure solve deems it physically plausible. In order to decrease the cost function, the minimization can produce non-smooth fields for which the pressure solve cannot correctly compute a smooth pressure field. To address this issue, we include an additional term that punishes the minimizer for producing fields that the pressure solve cannot project. More specifically, this term penalizes the minimizer when it produces fields with a high divergence. Therefore, this term is given by the squared divergence, that is,

$$\underset{\vec{c}}{\text{argmin}} \left(\beta \big(\nabla \cdot P(\vec{m} + \vec{c}) \big)^2 \right), \qquad (6)$$

where $\beta \geq 0$ is a constant weight. Here also the divergence computation is weighted using the variational solid boundaries approach by Batty *et al.* [BBB07] to ensure the flow does not leave the segmentation.

Our final cost functional f is a combination of Equations (5) and (6), so that the optimization problem becomes

$$\underset{\vec{c}}{\text{argmin}} \, f(\vec{c}) = \underset{\vec{c}}{\text{argmin}} \left(\begin{array}{l} \alpha \cdot \epsilon \cdot \big\| P(\vec{m} + \vec{c}) - \vec{m} \big\|^2 + \\ \beta \big(\nabla \cdot P(\vec{m} + \vec{c}) \big)^2 + \\ \gamma \cdot \epsilon \cdot \big\| \vec{c} \big\|^2 \end{array} \right), \qquad (7)$$

where α, β and γ are user-set parameters, and ϵ is defined by the data and depends on the measurement field $\vec{m}$.

The cost function is non-linear, therefore, we use a quasi-Newton type minimization. More specifically, we use the Limited-memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) algorithm [Noc80, LN89]. L-BFGS iteratively finds a minimizer by evaluating the cost function and its derivative. That is, the control vector field $\vec{c}$ provided by the minimizer is used to compute the cost functional and its gradient. The minimizer then uses this information to improve the control vector field $\vec{c}$, so that $P(\vec{m} + \vec{c})$ becomes the current approximation of the optimal physically plausible field that matches the measured field $\vec{m}$ as closely as possible. We set the algorithm to iterate until either the gradient value drops below a threshold, or the value of the cost functional did not decrease sufficiently in the last few iterations. The computation of the derivative of the cost function is done using automatic differentiation via the adjoint code of the implementation. Details on automatic differentiation and on how to convert code into its corresponding adjoint code can be found in the additional material and in the work by Giering and Kaminski [GK98] and Hogan [Hog14]. Note that it is possible to implement the adjoint code of an algorithm by hand [MTPS04]. However, this leads to two dependent codes that have to be maintained in parallel. To remove this dependency, we use the *Adept* [Hog14] library for C++ to compute the derivatives of our pressure and advection

solvers. A schematic overview of the minimization process is shown in Figure 4.

4.2. Spatial interpolation

To increase spatial resolution, we can use the system to interpolate physically plausible values between the known values. That is, when upsampling, we consider the information between the measured voxels to be missing data. Therefore, we double the spatial resolution and linearly interpolate between the known voxels to get a new measured field $\vec{m}$. As the interpolated voxels are most likely incorrect, we set their reliability ϵ to zero. That is, no matter what value they have, the minimizer will only take them into account for the divergence term given in Equation (6). Also the pressure solve does take them into account and will change them to make the field divergence-free. As the minimizer will try to reduce the error with respect to the measurements, the resulting field will be as close as possible to the measured data. Both the minimizer and pressure solve will change the interpolated voxels to best fit the measured data and have a physically plausible result.

4.3. Temporal interpolation

In order to perform temporal interpolation, we need a cost function that considers the temporal aspect of the data. Therefore, we need a model that describes how the flow evolves over time.

4.3.1. *Simulation model*

In order to simulate the behaviour of fluid over time, the Navier–Stokes equations as given by Equation (1) should be solved. Therefore, as we already have the pressure solve, we only require an advection scheme. That is, we need a model for velocity transport by the velocity field through time, that is, prescribe how the velocity field moves through time. Note that, in this work we assume blood to be inviscid. That is, we do not consider viscosity. This simplifies the model and is a common assumption regarding blood simulation [SKP18]. Moreover, our method requires simulating fluid with a negative timestep and only inviscid flow is shown to be time-reversible [DOW08].

To this end we use the semi-Lagrangian advection method introduced by Stam [Sta99]. The velocity field $\vec{u}$ is updated for every grid position by tracing backwards the path of a virtual particle. That is for a grid position p and time step Δt the new velocity is given by $\vec{u}_{\text{new}}(p) = \vec{u}(q)$, where q is the backtracked position of the virtual particle, that is, the location where the particle was at time $t - \Delta t$; hence $q = p - \Delta t \, \vec{u}(q)$. To avoid flow leaving the segmentation, we again constrain the flow on the boundaries by removing all vector components in the normal direction to the segmentation, that is, we use a Neumann-type boundary condition.

This method is stable, and furthermore, due to the iterative nature of the minimization, the simulation will not suffer much from numerical dissipation, as it will be limited to a single time step. Of course, more accurate hybrid schemes such as the fluid implicit particle method (FLIP) [BBB07] perform in general better than this simple scheme. However, a non-trivial conversion between

Table 1: *The mean and standard deviation of the absolute differences in magnitudes and angles given by |base case − result|.*

	Magnitude difference		Angle difference	
	Mean	Std dev	Mean	Std dev
Base case	1.66	4.77	14.58	20.12
SNR 10	2.43	2.68	14.77	21.63
SNR 4	2.43	3.03	17.66	22.19
Spatial interpolation	1.37	3.21	14.13	19.27
Temporal interpolation	1.64	1.89	11.61	19.44

particles and grid is needed for the derivative calculation. This makes FLIP not a suitable method and would also make it more computationally intensive.

4.3.2. *Minimization*

In order to have a physically plausible temporal interpolation, we search for a physically plausible field $\vec{y}$ at time $t + \frac{1}{2}\Delta t$, which is in between two measured velocity fields $\vec{m}_t$ and $\vec{m}_{t+\Delta t}$ at time step t and $t + \Delta t$, respectively. Note that this requires simulation backwards in time, that is, we simulate from $\vec{y}$ at time $t + \frac{1}{2}\Delta t$ to $\vec{m}_t$ at time t. As we have an inviscid fluid, the negation of the velocity field is equivalent to using a negative time step [DOW08].

Given a simulation function $S(\vec{v}_t, \Delta t)$ which evolves $\vec{v}$ as in Subsection 4.3.1 over a time step Δt, pressure solve function $P(\vec{v}_t)$ and input velocity field $\vec{v}_t$ at simulation time t, we can setup the following cost functional

$$f(\vec{c}) = \begin{pmatrix} \alpha \cdot \epsilon_1 \cdot \left\| S(P(\vec{m}_i + \vec{c}), -\frac{1}{2}\Delta t) - \vec{m}_t \right\|^2 + \\ \alpha \cdot \epsilon_2 \cdot \left\| S(P(\vec{m}_i + \vec{c}), +\frac{1}{2}\Delta t) - \vec{m}_{t+\Delta t} \right\|^2 + \\ \beta \left(\nabla \cdot P(\vec{m}_i + \vec{c}) \right)^2 + \\ \gamma \cdot \epsilon_i \cdot \left\| \vec{c} \right\|^2, \end{pmatrix}. \quad (8)$$

where $\vec{m}_i = (\vec{m}_t + \vec{m}_{t+\Delta t})/2$ is the linearly interpolated field of the two measurements and ϵ_1, ϵ_2 are the per-voxel reliability for the two measurements and ϵ is the interpolated reliability per voxel, that is, $\epsilon_i = (\epsilon_1 + \epsilon_2)/2$. This is used as the initial guess for which we want to determine the control field $\vec{c}$. Furthermore, we use α as the user-given weight to both measurements, as the temporal distance between the two measurements is equal. After the minimization of $f(\vec{c})$, the resulting physically plausible field is given by $P(\vec{m}_i + \vec{c})$. Note that his formulation is similar to Equation (7), albeit the first least-squared error term is replaced by two least-squared terms, after simulations with the nearest measurements, $\vec{m}_t$ and $\vec{m}_{t+\Delta t}$. As both measurements are not necessarily noise free, they can be denoised beforehand using the method described in Section 4.1.

5. Results and evaluation

In this section, we present results to illustrate our method and evaluate it based on our goals, that is, the resulting data should be patient-specific, physically plausible and high resolution. To this end, we

Figure 5: *Streamlines of the measured flow before our method was applied. All visualization parameters (number of lines, seeding position and cutting plane) are equal to those in Figure 6. Note the presence of a vortex in the top-right corner and the helical flow present in the top-left corner and from the centre towards the top-right corner.*

Figure 6: *Streamlines of the measured flow after our method was applied. All visualization parameters (number of lines, seeding position, cutting plane and camera parameters) are equal to those in Figure 5. Note that the flow features are preserved while more streamlines are visible due to the lack of divergence and flow not leaving the segmentation.*

will evaluate the sensitivity of our method to the most relevant parameters as well as its robustness to noise. After this, we evaluate the quality of the spatial and temporal interpolation when using our method. Finally, we demonstrate our method on *in vivo* measured data. The requirement of yielding physically plausible fields is mainly intrinsically satisfied by our method as the physical properties of the flow are part of the model. However, we also evaluate several aspects like to which extent the field is divergence free.
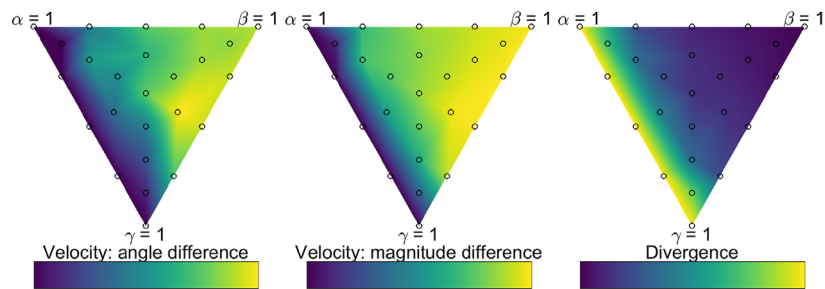
Figure 7: *A visual comparison of our method when used for denoising and interpolation. The velocity magnitude of a fixed slice through the volume is shown. For all streamline images, the same seeding positions and same number of streamlines are used. The input data are shown on the left, while the result of our method is shown on the right. The reference for the streamline images is shown by Figure 6. Note that, despite the different input scenarios, the images on the right are comparable. Moreover, the flow features as visualized by the streamlines in the reference image are also present after our approach was applied.*

We will also compare our approach to other state of the art methods. For the visualization of the flow, we use streamlines, which are commonly used for PC-MRI flow data [KYM*93, MDM*05, HH08, MFK*12]. The flow is unsteady, so the data consist of multiple time steps, and as such, pathlines would be suitable. However, the time step between the phases is relatively large in relation to the flow speed, and therefore, often short streamlines are used. Streamlines also illustrate the changes made to the flow field for a single phase, allowing us to compare individual phases. In all our experiments, the conjugate gradient method used by the pressure projection step converged in less than 20 iterations with an error tolerance of 1×10^4.

5.1. Evaluation data

There is no trivial ground-truth for blood-flow data. Ideally, one would use a ground truth for validation, for example, an analytical solution. However, while some analytical solutions to the Navier–Stokes equations exist for specific flows, they do not exist for non-trivial flows that possess the characteristics of blood flow [THZ98]. As a result, high-quality experimental data are often used for the validation of both simulations and new measurement techniques. One way to obtain such experimental data is by the use of a physical phantom. For the evaluation of our method, we use 4D PC-MRI measured data of a glass phantom that was created based on a 3D scan of an intracranial aneurysm in a patient. A pulsatile (time-varying) flow was generated based on the velocity profile measured in the internal carotid artery of the patient. More details regarding the acquisition of this type of data can be found in the work by van Ooij *et al.* [vOGP*12]. One of the advantages of phantom data over patient data is that the flow can be measured at a higher resolution and with a higher SNR, as the imaging time can be increased. Therefore, phantom data have a relatively low amount of measurement noise. Furthermore, using a glass model the vessel wall is well defined and hence less (motion) artefacts can be expected. The measured phantom data have a resolution of $80 \times 47 \times 51$ with a voxel size of $0.2 \times 0.33 \times 0.2$ mm. Flow was colour-encoded with flow speeds between 0 and 60 cm/s. The measured data are shown in Figure 5 without any processing which is common current practice. Figure 6 shows the results after applying our method. For all visual comparisons exactly the same visualization parameters are used. The streamlines use the same

seeding positions, and the same slice of the velocity magnitude is shown.

To illustrate the results and preservation of features with real data sets, several results with 4D PC-MRI data sets, as described in section 3, will be shown.

5.2. Parameter sensitivity

Our method relies on three weighting terms, α, β and γ, the ratio between these weights rather than the absolute value determines their importance for the cost function, and thus, influences the outcome. The sensitivity of our method to different ratios of the weights is shown in Figure 7. The circles represent the sampled positions of the parameter space in barycentric coordinates. Every sample represents a ratio, and the total sum of the parameters is 1, that is, $\alpha + \beta + \gamma = 1$. This means that in every corner of these equilateral triangles one of the parameters has value 1 and the others are 0. We evaluated how close the result is to the measured input by comparing the average of all velocity vectors, that is, the average difference in velocity magnitude and the average difference in angle. Furthermore, the average of the absolute divergence that is still present in the data is shown to evaluate the physical-plausibility. As the results vary a lot, the highest and lowest values were clamped. From the evaluation it follows, as expected, that the terms α and γ ensure the result is close to the measured data. Naturally, the term β leads to a more divergence-free field. If the term $\beta = 0$, the remaining divergence is relatively high, but the result is closer to the measured data. Based on this sensitivity analyses, throughout the paper, we use the following values for the weighting ratio of the terms: $\alpha = 0.4$, $\beta = 0.4$ and $\gamma = 0.2$. These values are close to the optimal shown in the figure. Although they are not the exact optimal values, this ratio seems to achieve a good trade-off between closeness to the measured data and remaining divergence. However, this setting is not critical as the influence of the parameters seems to be rather stable in our empirical analysis. Moreover, the optimal ratio deviates slightly for every measurement, and hence, would have to be determined per measurement.

Figure 5 shows the results of visualizing the measured data obtained from the scanner without any processing. Figure 6 shows
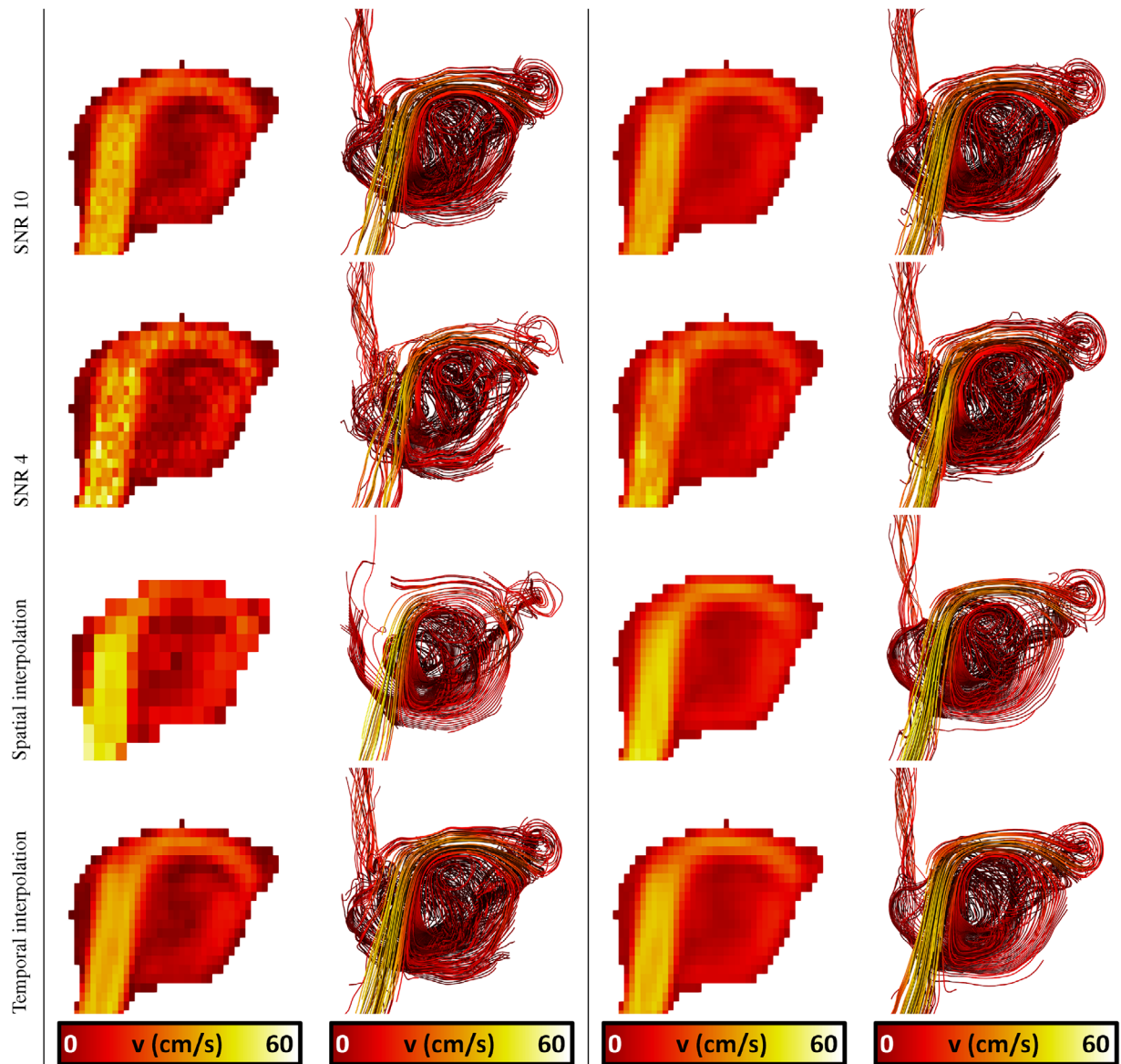
Figure 8: *The sensitivity of our method for different ratios of the three parameters α, β and γ. The two left-most triangles show how well the velocity vectors match the measured input. The third right-most triangle shows the remaining divergence. Colours represent the average value for the tested configuration. Note that darker colours mean the method performs better. The lowest and highest values have been clamped to show a better representation and the relative differences.*

the results after applying our method using the settings mentioned above. We can observe that, after our method was applied, the streamlines do not merge due to divergence and more streamlines can be traced for a longer amount of time without leaving the segmentation. The same features are visible in both figures, although they are much clearer after applying our method. Note that streamlines were seeded with same settings for both figures.

5.3. Denoising

The evaluation of the denoising capabilities of our method consists of two parts. We show the robustness of our method to various amounts of noise and by a comparison with previous methods.

5.3.1. Robustness

Given the lack of ground truth for PC-MRI, the denoised data, shown in Figure 6, will be used as a reference for our next evaluation step. To test the robustness of our method to various levels of noise, we need to control the amount of noise present in the data. Note that we do not use the original measured data as shown in Figure 5, because there is noise present in the data that would make it very difficult to determine the robustness of the method. For this, we
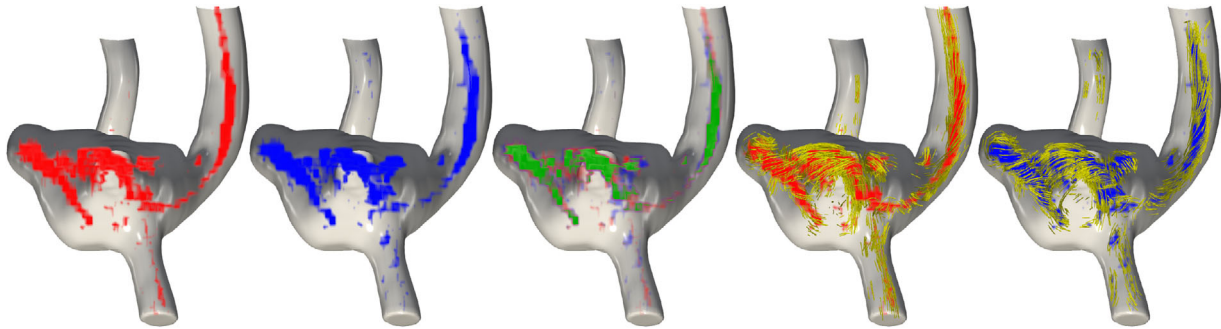
Figure 9: *Based on the lambda2 criterion, vortex cores can be found. Using volume rendering, the vortex cores are shown for the original measured data (red), the data after our approach were applied (blue) and the union of the two volumes (green). The most right two images show the corresponding streamlines seeded from the vortex cores of the measured data and the data after our approach, respectively.*

add various amounts of noise to the reference data, that is, the denoised data. In this way, we can compare the result of our approach with the expected reference data. This allows a direct comparison of the results and helps us determine how much and where the flow field is modified.

To mimic various SNR, we add Gaussian noise using the following function

$$\vec{u} = \vec{u} + G\left(0, \frac{1}{SNR} \cdot \|\vec{u}\|\right), \tag{9}$$

where $G(\mu, \sigma)$ draws a sample from the Gaussian distribution with mean μ and standard deviation σ, which approximates the noise present PC-MRI measured data [GP95]. This is done separately for every velocity component based on the signal strength $\|\vec{u}\|$ weighted by the desired SNR. Therefore, the average velocity information per voxel will be $\frac{1}{SNR} \times 100\%$ added noise. Note that, the lower the SNR, the more noise we add. Results are given for SNRs of 10 and 4, which correspond to 10% and 25% of the data being noise. Table 1 contains the comparison in velocity magnitude and angles in comparison with the ground truth. For a visual comparison, see Figure 8. For an SNR of 4, the result starts to deviate noticeably, the streamlines have a slightly different trajectory. However, the method still provides a result that is close to the reference flow. Besides a qualitative visualization of the flow using streamlines, derived flow features can be computed. One such feature is the lambda2 criterion that is commonly used to locate and visualize vortex cores and was computed for both the measured data as well as the data after our method was applied. As shown by Figure 9, the resulting vortex cores present after our method was applied match the measured data closely, indicating that features are preserved.

5.3.2. Comparison with previous work

We compare with previous methods for PC-MRI data denoising by computing the remaining divergence and difference with the measured data as shown in Figure 10. In the figure, the same slice is shown for multiple approaches to illustrate the remaining divergence and the difference between the velocity vector angles and magnitudes with the measured data. All methods are close to the measured data, except one, however, the amount of divergence that

remains present is minimal using our method. Ideally, the differences with the measured data should be small and the amount of divergence that is present in the data should be zero. As the measured data are not divergence-free, a compromise is needed, for example, to have a smaller amount of divergence, the difference with the measured data must be increased. However, for voxels where the data are less reliable, more changes can be expected. The reliability of the used data is shown in Figure 11. To this end we compare our method with existing denoising methods that have as goal to remove the divergence, namely DFW (with and without cycling spinning) [OUT*15], finite difference method [SNGP93] and RBF [BGWK13]. Furthermore, we compare the result with the approach by de Hoon *et al.* [HJEV16]. To evaluate the result, the remaining divergence and difference with the input measurement data is plotted. The difference is not weighted with the reliability for a direct comparison. The difference with the input data is assessed by the absolute difference in velocity magnitude and angle between the velocity vectors. This comparison is shown in Figure 10, and more slices (which give comparable results) can be visualized using the interactive Matlab plots in the additional materials. Besides the method by de Hoon *et al.*, which suffers from damping, all methods have small differences in velocity magnitude and angle. However, none of the methods we compare with is able to remove all divergence which corresponds with the findings of Sereno *et al.* [SKP18]. Note that most divergence is present where the reliability is relatively low, that is, near the vessel wall. Additionally, as the existing denoising approaches did not remove much of the divergence, we apply two possible approaches, that is, a denoising method D that focuses on the reduction of noise in the data, and our pressure solver P, which targets the reduction of divergence. This should circumvent the bias of the approaches towards a specific aspect of the measured flow. For this we chose to use DFW with spin, which, of the tested denoising approaches, yields results that are the closets to the measured data. The result is shown in Figure 10, under the label $P(D)$. Note that the result is comparable to applying either of the approaches individually. The amount of divergence that remains present in the data is lowest for our method, suggesting that our method is able to remove more divergence while keeping the results close to the input measurements. Further note that when $\beta = 0$ for our method, that is, the divergence term is neglected, the result is closer to the measured data as is to be expected, however, more divergence remains present.
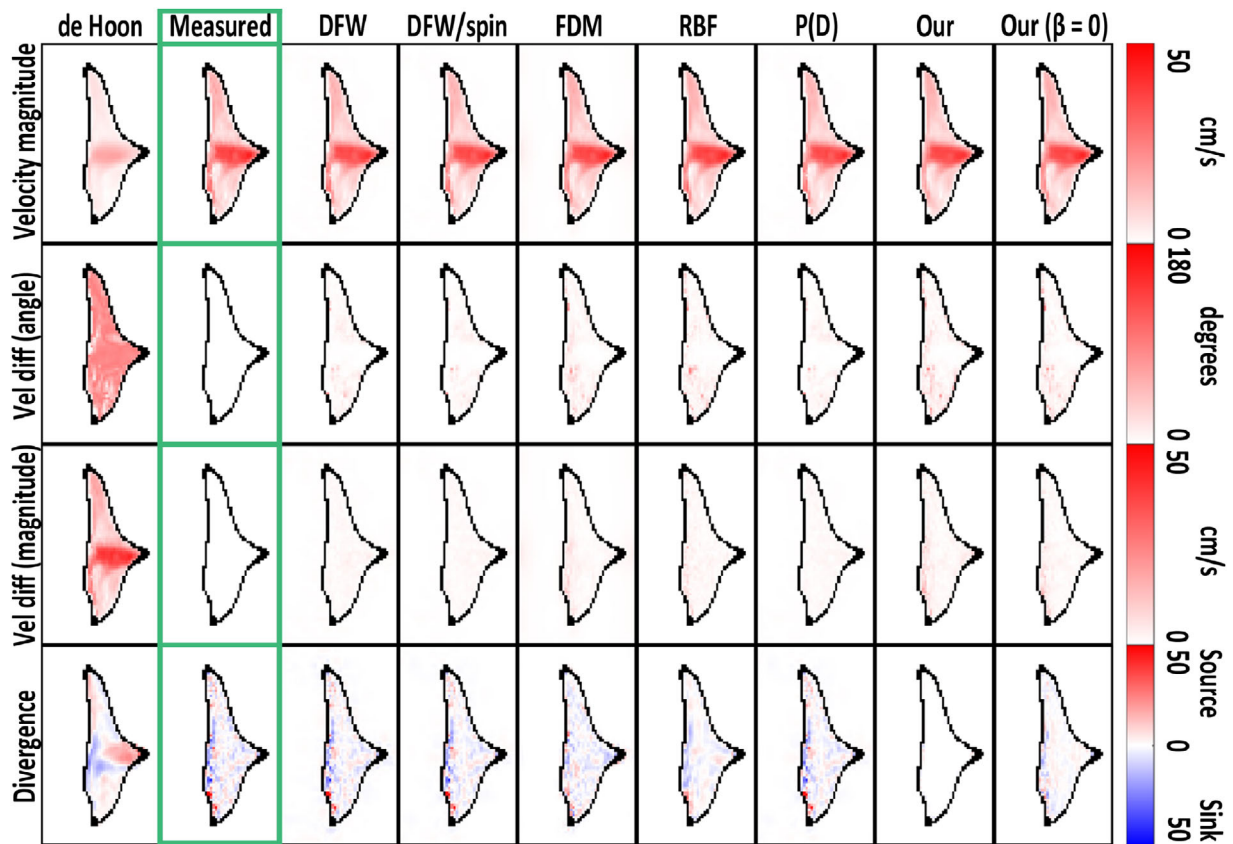
Figure 10: *A visual comparison of our method with previous methods; the measured data that are used as input are highlighted by the green border. The same slice is shown in all images. The top row shows the velocity magnitude of the resulting field, while the second and third row show the difference with the measured input data. The bottom row shows the divergence that is present in the resulting velocity field. Note that all methods, except for de Hoon et al., are close to the measured data, as shown by the difference between the velocity vector angles and magnitudes. The divergence that remains present in the data is minimal using our method. Note that the method by de Hoon et al. is shown separately to allow for a more direct visual comparison of the remaining methods.*
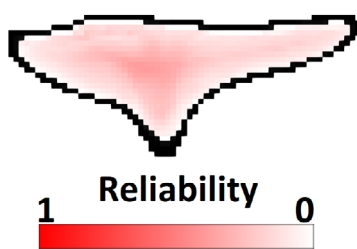

Figure 11: *The same slice as in Figure 10 through the aneurysm shows the reliability per voxel of the measured data. Note that the reliability is lowest near the vessel wall.*

5.4. Interpolation

A common approach to test any interpolation method is to remove parts of the data and see how well the interpolation can reconstruct the removed information. To evaluate the *spatial interpolation* capability of our method, all voxels for which one of its coordinates is

even are removed from the data. Therefore, the downsampled data is 1/8 of the original size, as shown by the penultimate row of Figure 8.

We apply our spatial interpolation scheme from Section 4.2 on this lower resolution data and compare the result with the input data in Table 1. Visually, the resulting flow field deviates slightly from the reference mainly due the loss of small-scale details, as can be seen in Figure 8.

For the evaluation of the *temporal interpolation* described in Section 4.3 the same concept is applied. Therefore, one of the measured phases is removed, and the duration of the phases is doubled. The final row in Figure 8 shows the input field, which is linearly interpolated in between two consecutive phases. Linear interpolation is often used for the visualization of PC-MRI data, however, it does not guarantee a field that is physically plausible. Visually, the result of our method matches the reference flow data well and better than linear interpolation. This is supported by the small differences in velocity magnitude and angle differences, as given in Table 1. This is also shown by Figure 12 that gives a direct comparison with linear interpolation, the reference measured data and our method. Moreover, we compare our method and linear interpolation with a single
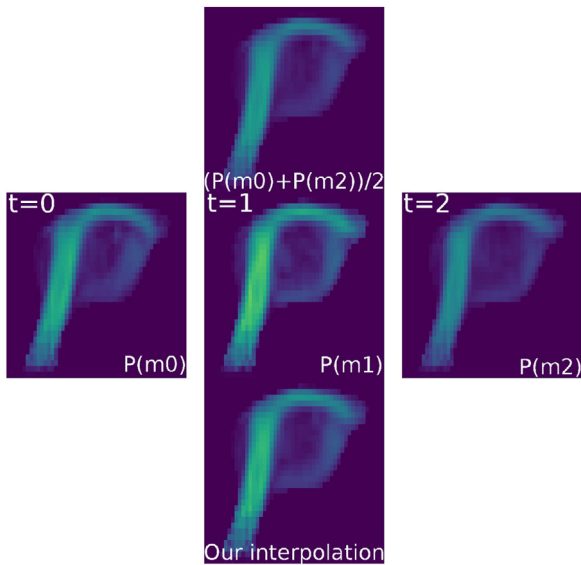
Figure 12: *Comparison of temporal interpolation based on two pressure solved velocity fields (left and right) using linear interpolation (top), the expected pressure solved measured field (centre) and our approach (bottom).*

Figure 13: *The error in comparison to a single time step of the Navier–Stokes simulation and linear interpolation (left) and our approach (right). Our approach has a small error. Linear interpolation on the other hand has a large error in the range of the velocity magnitude of the underlying velocity field.*

time step of the Navier–Stokes simulation without minimization. That is, we evaluate our approach to determine whether it fulfils the Navier–Stokes equation by comparing our result with the right-hand side of Equation (1). The right-hand side of Equation (1) provides us with an updated velocity field after a single time step of the Navier–Stokes simulation. The changes in the velocity field after this time step provides us with the left-hand side of Equation (1), that is, the temporal derivative $\frac{\partial \vec{u}}{\partial t}$. Note that this temporal derivative can also be calculated from any two consecutive velocity fields by taking the difference between the two velocity fields: $\frac{\partial \vec{x}}{\partial t} = \vec{x}_{t+1} - \vec{x}_t$. Hence, we can compare the temporal derivative of different approaches to the solution provided by the Navier–Stokes simulation and compute the error: error $= \frac{\partial \vec{u}}{\partial t} - \frac{\partial \vec{x}}{\partial t}$. Figure 13 shows the magnitude of this error for both linear interpolation and for our approach. It shows that our approach has only a small error. Note that, linear interpolation and higher order interpolation methods do not reproduce the influence of advection that is present in flow data, which could explain the large error.

5.5. Vorticity near the aortic valve

Next, we illustrate the advantage of our method for not requiring an accurate model of the anatomy, which, in some cases, can be very difficult or impossible to model, for example, the anatomy near the heart valves. However, this underlying anatomy can have a big impact on the resulting flow. In the aorta valve, three sinuses exist that correspond to the three cusps of the aortic valve. These sinuses are known to be important for an efficient blood flow. The vortices that form in these sinuses help to close the valve leaflets [BB68, BDC14, VFdH*18] and prevent regurgitation of the blood. This interaction between flow and anatomy is difficult to model [MNvTK*15] due to
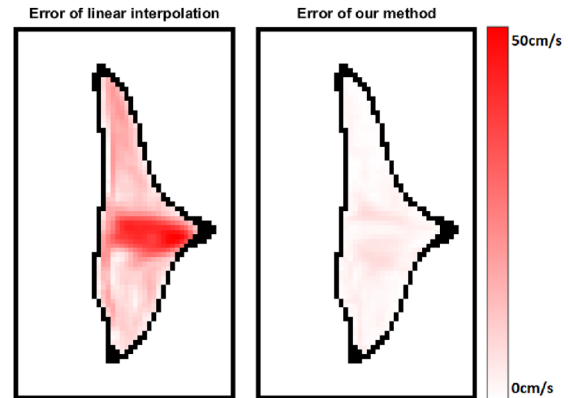
the small scale and lack of a possibility to measure this phenomena *in vivo* [TBE*16].

The vortices, however, can be measured using PC-MRI, as can be seen in Figure 14. This data have a resolution of $127 \times 127 \times 23$ with a voxel size of $2.5 \times 2.5 \times 2.5$ mm. As our method targets the whole velocity field, and not just in/out-flow conditions, it can correct the (measured) flow in the sinuses without removing the important vortices present in the PC-MRI data. Even without a model of the aortic valve and its leaflets, our method maintains the vortices. Figure 14 shows the result of applying our denoising and spatial interpolation on the flow near the aortic valve. Vortices that help closing the valve leaflets are maintained. Flow in the normal direction of the segmentation is present in the original measurement as indicated by the blue circle. The flow in the normal direction of the segmentation is corrected after the application of our method keeping the features of the flow clearly.

5.6. Circle of Willis

The circle of Willis is a relatively small circular vessel structure that supplies blood to the brain and the surrounding structures. The arteries of the circle of Willis have diameters smaller than 1 cm. Although the flow is measured using PC-MRI [vOZV*13], the SNR is negatively affected, as a very fine spatial resolution is needed, as a result analysis of this data is difficult. The data used in this experiment have a resolution of $383 \times 383 \times 39$, with a voxel size of $0.47 \times 0.47 \times 0.5$ mm. Figure 15 shows the measured flow in a section of the circle of Willis. The data are denoised using our method, as can be seen in Figure 16. For instance, there is flow perpendicular to the main flow direction near a vortex, which is highly unlikely. Our method does correct for this, but more importantly, it preserves the subtle flow features, such as vortices that are present in the data.

6. Performance

In this section we present an indication of the computational and memory costs of our approach. The experiments were performed
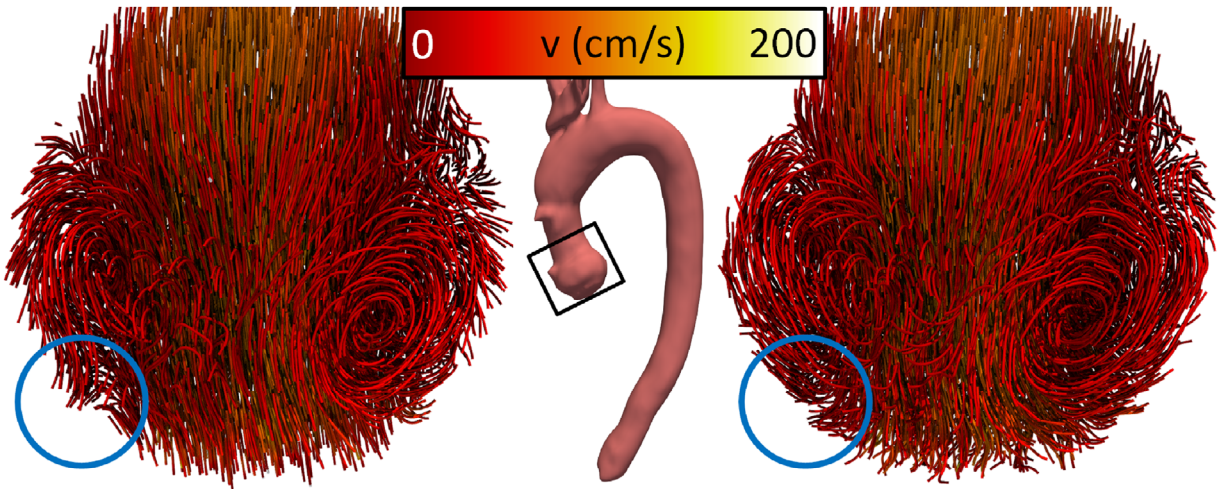
Figure 14: *Flow in the aorta root showing the vorticity in the sinuses. These vortices that help closing the valve leaflets are maintained. On the left, measured flow in the aorta: vortices can be seen in the aorta root. Short streamlines seeded from a spherical seeding region are used for both flow visualizations.*
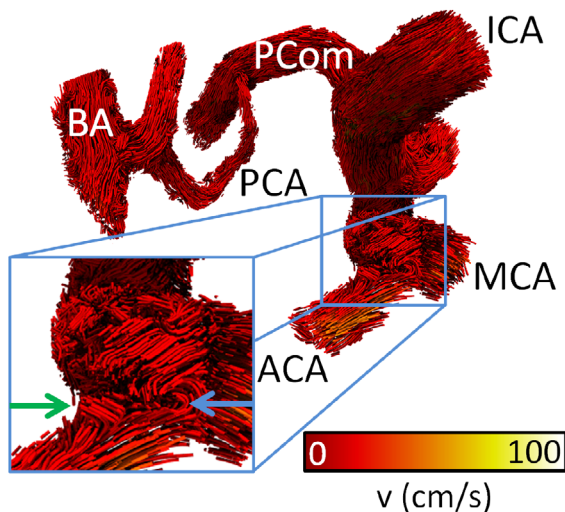

Figure 15: *Streamline visualization of the measured flow in a section of the circle of Willis. Note the unlikely flow perpendicular to the main flow direction and the nearby vortex indicated by the green and blue arrows, respectively. The arteries shown are the basilar artery (BA), posterior cerebral artery (PCA), posterior communicating artery (PCom), internal carotid artery (ICA), middle cerebral artery (MCA) and anterior cerebral artery (ACA).*
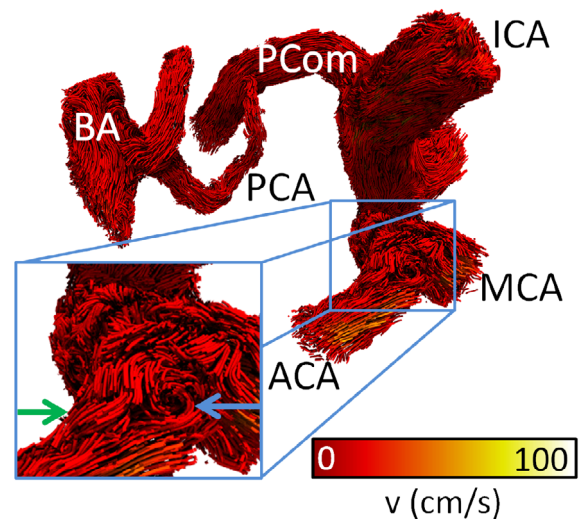
Figure 16: *Streamline visualization of the corrected flow in a section of the circle of Willis. Our method corrected the perpendicular flow, as shown by the green arrow, while the nearby vortex is maintained (blue arrow).*

on a Desktop computer with an Intel i7-3820 CPU and 32GB of memory. For most of the evaluations using the phantom data and the upsampled aorta root flow, our method needed between 200 and 300 iterations to converge and takes around 5-10 min. At most 8GB of memory was used. Temporal interpolation required much more memory (around 20GB) as two simulations are performed every iteration and takes around half an hour to converge. The circle of Willis data required more iterations to converge (around 600), which

in the worst case took 45 min to converge. Note that these data have a relatively high resolution and low SNR.

All evaluations could be executed in parallel on multiple machines or a computer cluster. This would have reduced the time further. Furthermore, parallelization of the code might be possible, but further complicates the computation of the gradient of the cost functionals. It is difficult to compare the performance to other methods but concerning the denoising approaches we compared with our method requires more time and resources. Concerning data assimilation methods, the times reported are usually much higher than we found using our method.

7. Conclusion and Future work

4D PC-MRI provides measurements of time-varying volumetric blood flows. Despite providing the most complete data amongst all imaging methods, it is still prone to artefacts, noise and has limited spatial and temporal resolutions. Hence, the derived flow does not follow the fluid physics laws.

In this work, we presented a novel data-assimilation approach for PC-MRI that combines a physics-based model and measured PC-MRI data to obtain physically plausible, patient-specific data. Different from other data assimilation methods, our methodology focuses on improving the measured data to become physically plausible while minimizing the difference to the original measurements by applying automatic differentiation. Using three parameters the user can tune minimization to balance both physically plausibility and deviation from the given measured data. We also aim at a limited sensitivity with respect to the accuracy of the given anatomical model.

We have shown, using phantom data, that the resulting method is capable of denoising PC-MRI measurements and yields physically plausible data, with minimal deviation from to the measured flow field. Furthermore, we have shown that our method enables physics-based interpolation in both the spatial and temporal domains preserving the existing features. In addition to using phantom data, we have shown, as a proof of concept, that our method preserves important features in PC-MRI flow data. However, more extensive validation and comparison will be needed for the adaptation of the method to specific applications. Such validation is tedious and requires effort beyond the scope of the paper.

Currently, our implementation did not focus on performance, especially the memory usage could be improved, for example, by excluding empty voxels from the minimization or an optimized implementation of the gradient computation.

Furthermore, our approach does not specifically avoid potential local minima. However, as the minimization is initialized close to the measured data, a local minimum would be relatively close to the measured, target, data. To avoid local minima, other optimization methods could be tested, such as simulated annealing, introducing randomness in the initialization of the minimization process and therefore using multiple starting points for the minimization. In this case, finding the optimal solution would be more likely. However, preliminary experiments seem to yield velocity fields that are close to the current solution, moreover, the computation time naturally increases. Nevertheless, it would be insightful to further investigate local-minima avoidance.

Another improvement would be the support of soft boundaries, that is, allowing some flow to exist outside the segmentation. In this way, we can reduce the sensitivity of our method to the given anatomy even more. This could potentially be achieved by penalizing flows outside the segmentation. Another option would be to use control forces, for instance, using the geometric potential field from Hong *et al.* [HK04]. Furthermore, improving the model would most likely improve the accuracy of the presented method further by considering the hemorheology, that is, the properties of blood, more accurately. For example, by considering the non-Newtonian viscosity and deformability of blood cells.

In the future, our approach shows potential to be used to reconstruct compressed PC-MRI data or enhance measured data. For the measured data, the spatial and temporal resolutions determine the SNR. It would be interesting to determine scanning parameters for which we can achieve the highest noiseless data after applying our spatial and temporal denoising. The evaluation of such methods is especially challenging given the lack of ground truth in this domain.

To the extent of our knowledge, we presented the first data assimilation method for PC-MRI that updates the full measured data, and enables denoising while supporting spatial and temporal interpolation.

References

[BB68] BELLHOUSE B., BELLHOUSE F.: Mechanism of closure of the aortic valve. *Nature 217*, 5123 (1968), 86–87.

[BBB07] BATTY C., BERTAILS F., BRIDSON R.: A fast variational framework for accurate solid-fluid coupling. *ACM Transactions on Graphics 26*, 3 (2007), 100.

[BDC14] BISSELL M. M., DALL'ARMELLINA E., CHOUDHURY R. P.: Flow vortices in the aortic root: In vivo 4D-MRI confirms predictions of Leonardo da Vinci. *European Heart Journal 35*, 20 (2014), 1344.

[BGWK13] BUSCH J., GIESE D., WISSMANN L., KOZERKE S.: Reconstruction of divergence-free velocity fields from cine 3D phase-contrast flow measurements. *Magnetic Resonance in Medicine 69*, 1 (2013), 200–210.

[BLV*15] BOSTAN E., LEFKIMMIATIS S., VARDOULIS O., STERGIOPULOS N., UNSER M.: Improved variational denoising of flow fields with application to phase-contrast MRI data. *IEEE Signal Processing Letters 22*, 6 (June 2015), 762–766.

[BNPB13] BHATIA H., NORGARD G., PASCUCCI V., BREMER P.: The Helmholtz-Hodge decomposition-a survey. *IEEE Transactions on Visualization and Computer Graphics 19*, 8 (August 2013), 1386–1404.

[Buo94] BUONOCORE M. H.: Algorithms for improving calculated streamlines in 3-d phase contrast angiography. *Magnetic Resonance in Medicine 31*, 1 (1994), 22–30.

[CT17] CHU M., THUEREY N.: Data-driven synthesis of smoke flows with cnn-based feature descriptors. *ACM Transactions on Graphics 36*, 4 (July 2017), 69:1–69:14.

[DOW08] DUPONCHEEL M., ORLANDI P., WINCKELMANS G.: Time-reversibility of the euler equations as a benchmark for energy conserving schemes. *Journal of Computational Physics 227*, 19 (2008), 8736–8752.

[DPV12] D'ELIA M., PEREGO M., VENEZIANI A.: A variational data assimilation procedure for the incompressible Navier-Stokes equations in hemodynamics. *Journal of Scientific Computing 52*, 2 (August 2012), 340–359.

[DUS*11] DEE D. P., UPPALA S. M., SIMMONS A. J., BERRISFORD P., POLI P., KOBAYASHI S., ANDRAE U., BALMASEDA M. A., BALSAMO G., BAUER P., BECHTOLD P., BELJAARS A. C. M., VAN DE BERG L., BIDLOT J., BORMANN N., DELSOL C., DRAGANI R., FUENTES M., GEER A. J., HAIMBERGER L., HEALY S. B., HERSBACH H., HSLM E. V., ISAKSEN L., KĔLLBERG P., KÖHLER M., MATRICARDI M., MCNALLY A. P., MONGE-SANZ B. M., MORCRETTE J.-J., PARK B.-K., PEUBEY C., DE ROSNAY P., TAVOLATO C., THÉPAUT J.-N., VITART F.: The era-interim reanalysis: Configuration and performance of the data assimilation system. *Quarterly Journal of the Royal Meteorological Society 137*, 656 (2011), 553–597.

[FA03] FATOURAEE N., AMINI A.: Regularization of flow streamlines in multislice phase-contrast MR imaging. *IEEE Transactions on Medical Imaging 22*, 6 (June 2003), 699–709.

[FBB*18] FATHI M. F., BAKHSHINEJAD A., BAGHAIE A., SALONER D., SACHO R. H., RAYZ V. L., D'SOUZA R. M.: Denoising and spatial resolution enhancement of 4D flow MRI using proper orthogonal decomposition and lasso regularization. *Computerized Medical Imaging and Graphics 70* (2018), 165–172.

[FHH*10] FRIMAN O., HENNEMUTH A., HARLOFF A., BOCK J., MARKL M., PEITGEN H.: Probabilistic 4D blood flow mapping. *MICCAI 13*, 3 (2010), 416–423.

[FHH*11] FRIMAN O., HENNEMUTH A., HARLOFF A., BOCK J., MARKL M., PEITGEN H.: Probabilistic 4D blood flow tracking and uncertainty estimation. *Medical Image Analysis 15*, 5 (2011), 720–728. (Special Issue on the 2010 Conference on Medical Image Computing and Computer-Assisted Intervention.)

[FNE*18] FUNKE S. W., NORDAAS M., EVJU Ø., ALNÆS M. S., MARDAL K.-A.: Variational data assimilation for transient blood flow simulations. *International Journal for Numerical Methods in Biomedical Engineering 35*, 1 (2019), e3152.

[Gas14] GASTEIGER R.: *Visual Exploration of Cardiovascular Hemodynamics*. PhD thesis, Otto-von-Guericke University, Magdeburg, 2014.

[GCM*18] GUERRA T., CATARINO C., MESTRE T., SANTOS S., TIAGO J., SEQUEIRA A.: A data assimilation approach for non-Newtonian blood flow simulations in 3D geometries. *Applied Mathematics and Computation 321* (2018), 176–194.

[GITH14] GREGSON J., IHRKE I., THUEREY N., HEIDRICH W.: From capture to simulation: Connecting forward and inverse problems in fluids. *ACM Transactions on Graphics 33*, 4 (July 2014), 139:1–139:11.

[GK98] GIERING R., KAMINSKI T.: Recipes for adjoint code construction. *ACM Trans. Math. Softw. 24*, 4 (December 1998), 437–474.

[GP95] GUDBJARTSSON H., PATZ S.: The Rician distribution of noisy MRI data. *Magnetic Resonance in Medicine 34* (1995), 910–914.

[HBB*10] HIRATZKA L., BAKRIS G., BECKMAN J., BERSIN R., CARR V., CASEY JR D., EAGLE K., HERMANN L., ISSELBACHER E., KAZEROONI E., KOUCHOUKOS N., LYTLE B., MILEWICZ D., REICH D., SEN S., SHINN J., SVENSSON L., WILLIAMS D.: Guidelines for the diagnosis and management of patients with thoracic aortic disease. *Journal of the American College of Cardiology 55*, 14 (February 2010), e27–e129.

[HH08] HOPE T. A., HERFKENS R. J.: Imaging of the thoracic aorta with time-resolved three-dimensional phase-contrast MRI: A review. *Seminars in Thoracic and Cardiovascular Surgery 20*, 4 (2008), 358–364.

[HJEV16] HOON N. H. L. C. d., JALBA A. C., EISEMANN E., VILANOVA A.: Temporal interpolation of 4D PC-MRI blood-flow measurements using bidirectional physics-based fluid simulation. In *Eurographics Workshop on Visual Computing for Biology and Medicine*. S. Bruckner, B. Preim, A. Vilanova, H. Hauser, A. Hennemuth, and A. Lundervold (Eds.). The Eurographics Association, Aire-la-Ville, Switzerland (2016), pp. 59–68.

[HK04] HONG J.-m., KIM C.-h.: Controlling fluid animation with geometric potential. *Computer Animation and Virtual Worlds 15*, 3-4 (2004), 147–157.

[Hog14] HOGAN R. J.: Fast reverse-mode automatic differentiation using expression templates in c++. *ACM Transactions on Mathematical Software 40*, 4 (July 2014), 26:1–26:16.

[IAWW18] II S., ADIB M. A. H. M., WATANABE Y., WADA S.: Physically consistent data assimilation method based on feedback control for patient-specific blood flow analysis. *International Journal for Numerical Methods in Biomedical Engineering 34*, 1 (2018), e2910. e2910 cnm.2910.

[KBvP*16] KÖHLER B., BORN S., VAN PELT R. F. P., HENNEMUTH A., PREIM U., PREIM B.: A survey of cardiac 4D PC-MRI data processing. *Computer Graphics Forum 36*, 6 (2016), 5–35.

[KYM*93] KILNER P., YANG G., MOHIADDIN R., FIRMIN D., LONGMORE D.: Helical and retrograde secondary flow patterns in the aortic arch studied by three-directional magnetic resonance velocity mapping. *Circulation 88* (1993), 2235–2247.

[LBC*15] LORENC A. C., BOWLER N. E., CLAYTON A. M., PRING S. R., FAIRBAIRN D.: Comparison of hybrid-4denvar and hybrid-4dvar data assimilation methods for global nwp. *Monthly Weather Review 143*, 1 (2015), 212–229.

[LEVDGS19] LE T. B., ELBAZ M. S. M., VAN DER GEEST R. J., SOTIROPOULOS F.: High resolution simulation of diastolic left ventricular hemodynamics guided by four-dimensional flow magnetic resonance imaging data. *Flow, Turbulence and Combustion 102*, (January 2019), 3–26.

[LKTW12] LOECHER M., KECSKEMETI S., TURSKI P., WIEBEN O.: Comparison of divergence-free algorithms for 3D MRI with three-directional velocity encoding. *Journal of Cardiovascular Magnetic Resonance 14*, 1 (February 2012), W64.

[LN89] LIU D. C., NOCEDAL J.: On the limited memory bfgs method for large scale optimization. *Mathematical Programming 45*, 1 (August 1989), 503–528.

[MBG*15] MOZAFFARIAN D., BENJAMIN E., GO A., ARNETT D., GLAHA M., CUSHMAN M., DE FERRANTI S., DESPRÉS J., FULLERTON H., HOWARD V., MUFFMAN M., JUDD S., KISSELA B., LACKLAND D., LICHTMAN J., LISABETH L., LIU S., MACKEY R., MATCHAR D., MCGUIRE D., 3RD E. M., MOY C., MUNTNER P., MUSSOLINO M., NASIR K., NEUMAR R., NICHOL G., PALANIAPPAN L., PANDEY D., REEVES M., RODRIQUEZ C., SORLIE P., STEIN J., TOWFIGHI A., TURAN T., VIRANI S., WILLEY J., WOO D., YEH R., TURNER M.: Heart disease and stroke statistics–2015 update: a report from the American Heart Association. *Circulation 131*, 4 (Jan 2015), e29–e322.

[MDM*05] MARKL M., DRANEY M. T., MILLER D. C., LEVIN J. M., WILLIAMSON E. E., PELC N. J., LIANG D. H., HERFKENS R. J.: Time-resolved three-dimensional magnetic resonance velocity mapping of aortic flow in healthy volunteers and patients after valve-sparing aortic root replacement. *The Journal of Thoracic and Cardiovascular Surgery 130*, 2 (2005), 456–463. http://www.sciencedirect.com/science/article/pii/S0022522305001248, https://doi.org/10.1016/j.jtcvs.2004.08.056.

[MFK*12] MARKL M., FRYDRYCHOWICZ A., KOZERKE S., HOPE M., WIEBEN O.: 4D flow MRI. *Journal of Magnetic Resonance Imaging 36*, 5 (2012), 1015–1036. https://doi.org/10.1002/jmri.23632.

[MNvTK*15] MORRIS P. D., NARRACOTT A., VON TENGG-KOBLIGK H., SILVA SOTO D. A., HSIAO S., LUNGU A., EVANS P., BRESSLOFF N. W., LAWFORD P. V., HOSE D. R., GUNN J. P.: Computational fluid dynamics modelling in cardiovascular medicine. *Heart* (2015). https://heart.bmj.com/content/early/2015/10/28/heartjnl-2015-308044, https://heart.bmj.com/content/early/2015/10/28/heartjnl-2015-308044.full.pdf, https://heart.bmj.com/content/early/2015/10/28/heartjnl-2015-308044.full.pdf.

[MTPS04] MCNAMARA A., TREUILLE A., POPOVIĆ Z., STAM J.: Fluid control using the adjoint method. *ACM Transactions on Graphics 23*, 3 (Aug. 2004), 449–456.

[NC10] NIELSEN M. B., CHRISTENSEN B. B.: Improved variational guiding of smoke animations. *Computer Graphics Forum 29*, 2 (2010), 705–712.

[Noc80] NOCEDAL J.: Updating quasi-Newton matrices with limited storage. *Mathematics of Computation 35*, 151 (1980), 773–782.

[OUT*15] ONG F., UECKER M., TARIQ U., HSIAO A., ALLEY M. T., VASANAWALA S. S., LUSTIG M.: Robust 4d flow denoising using divergence-free wavelet transform. *Magnetic Resonance in Medicine 73*, 2 (2015), 828–842.

[PBC*16] PENNY S. G., BEHRINGER D., CARTON J., KALNAY E., XUE Y.: Towards an operational hybrid global ocean data assimilation system (hybrid-godas) at ncep. In *American Geophysical Union, Ocean Sciences Meeting 2016* (February 2016).

[PM17] PAN Z., MANOCHA D.: Efficient solver for spacetime control of smoke. *ACM Transactions on Graphics 36*, 5 (July 2017). 1–13. https://doi.org/10.1145/3016963.

[RNNC15] RISPOLI V. C., NIELSEN J. F., NAYAK K. S., CARVALHO J. L. A.: Computational fluid dynamics simulations of blood flow regularized by 3D phase contrast MRI. *BioMedical Engineering OnLine 14*, 1 (November 2015), 110.

[SKP18] SERENO M. F., KÖHLER B., PREIM B.: Comparison of divergence-free filters for cardiac 4D PC-MRI data. In *Bildverarbeitung für die Medizin 2018*. A. Maier, T. M. Deserno, H. Handels, K. H. Maier-Hein, C. Palm, T. Tolxdorf (Eds.). Springer, Berlin–Heidelberg, Germany (2018), pp. 139–144.

[SNGP93] SONG S. M., NAPEL S., GLOVER G. H., PELC N. J.: Noise reduction in three-dimensional phase-contrast MR velocity measurementsl. *Journal of Magnetic Resonance Imaging 3*, 4 (1993), 587–596.

[Sta99] STAM J.: Stable fluids. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques* (1999), ACM Press/Addison-Wesley Publishing Co., pp. 121–128.

[TBE*16] TOMA M., BLOODWORTH C. H., EINSTEIN D. R., PIERCE E. L., COCHRAN R. P., YOGANATHAN A. P., KUNZELMAN K. S.: High-resolution subject-specific mitral valve imaging and modeling: experimental and computational methods. *Biomechanics and Modeling in Mechanobiology 15*, 6 (December 2016), 1619–1630.

[THZ98] TAYLOR C. A., HUGHES T., ZARINS C.: Finite element modeling of blood flow in arteries. *Computer Methods in Applied Mechanical Engineering 7825*, 97 (1998), 155–196.

[TKPR06] THÜREY N., KEISER R., PAULY M., RÜDE U.: Detail-preserving fluid control. In *Proceedings of the 2006 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Aire-la-Ville, Switzerland, 2006), Eurographics Association, pp. 7–12.

[VFdH*18] VENDRIK J., FARAG E. S., DE HOON N. H. L. C., KLUIN J., BAAN J.: Presence of aortic root vortex formation after TAVI with CENTERA confirmed using 4D-flow magnetic resonance imaging. *The International Journal of Cardiovascular Imaging 34*, (August 2018), 1947–1948.

[vOGP*12] VAN OOIJ P. J., GUÉDON A., POELMA C., SCHNEIDERS J. J., RUTTEN M. C. M., MARQUERING H. A., MAJOIE C. B. L. M., VANBAVEL E., NEDERVEEN A. J.: Complex flow patterns in a real-size intracranial aneurysm phantom: phase contrast MRI compared with particle image velocimetry and computational fluid dynamics. *NMR in Biomedicine 25* (2012), 14–26.

[vOZV*13] VAN OOIJ P., ZWANENBURG J. J. M., VISSER F., MAJOIE C. B., VANBAVEL E., HENDRIKSE J., NEDERVEEN A. J.: Quantification and visualization of flow in the Circle of Willis: Time-resolved three-dimensional phase contrast MRI at 7T compared with 3T. *Magnetic Resonance in Medicine 69*, 3 (2013), 868–876.

[YBK*96] YANG G. Z., BURGER P., KILNER P. J., KARWATOWSKI S. P., FIRMIN D. N.: Dynamic range extension of cine velocity measurements using motion registered spatiotemporal phase unwrapping. *Journal of Magnetic Resonance Imaging 6*, 3 (1996), 495–502.

Supporting Information

Additional supporting information may be found online in the Supporting Information section at the end of the article.

Data S1

Data S2

Data S3

Data Video S4