Proceedings of the ASME 2021
Fluids Engineering Division Summer Meeting
FEDSM2021
August 10-12, 2021, Virtual, Online

DRAFT-FEDSM2021-66080

INVERSE PROBLEMS IN MAGNETIC RESONANCE VELOCIMETRY: SHAPE, FORCING AND BOUNDARY CONDITION INFERENCE

Alexandros Kontogiannis and Matthew Juniper
Department of Engineering
University of Cambridge
Cambridge, United Kingdom, CB21PZ
ak2239@cam.ac.uk, mpj1001@cam.ac.uk

ABSTRACT

We derive and implement an algorithm that takes noisy magnetic resonance velocimetry (MRV) images of Stokes flow and infers the velocity field, the most likely position of the boundary, the inlet and outlet boundary conditions, and any body forces. We do this by minimizing a discrepancy norm of the velocity fields between the MRV experiment and the Stokes problem, and at the same time we obtain a filtered (denoised) version of the original MRV image. We describe two possible approaches to regularize the inverse problem, using either a variational technique, or Gaussian random fields. We test the algorithm for flows governed by a Poisson or a Stokes problem, using both real and synthetic MRV measurements. We find that the algorithm is capable of reconstructing the shape of the domain from artificial images with a low signal-to-noise ratio.

1 INTRODUCTION

Magnetic resonance velocimetry is a noninvasive experimental technique that can measure all three components of a three-dimensional (3D) velocity field, although these images are often noisy. Combining *a priori* knowledge (in the form of a physical model) with noisy experimental data, yields the possibility of both inferring unknown quantities (e.g. pressure) and reconstructing (de-noising) the MRV signal. Inverse problems such as this for the Navier–Stokes equations have been intensively studied during the last decade, mainly enabled by the increase of available computing power. Recent applications in fluid mechanics range from the forcing inference problem [1], to the reconstruction of particle image velocimetry (PIV) signals [2] and the identification of optimal sensor arrangements [3,4]. Regularization methods for ill-posed problems are reviewed in [5] and [6], from a Bayesian and a variational perspective, respectively, and the well-posedness of such Bayesian inverse problems is addressed in [7].

In a recent study [8], the inverse problem of finding the Dirichlet (inlet) boundary condition, to match a Navier–Stokes problem solution with an MRV image, is treated for steady 3D flows in a glass replica of the human aorta. Their MRV measurements are used only in a restricted domain that excludes the near-wall regions. The model-data discrepancy is measured with the L^2-norm and additional regularization terms are used for the boundary condition. The goal is to improve near-wall velocity predictions and increase the accuracy of the computed wall-shear stress of blood flows, in order to assist in the clinical assessment of endothelial disorders. The same formulation is extended to periodic flows in [9, 10]. The authors use the harmonic balance method for the temporal discretization of the Navier–Stokes problem, and report that their method is 15 times faster than its traditional unsteady counterpart. In [11] the problem of inferring both the Dirichlet (inlet) boundary condition and the initial condition is studied for unsteady blood flows and 4D MRV data, with applications to cerebral aneurysms for 2D and 3D unsteady flows. We note that the above studies [9, 10, 11] consider rigid (i.e. non-flexible) boundaries and require *a priori* an accurate (and time-averaged) geometric representation of the blood vessel.

To find the geometry of the blood vessel, computed tomography or magnetic resonance angiography is often used and the acquired signal is subsequently reconstructed, segmented, and smoothed. This process not only requires substantial effort and the design of an additional experiment, but it also introduces geometric uncertainties [12, 13], which, in turn, affect the predictive confidence of arterial shear-stress distributions and their mappings [14]. More importantly, the assumption of rigid boundaries implies that a time-averaged geometry has to be used, as most problems in haemodynamics involve flexible boundaries and periodic flows. A more consistent approach to this problem would be to include the blood vessel geometry in the reconstruction process. In this way, the physical model can better adapt to the MRV experiment and correct the geometric errors. In this case, only a starting guess is needed for the blood vessel geometry. This starting guess can be obtained by automatically segmenting the MRV data to obtain an approximation of the blood vessel boundary.

This paper considers only 2D steady Stokes flow. A future paper will extend the methods discussed hereafter to the 2D steady Navier–Stokes problem. This paper is structured as follows. In section 2.1 we formulate a shape inference (reconstruction) algorithm. Section 2.2 describes the classical approach to obtain the generalized gradients for the forcing term and the boundary conditions. The numerical approach that we adopt is given in section 2.3. Lastly, applications of the algorithm can be found in sections 2.4, 2.5, 2.6, 2.7.

2 RECONSTRUCTION OF UNKNOWN QUANTITIES

Let $\mathbf{u}^\star$ be the measurement of a velocity field inside or around an object $\Omega^\bullet$, with boundary $\partial\Omega^\bullet$, and $\mathbf{u}$ the corresponding prediction using a physical model such as the Stokes equations with the appropriate boundary conditions. Then the process of inferring $\partial\Omega^\bullet$ such that $\mathbf{u}^\star \simeq \mathbf{u}(\partial\Omega^\bullet)$, is an inverse problem. In the following sections we study problems in which the hidden quantities of interest are the geometry of the object $\Omega^\bullet$, the unknown body forces $\mathbf{f}$, and the boundary conditions $\mathbf{g}$.

2.1 Shape of the Object

We first derive an algorithm that infers the shape of an object $\Omega^\bullet$, embedded in an image $I \subset \mathbb{R}^2$ with boundary ∂I. The image I depicts the signal $\mathbf{u}^\star$, acquired from a measurement (e.g. MRV, PIV), and we assume that the Stokes equation describes the fluid motion inside or around $\Omega^\bullet$.

Error Functional and its Shape Derivative. The discrepancy between $\mathbf{u}$ and $\mathbf{u}^\star$ is measured using the functional

$$\mathscr{J}(\mathbf{u}) = \frac{1}{2}\|\mathbf{u}-\mathbf{u}^\star\|^2_{L^2(I)} = \frac{1}{2}\int_I (\mathbf{u}-\mathbf{u}^\star)^2 \tag{1}$$

where $\mathbf{u} = (u_1, u_2)$ and $\mathbf{u}^\star = (u_1^\star, u_2^\star)$. Considering Ω as a deformable domain, subject to a speed field $\mathscr{V}$ [15, 16, 17] (figure 1), and using Reynolds transport theorem, we obtain the shape derivative of $\mathscr{J}$ along $\mathscr{V}$

$$\begin{aligned} D_{\mathscr{V}}\mathscr{J}(\mathbf{u}) &= \frac{1}{2}\Big(\int_I \big((\mathbf{u}-\mathbf{u}^\star)^2\big)' + \int_{\partial I}(\mathbf{u}-\mathbf{u}^\star)^2(\mathscr{V}\cdot\nu)\Big) \\ &= \int_I (\mathbf{u}-\mathbf{u}^\star)\mathbf{u}' = \int_\Omega (\mathbf{u}-\mathbf{u}^\star)\mathbf{u}' \end{aligned} \tag{2}$$

where $\mathbf{u}'$ is the shape derivative of $\mathbf{u}$ (in the 'direction' of $\mathscr{V}$), ν is the surface normal vector, $\mathscr{V} \equiv 0$ on ∂I, and $\mathbf{u}' \equiv 0$ everywhere except in Ω.

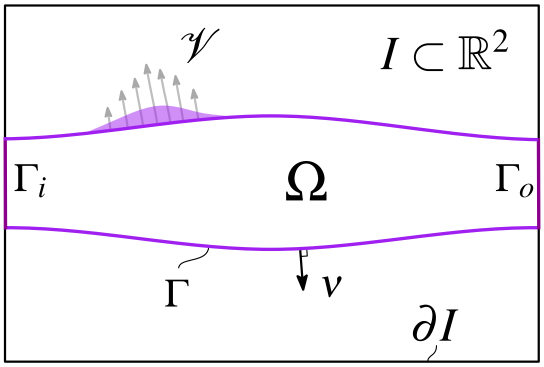


FIGURE 1: Notation for the Stokes problem in moving domains.

Stokes Problem for the Shape Derivatives. We start with the following Stokes problem

$$\begin{cases} -\Delta\mathbf{u} + \nabla p = \mathbf{f} & \text{in} \quad \Omega \\ \nabla\cdot\mathbf{u} = 0 & \text{in} \quad \Omega \\ \mathbf{u} = \mathbf{0} & \text{on} \quad \Gamma \\ -\partial_\nu\mathbf{u} + p\nu = \mathbf{g} & \text{on} \quad \Gamma_i \\ -\partial_\nu\mathbf{u} + p\nu = \mathbf{0} & \text{on} \quad \Gamma_o \end{cases} \tag{3}$$

where $\mathbf{u}$ is the velocity of the fluid, p the pressure, and $\partial_\nu \equiv \nu\cdot\nabla$. To find the corresponding shape derivatives problem we start with the weak form of (3) for test functions $\mathbf{v} \in H^1(\Omega)\times H^1(\Omega)$, with $\mathbf{v}$ also vanishing at Γ, and $q \in L^2(\Omega)$

$$\begin{aligned} &\int_\Omega \nabla\mathbf{v} : \nabla\mathbf{u} + \int_\Gamma \mathbf{v}\cdot(-\partial_\nu\mathbf{u} + p\nu) - \int_\Omega (\nabla\cdot\mathbf{v})p \\ &- \int_\Omega q(\nabla\cdot\mathbf{u}) = \int_\Omega \mathbf{v}\cdot\mathbf{f} + \int_{\Gamma_i}\mathbf{v}\cdot\mathbf{g} \quad . \end{aligned} \tag{4}$$

Using the Reynolds transport theorem on (4), we recover the problem for the shape derivatives $(\mathbf{u}', p')$,

$$\begin{cases} -\Delta\mathbf{u}' + \nabla p' = \mathbf{f}' & \text{in} \quad \Omega \\ \nabla\cdot\mathbf{u}' = 0 & \text{in} \quad \Omega \\ \mathbf{u}' = -\partial_\nu\mathbf{u}\,(\mathscr{V}\cdot\nu) & \text{on} \quad \Gamma \\ -\partial_\nu\mathbf{u}' + p'\nu = \mathbf{g}' & \text{on} \quad \Gamma_i \\ -\partial_\nu\mathbf{u}' + p'\nu = \mathbf{0} & \text{on} \quad \Gamma_o \end{cases} \quad . \tag{5}$$

If the forcing term $\mathbf{f}$ does not depend on perturbations of Ω, then $\mathbf{f}' \equiv 0$. Otherwise, if $\mathbf{f}$ is a field that changes when Ω deforms, additional physical knowledge is required to specify the form of $\mathbf{f}'$. The same applies for the boundary condition $\mathbf{g}$ and its shape derivative $\mathbf{g}'$.

At this point, we observe that one can choose a speed field $\mathscr{V}$ to solve the shape derivatives problem (5) for $\mathbf{u}'$. Proceeding to compute problem (5) for 'infinitely' many speed fields of the form $\mathscr{V} = \delta(x)\nu(x)$, for $x \in \partial\Omega$, where δ denotes the Dirac measure, leads to a brute-force estimation of the shape gradient $\mathscr{V}^\star$ of (2). Fortunately, the adjoint problem provides a much quicker method to find $\mathscr{V}^\star$.

Constructing the Adjoint Problem. Multiplying the shape derivatives problem (5) with test (adjoint) functions $(\mathbf{v},q)$ we obtain

$$\int_\Omega \mathbf{v}\cdot\big(-\Delta\mathbf{u}'+\nabla p'-\mathbf{f}'\big)-\int_\Omega q\big(\nabla\cdot\mathbf{u}'\big)=\mathbf{0} \quad . \tag{6}$$

After integrating by parts twice, and using the boundary conditions of problem (5), we find

$$\begin{aligned}\int_\Omega\big(-\Delta\mathbf{v}+\nabla q\big)\cdot\mathbf{u}' = \int_\Gamma\big(\partial_\nu\mathbf{v}\cdot\partial_\nu\mathbf{u}-q\,\partial_\nu\mathbf{u}\cdot\nu\big)\big(\mathscr{V}\cdot\nu\big)\\ -\int_{\Gamma_i}\mathbf{v}\cdot\mathbf{g}'+\int_\Omega\mathbf{v}\cdot\mathbf{f}' \quad .\end{aligned} \tag{7}$$

We see that

$$D_{\mathscr{V}}\mathscr{J}(\mathbf{u})(\mathscr{V})=\int_\Omega(\mathbf{u}-\mathbf{u}^\star)\mathbf{u}'=\int_\Omega\big(-\Delta\mathbf{v}+\nabla q\big)\cdot\mathbf{u}' \tag{8}$$

holds true for all $\mathbf{u}'$ satisfying the boundary conditions of problem (5) when $(\mathbf{v},q)$ is the solution of the following adjoint problem

$$\begin{cases}-\Delta\mathbf{v}+\nabla q=\mathbf{u}-\mathbf{u}^\star & \text{in} \quad \Omega\\ \nabla\cdot\mathbf{v}=0 & \text{in} \quad \Omega\\ \mathbf{v}=\mathbf{0} & \text{on} \quad \Gamma\\ -\partial_\nu\mathbf{v}+q\nu=\mathbf{0} & \text{on} \quad \Gamma_i\cup\Gamma_o\end{cases} \quad . \tag{9}$$

Therefore, due to (7) and (8) we find

$$\begin{aligned}\big\langle D_{\mathscr{V}}\mathscr{J},\mathscr{V}\cdot\nu\big\rangle_{\partial\Omega}=\big\langle\partial_\nu\mathbf{v}\cdot\partial_\nu\mathbf{u}-q\,\partial_\nu\mathbf{u}\cdot\nu,\ \mathscr{V}\cdot\nu\big\rangle_\Gamma\\ -\big\langle\mathbf{v}\cdot\widetilde{\mathbf{g}},\ \mathscr{V}\cdot\nu\big\rangle_{\Gamma_i}+\big\langle\mathbf{v}\cdot\widetilde{\mathbf{f}},\ \mathscr{V}\cdot\nu\big\rangle_{\partial\Omega}\end{aligned} \quad , \tag{10}$$

with the existence of $\widetilde{\mathbf{f}}$, $\widetilde{\mathbf{g}}$ implied by the Hadamard-Zolesio structure theorem for the shape derivative [15, 16, 17]. If the forcing and the boundary conditions do not depend on the movement of the domain Ω, we can consider $\mathbf{f}'\equiv 0$ and $\mathbf{g}'\equiv 0$, otherwise a physical relation must be established between them which depends on the problem at hand. We further restrict our interest to $\mathscr{V}$'s that can be defined through $\partial\Omega$'s normal vector ν and a scalar function ζ defined on $\partial\Omega$ (Hadamard parameterization). Thus, if $\mathbf{f}'\equiv 0$, $\mathbf{g}'\equiv 0$, and $\mathscr{V}=\zeta\nu$, (10) becomes

$$\big\langle D_{\mathscr{V}}\mathscr{J},\ \zeta\big\rangle_\Gamma=\big\langle\partial_\nu\mathbf{v}\cdot\partial_\nu\mathbf{u}-q\,\partial_\nu\mathbf{u}\cdot\nu,\ \zeta\big\rangle_\Gamma \quad . \tag{11}$$

Propagating the Boundary of Ω. Formula (11) provides the functional derivative that drives the geometric gradient flow minimizing $\mathscr{J}=\frac{1}{2}\|\mathbf{u}-\mathbf{u}^\star\|^2_{L^2(I)}$. In theory, one can simply write

$$x\mapsto x+\tau\zeta^\star(x)\nu(x) \quad \text{for every} \quad x\in\partial\Omega \tag{12}$$

to deduce the geometric flow for $\partial\Omega$ at given pseudotime $\tau>0$ and for $\zeta^\star=-\partial_\nu\mathbf{v}\cdot\partial_\nu\mathbf{u}+q\,\partial_\nu\mathbf{u}\cdot\nu$. However, (12), as is, leads to inefficient numerical algorithms because the phrase 'for every $x\in\partial\Omega$', hides the difficulty of locating a surface in space. Here, to tackle this problem, we use signed distance functions $\phi_\pm$ to represent general surfaces (closed curves) in $\mathbb{R}^2$. Then the object Ω and its boundary $\partial\Omega$ are identified with a particular function $\phi_\pm$ so that the following holds

$$\Omega=\big\{x\in\Omega:\ \phi_\pm(x)<0\big\} \quad , \quad \partial\Omega=\big\{x\in\Omega:\ \phi_\pm(x)=0\big\}.$$

Consequently, we approximate (12) by transporting $\phi_\pm$ under the speed field $\mathscr{V}^\star=\zeta^\star\nu$. The convection-diffusion initial value problem for $\phi_\pm(x,t)$ reads

$$\begin{cases}\partial_t\phi_\pm+\overset{\circ}{\mathscr{V}}\cdot\nabla\phi_\pm-\varepsilon\Delta\phi_\pm=0 & \text{in } I\times(0,\tau]\\ \phi_\pm=(\phi_\pm)_0 & \text{in } I\times\{t=0\}\end{cases} \tag{13}$$

where $(\phi_\pm)_0$ denotes the signed distance function of the current domain Ω, $0<\varepsilon\ll 1$ the diffusion coefficient, and $\overset{\circ}{\mathscr{V}}:=\overset{\circ}{\mathscr{V}}{}^\star: I\to\mathbb{R}\times\mathbb{R}$ is an extension of $\mathscr{V}^\star:\partial\Omega\to\mathbb{R}\times\mathbb{R}$. If we solve (13) for $\phi_\pm(x,\tau)$ we obtain the implicit representation of the perturbed domain Ω_τ, but to do so we first need to extend $\mathscr{V}^\star$ to the whole space of the image I. First, observe that $\mathscr{V}^\star=\zeta^\star\nu$, where both $\zeta^\star(x)$ and $\nu(x)$ are defined only on $\partial\Omega$. Therefore, to extend $\mathscr{V}^\star$ to I we first try to extend the normal vector ν and then the scalar function $\zeta^\star$.

Initially, the normal vector is only known on the surface $\partial\Omega$, but, due to the implicit characterization of $\partial\Omega$ through $\phi_\pm$, we can write

$$\overset{\circ}{\nu}(x)=\frac{\nabla\phi_\pm}{|\nabla\phi_\pm|}=\nabla\phi_\pm \quad , \quad x\in I \tag{14}$$

since $\phi_\pm$ is a distance function, i.e. $|\nabla\phi_\pm|=1$. Then, to compute an extension of the normal vector in I, it suffices to compute the gradient of the signed distance function.

The traditional method of computing the signed distance function is to solve a nonlinear hyperbolic equation (wave propagation), namely the Eikonal equation

$$|\nabla\phi_\pm(x)|=1 \quad \text{subject to} \quad \phi_\pm\big|_{\partial\Omega}=0 \quad , \quad x\in I. \tag{15}$$

One way to solve this problem is with level-set methods, which are well-studied and described in [18,19,20,21,22]. A numerical algorithm to solve (15), the *fast marching method*, is given in [23], having $n\log n$ complexity with n the total number of mesh points. The same algorithm is used to tackle the problem of both extending $\mathscr{V}^\star$ and propagating $\phi_\pm$ in [24]. We, however, choose a different approach, which relies on the heat equation [25]. The fundamental solution of the heat equation, the heat kernel

$$\Phi(x,t)=(4\pi t)^{-1}e^{-\|x\|^2/4t} \quad , \quad x\in\mathbb{R}^2,\ t>0 \tag{16}$$

solves $\partial_t \Phi - \Delta\Phi = 0$ for $\lim_{t\to 0}\Phi = \delta_x$. Varadhan [26] showed that the heat kernel is intimately related to distance by proving that

$$\lim_{t\to 0}\big(-2t\log\Phi(x-y,t)\big) = \|x-y\|^2 \quad , \tag{17}$$

and extended his results to Riemannian metrics by considering general diffusion tensors. The same author also proved relation (17), and its generalizations, from the viewpoint of stochastic diffusion processes in [27]. We justify the use of the heat equation for the approximation of $\phi_\pm$ using a theorem from Varadhan [26]. We slightly adapt this and restrict it to Euclidean distances such that

$$d(x,\partial\Omega) = \lim_{\tau_1\to 0}\big(-\frac{\sqrt{\tau_1}}{2}\log u(x,\tau_1)\big) \quad , \quad x\in I \tag{18}$$

where $d(x,\partial\Omega)$ is the Euclidean distance between any point $x\in I$ and $\partial\Omega$, and u is the solution of heat propagation away from $\partial\Omega$

$$\begin{cases} (I-\tau_1\Delta)u = 0 & \text{in } I \\ u = 1 & \text{on } \partial\Omega \end{cases} \quad . \tag{19}$$

Crane *et al.* [25] used Varadhan's results to implement a (smoothed) distance function computation algorithm which they called the '*heat method*'. Here, we adapt this algorithm to compute *signed* distance functions $\phi_\pm$. Therefore, to compute $\phi_\pm$ we first solve (19) for $\tau_1 \ll 1$ [1]. Finally, $\phi_\pm$ is given by

$$\begin{cases} \Delta\phi_\pm = \nabla\cdot X & \text{in } I \\ \partial_\nu\phi_\pm = X\cdot\nu & \text{on } \partial I \\ \phi_\pm = 0 & \text{on } \partial\Omega \end{cases} \quad , \quad X = -\mathrm{sgn}(\psi)\frac{\nabla u}{|\nabla u|} \tag{20}$$

with X the normalized heat flux and ψ a signed function so that $\psi(x)$ is negative for points x in Ω and positive for points x outside $\overline{\Omega}$. This intermediate step (the solution of two Poisson problems (19)-(20) instead of one) is taken to ensure that $|\nabla\phi_\pm| = 1$. Accordingly, $\phi_\pm$ leads to a natural extension of $\partial\Omega$'s normal vector ν to I through

$$\mathring{\nu}(x) = \mathrm{sgn}(\psi)X(x) \quad , \quad x\in I \tag{21}$$

where the term $\mathrm{sgn}(\psi)$ is used to correct the orientation of $\mathring{\nu}$ so that it points away from $\partial\Omega$. Next, we use the extension $\mathring{\nu}$ to extend $\zeta^\star$ to $\mathring{\zeta} := \mathring{\zeta}^\star$, with the help of the convection-diffusion problem

$$\begin{cases} \partial_t\mathring{\zeta} + \mathring{\nu}\cdot\nabla\mathring{\zeta} - \varepsilon\Delta\mathring{\zeta} = 0 & \text{in} \quad I\times(0,\tau_\zeta] \\ \mathring{\zeta} = \zeta^\star & \text{on} \quad \partial\Omega\times(0,\tau_\zeta] \\ \mathring{\zeta} \equiv 0 & \text{in} \quad I\times\{t=0\} \end{cases} \quad . \tag{22}$$

In other words, we convect ζ along the predefined $\mathring{\nu}$-streamlines and add isotropic diffusion for stabilization (note that $\mathring{\nu}$-streamlines may have complicated behaviour and, without diffusion, the problem may not have a unique solution). The timestep τ_ζ is taken large enough to reach a steady-state.

Finally, we recast the initial value problems (13) and (22) into their corresponding boundary value problems. This is possible due to the linearity of the problem and the fact that the time-dependent solution does not interest us here. Therefore, instead of problem (13) we solve the elliptic problem

$$\big(I-\tau A_{\mathscr{V}}\big)\phi_\pm = (\phi_\pm)_0 \quad \text{in } I \tag{23}$$

where $A_{\mathscr{V}}\phi_\pm := -\varepsilon\Delta\phi_\pm + \mathring{\mathscr{V}}\cdot\nabla\phi_\pm$. Note that (23) propagates $\phi_\pm$ from pseudotime $t=0$ to $t=\tau$ under the speed field $\mathring{\mathscr{V}}$. In the same spirit,

$$\begin{cases} \big(I-\tau_\zeta A_\nu\big)\mathring{\zeta} = 0 & \text{in } I \\ \mathring{\zeta} = \zeta^\star & \text{on } \partial\Omega \end{cases} \tag{24}$$

replaces (22), with $A_\nu\mathring{\zeta} := -\varepsilon\Delta\mathring{\zeta} + \mathring{\nu}\cdot\nabla\mathring{\zeta}$, and describes the propagation of ζ, away from $\partial\Omega$ and along $\mathring{\nu}$, from pseudotime $t=0$ to $t=\tau_\zeta$.

To summarize, we propagate the boundary of Ω by executing three tasks:

1. Compute the heat flux X away from $\partial\Omega$ by solving (19) for $\tau_1 \ll 1$, in order to obtain $\mathring{\nu} = \mathrm{sgn}(\psi)X$.
2. Convect the shape gradient $\zeta^\star$ along $\mathring{\nu}$, using (24), in order to obtain $\mathring{\mathscr{V}} = \mathring{\zeta}\mathring{\nu}$.
3. Convect the signed distance function $\phi_\pm$ along $\mathring{\mathscr{V}}$, using (23) with timestep τ, in order to obtain the new boundary $\partial\Omega_\tau$.

Otherwise, if reinitialization of $\phi_\pm$ is needed, we additionally solve (20) immediately after Step 1. Algorithm 1 provides the instructions in order to obtain $\mathring{\mathscr{V}}$.

```
Algorithm 1: Compute_Optimal_Speed_Field $\mathring{\mathscr{V}}$
  Input: $\mathbf{u}, \mathbf{u}^\star, \phi_\pm, \varepsilon$
  begin
    1. $\mathbf{v}, q \leftarrow$ Adjoint_Problem $(\mathbf{u}, \mathbf{u}^\star, \phi_\pm)$   (eq. (9))
    2. $\zeta^\star \leftarrow$ Shape_Gradient $(\mathbf{u}, \mathbf{v}, q, \phi_\pm)$   (eq. (11))
    3. $\mathring{\nu} \leftarrow$ Normal_Vector_Extension $(\phi_\pm)$   (eq. (19))
    4. $\mathring{\zeta} \leftarrow$ Extend_Function $(\zeta^\star, \mathring{\nu}, \phi_\pm, \varepsilon)$   (eq. (24))
  Output: $\mathring{\mathscr{V}} \equiv \mathring{\zeta}\mathring{\nu}$
```

All of the above methodologies result in an algorithm, compactly presented as algorithm 2, which uses the Stokes problem to infer the shape of an object $\Omega^\bullet$, embedded in an image I.

[1] For the numerical problem we take $\tau_1 = 10h^2$ with h the mesh size [25].

Algorithm 2: Shape reconstruction from (Stokes) flow images.

```
Data input: images ($I$) depicting the components of $\mathbf{u}^\star$ (measured flow field).
Model input: $\mathbf{f}$ (forcing term), $\mathbf{g}$ (inlet natural b.c.), $\varepsilon$ (level-set diffusion coefficient), $(\phi_\pm)_{i=0}$ (initial level-set).
Optimization input: $\tau_{i=0}$ (initial timestep), $\varepsilon_{\text{tol}}$ (tolerance), $i_{\max}$ (maximum iterations), $\ell_{\max}$ (maximum line searches).
begin
    Zero-th iteration: Set $i \leftarrow 0$, solve Stokes problem (3) for $\mathbf{u}_i$ in $\Omega_i$ (defined by $(\phi_\pm)_i$),
    and evaluate the discrepancy functional (1), setting $\mathscr{J}_i \leftarrow \mathscr{J}(\mathbf{u}_i)$.
    Geometric gradient flow:
    while $\left(\varepsilon \geq \varepsilon_{tol} \ \ and \ \ i \leq i_{max}\right)$ do
        1. $\mathring{\mathscr{V}}_i \leftarrow$ Compute_Optimal_Speed_Field $(\mathbf{u}_i, \mathbf{u}^\star, (\phi_\pm)_i, \varepsilon)$      (algorithm 1)
        Set $\ell_{\text{search}} \leftarrow 0$ and repeat
            2. $(\phi_\pm)_{\text{search}} \leftarrow$ Gradient_Flow$\big((\phi_\pm)_i, \mathring{\mathscr{V}}_i, \tau_i\big)$      (equation (23))
            3. $\mathbf{u}_{\text{search}} \leftarrow$ Stokes_Problem$\big(\mathbf{f}, \mathbf{g}, (\phi_\pm)_{\text{search}}\big)$      (equation (3))
            4. $\mathscr{J}_{\text{search}} \leftarrow \mathscr{J}(\mathbf{u}_{\text{search}})$      (equation (1))
            $\tau_i \leftarrow \tau_i/2$ if $(\mathscr{J}_{\text{search}} \geq \mathscr{J}_i)$;  $\ell_{\text{search}} \leftarrow \ell_{\text{search}} + 1$
        until $\left(\mathscr{J}_{search} < \mathscr{J}_i \ \ or \ \ \ell_{search} > \ell_{max}\right)$
        $i \leftarrow i+1$
        if $\left(\mathscr{J}_{search} < \mathscr{J}_i\right)$ then
            $(\phi_\pm)_i \leftarrow (\phi_\pm)_{\text{search}}$, $\mathbf{u}_i \leftarrow \mathbf{u}_{\text{search}}$, $\mathscr{J}_i \leftarrow \mathscr{J}_{\text{search}}$, $\tau_i \leftarrow 3\tau_i/2$
        else
            terminate (line search could not find a better solution)
        $\varepsilon \leftarrow |\mathscr{J}_i - \mathscr{J}_{i-1}|/\mathscr{J}_i$
    Output: Reconstructed domain $\Omega^\bullet$ minimizing $\mathscr{J}$, and filtered velocity magnitude $\mathbf{u} \simeq \mathbf{u}^\star$.
```

2.2 Forcing and Boundary Conditions

The Stokes problem (3) involves additional parameters, such as the forcing term $\mathbf{f}$ and the inlet natural boundary condition $\mathbf{g}$. If these parameters cannot be specified with some degree of certainty, the physical model may not be able to interpret the experimental velocity fields $\mathbf{u}^\star$. In turn, this implies that either the model velocity $\mathbf{u}$ will be incapable of approximating the experimental measurement $\mathbf{u}^\star$ (no matter the shape of Ω), or that matching $\mathbf{u}$ with $\mathbf{u}^\star$ will probably produce a reconstructed domain that does not approximate $\Omega^\bullet$. This motivates the formulation of a combined optimization problem, in which the aforementioned parameters are included in the reconstruction process, using similar, yet simpler, methods than those of section 2.1.

Euler–Lagrange System. To evaluate the sensitivity of the inverse problem to the model parameters, we start by writing the functional

$$\begin{aligned}\mathscr{J}(\mathbf{u},\mathbf{v},p,q;\mathbf{f},\mathbf{g}) &= \frac{1}{2}\int_I (\mathbf{u}-\mathbf{u}^\star)^2 - \int_\Omega \nabla\mathbf{v} : \nabla\mathbf{u} - \int_{\Gamma_i} \mathbf{v}\cdot\mathbf{g} \\ &- \int_\Gamma \mathbf{v}\cdot(-\partial_\nu \mathbf{u} + p\nu) + \int_\Omega (\nabla\cdot\mathbf{v})p + \int_\Omega q(\nabla\cdot\mathbf{u}) + \int_\Omega \mathbf{v}\cdot\mathbf{f} \\ &- \mathscr{N}(\mathbf{u},\mathbf{v},q,\eta) = \mathbf{0} \quad ,\end{aligned} \tag{25}$$

where

$$\mathscr{N}(\mathbf{u},\mathbf{v},q,\eta) = \int_\Gamma (-\partial_\nu \mathbf{v} + q\nu)\cdot(\mathbf{u}-\mathbf{0}) + \eta\ \mathbf{v}\cdot(\mathbf{u}-\mathbf{0}) \tag{26}$$

is the Nitsche term for the weak imposition of the no-slip (Dirichlet) boundary condition with penalty parameter η [28, 29], $\mathbf{v}$ is the adjoint velocity, and q is the adjoint pressure.

The first variations of $\mathscr{J}$ with respect to the parameters $(\mathbf{f},\mathbf{g})$ are

$$\frac{d}{d\tau}\mathscr{J}(\ldots,\mathbf{f}+\tau\mathbf{f}',\ldots)\Big|_{\tau=0} = \left\langle \frac{\partial\mathscr{J}}{\partial\mathbf{f}}, \mathbf{f}' \right\rangle_\Omega = \left\langle \mathbf{v},\ \mathbf{f}' \right\rangle_\Omega \tag{27a}$$

$$\frac{d}{d\tau}\mathscr{J}(\ldots,\mathbf{g}+\tau\mathbf{g}')\Big|_{\tau=0} = \left\langle \frac{\partial\mathscr{J}}{\partial\mathbf{g}}, \mathbf{g}' \right\rangle_{\Gamma_i} = \left\langle -\mathbf{v},\ \mathbf{g}' \right\rangle_{\Gamma_i} \tag{27b}$$

where $\mathbf{f}'$ and $\mathbf{g}'$ are perturbations of $\mathbf{f},\mathbf{g}$, respectively, and are not to be confused with the shape derivatives $\mathbf{f}'$, $\mathbf{g}'$, which are zero.

Regularization.

Usually there are many different sets of parameters $(\Omega,\mathbf{f},\mathbf{g})$, producing solutions $\mathbf{u}$ that minimize $\mathscr{J}$. For example, it is possible that the solution $\mathbf{u}$ is invariant under certain perturbations of the model parameters, or that two parameters have a counteracting effect on $\mathbf{u}$. Numerically, this means that the solution to the optimization problem will be particularly sensitive to the optimization parameters (step-size, initial guess etc.), and difficult to converge. Therefore, we need to regularize the problem with one of the methods below.

Variational Techniques. The need for regularization arises from the assumption that $\mathbf{f},\mathbf{g}$, for example, are functions in L^2, i.e. square-integrable. As L^2 is a very rich space of 'wild' functions, we would like to restrict $\mathbf{f}, \mathbf{g}$ to a subspace of L^2, of 'tamable' functions: for instance, H^k, $k > 0$, the space of square-integrable functions with k square-integrable derivatives, for which $H^k \subset L^2$. For any function u to be a member of H^k, it suffices to show that $\|u\|_{H^k} < \infty$. Therefore, to regularize $\mathbf{f}$ we may add the term

$$\mathscr{R}(\mathbf{f}, \alpha) = \frac{\alpha}{2}\|\mathbf{f}-\mathbf{f}_0\|^2_{H^1(\Omega)} = \frac{\alpha}{2}\Big(\int_I (\mathbf{f}-\mathbf{f}_0)^2 + \nabla(\mathbf{f}-\mathbf{f}_0)^2\Big)$$

to the functional (25), where $\alpha > 0$ is a weighting constant, and $\mathbf{f}_0$ is the prior knowledge; in other words, we expect $\mathbf{f}$ to be close to $\mathbf{f}_0$. Adding $\mathscr{R}(\mathbf{f}, \alpha)$ to (25), (27a) becomes

$$\Big\langle \frac{\partial \mathscr{J}}{\partial \mathbf{f}}, \mathbf{f}' \Big\rangle_\Omega = \Big\langle -\alpha\Delta\mathbf{f} + \alpha\mathbf{f} + \big(\mathbf{v} - \alpha\mathbf{f}_0 + \alpha\Delta\mathbf{f}_0\big),\, \mathbf{f}' \Big\rangle_\Omega \quad (28)$$

yielding a screened Poisson problem for $\mathbf{f}$. We could instead consider only the H^1-seminorm

$$\mathscr{R}(\mathbf{f}, \alpha) = \frac{\alpha}{2}|\mathbf{f}-\mathbf{f}_0|^2_{H^1(\Omega)} = \frac{\alpha}{2}\Big(\int_I \nabla(\mathbf{f}-\mathbf{f}_0)^2\Big)$$

and find

$$\Big\langle \frac{\partial \mathscr{J}}{\partial \mathbf{f}}, \mathbf{f}' \Big\rangle_\Omega = \Big\langle -\alpha\Delta\mathbf{f} + \big(\mathbf{v} + \alpha\Delta\mathbf{f}_0\big),\, \mathbf{f}' \Big\rangle_\Omega \quad (29)$$

yielding a Poisson problem for $\mathbf{f}$. In both of these cases we smooth $\mathbf{f}$ by solving a Poisson problem. The same regularization techniques apply to $\mathbf{g}$. Regularization methods for ill-posed variational inverse problems with applications to image processing are reviewed in [6].

Gaussian Random Fields. Interestingly, we can reach appropriate regularization norms $\mathscr{R}$ by formulating a stochastic problem. We first consider a Hilbert space (e.g. L^2) of Gaussian random fields (functions). The Gaussian measure γ has the property that its finite-dimensional projections are multivariate Gaussian distributions, and it is uniquely defined by its mean $m \in L^2$, given by

$$m = \mathbb{E}x = \int_{L^2} x\, \gamma(dx) \quad , \quad (30)$$

and its covariance $C : L^2 \times L^2 \to \mathbb{R}$, for any pair $h, h' \in L^2$, given by

$$C(h, h') = \int_{L^2} \langle h, x\rangle \langle h', x\rangle\, \gamma(dx) \quad . \quad (31)$$

The above (Bochner) integrals define integration over the function space L^2, under the measure γ. We also define the covariance operator $\mathcal{C} : L^2 \to L^2$ as

$$\mathcal{C}h = \int_{L^2} x\langle h, x\rangle\, \gamma(dx) \quad (32)$$

and note that $\langle \mathcal{C}h, h'\rangle = C(h, h')$. What we have written above is simply a generalization of Gaussian measures in Hilbert spaces, and fortunately makes sense because the (Bochner) integrals are well defined due to Fernique's theorem [30]. As we mentioned before, to regularize the problem we restrict our attention to a subspace of L^2 having smoother functions (e.g. H^1). Now, we will see that an appropriate subspace can also arise by choosing a covariance operator $\mathcal{C}$ (which acts as averaging/smoothing).

As any Gaussian measure in L^2 is uniquely defined by its mean and covariance operator, we write $\gamma = \mathcal{N}(m, \mathcal{C})$. It can be shown that there is a natural Hilbert space H_γ that corresponds to γ and that [31]

$$H_\gamma = \sqrt{\mathcal{C}}(L^2) \quad ,$$

i.e. any function in H_γ belongs to the image of $\sqrt{\mathcal{C}}$. The corresponding inner product

$$\langle h, h'\rangle_\gamma = \langle \mathcal{C}^{-1/2}h,\, \mathcal{C}^{-1/2}h'\rangle \quad (33)$$

defines the norm $\|h\|^2_\gamma$, which is the variance of h. Therefore, an appropriate subspace of L^2 is H_γ, given that we can specify a covariance operator for, e.g. the behaviour of the model parameter $\mathbf{f}$ as a Gaussian random field. This is how prior knowledge is now inserted into our formulation to mitigate the ill-posedness of the inverse problem. For example, if we take $\mathcal{C} = -\sigma^2\Delta^{-1}$, for $\mathbb{R} \ni \sigma > 0$ and consider functions u in Ω such that $u\big|_{\partial\Omega} = 0$, from (33) we find that [2]

$$\langle h, h'\rangle_\gamma = \frac{1}{\sigma^2}\langle \nabla h, \nabla h'\rangle$$

with the inner product generating the H^1-seminorm, and thus $H_\gamma(\Omega) = H^1_0(\Omega)$. More examples can be found in [35, Chapter 7.21]. This approach, in which the appropriate norm $\mathscr{R}$ is implicitly defined through $\mathcal{C}$ by considering randomness in the model parameters, is discussed in [5] and provides an immediate link between variational approaches and Bayesian inference.

2.3 Numerics

To solve the previous problems numerically, we adopt an essentially *meshless* method [36], known as the fictitious domain or immersed boundary finite element method. This method presents

[2] In general, we can consider the fractional Laplacian operator $-\Delta^s$, for $s > 0$ [32,33], or a covariance kernel function (e.g. squared exponential) as in machine learning [34].

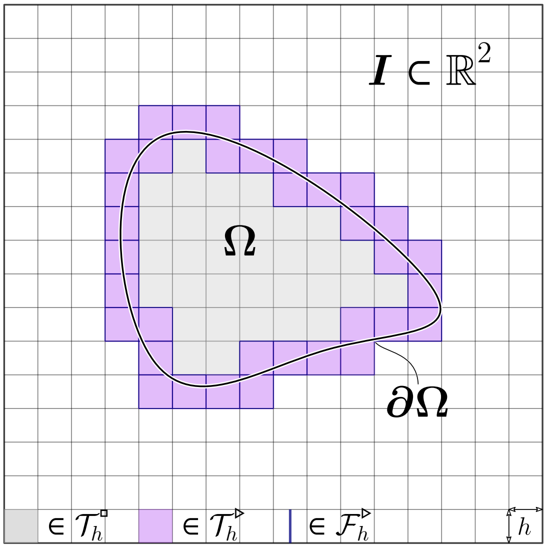

$\boldsymbol{I} \subset \mathbb{R}^2$
Ω
$\partial\Omega$
$\in \mathcal{T}_h^{\square}$
$\in \mathcal{T}_h^{\triangleright}$
$\in \mathcal{F}_h^{\triangleright}$
h

FIGURE 2: Notation for the fictitious domain method, also known as the immersed finite element method.

several advantages over traditional mesh-dependent methods when working with complicated, moving geometries. In particular, we implement the fictitious domain cut-cell finite element method, introduced by Burman and Hansbo [37,38] for the Poisson problem, and extended by Massing *et al.* [29] to the Stokes problem. The method uses Nitsche's [28] approach: a consistent and stable way to satisfy Dirichlet boundary conditions on $\partial\Omega$ (the immersed boundary), in a weak sense.

Subsequently, to discretize the problem, we define $\mathcal{T}_h$ to be a tessellation of I produced by square cells (pixels) $K \in \mathcal{T}_h$, having sides of length h. We also define the set of cut-cells $\mathcal{T}_h^{\triangleright}$ consisting of the cells that are cut by the boundary $\partial\Omega$, and $\mathcal{T}_h^{\square}$ the set of cells that are found inside Ω and which remain intact (not cut) (see figure 2). We assume that the boundary $\partial\Omega$ is well-resolved, i.e. $\ell_{\partial\Omega}/h \gg 1$ with $\ell_{\partial\Omega}$ the smallest length scale of $\partial\Omega$. For the detailed assumptions on $\partial\Omega$ we cite [38]. To every cell K a bilinear quadrilateral finite element $\mathcal{Q}_1$ is assigned, and this generates the discretized space

$$V_h = \left\{ u_h \in C^0(I) \text{ such that } u_h|_K \in \mathcal{Q}_1 \text{ for all } K \in \mathcal{T}_h \right\} \qquad (34)$$

where $\mathcal{Q}_1 = \left\{ \sum_\ell c_\ell \, p_\ell(x) q_\ell(y),\ p_\ell, q_\ell \text{ polynomials of deg.} \leq 1 \right\}$. We choose to weakly satisfy the zero-Dirichlet (no-slip) condition on Γ using Nitsche's method, and write

$$\begin{aligned} &\int_\Omega \nabla\mathbf{v} : \nabla\mathbf{u} + \int_\Gamma \mathbf{v}\cdot(-\partial_\nu \mathbf{u} + p\nu) - \int_\Omega (\nabla\cdot\mathbf{v})p \\ &-\int_\Omega q(\nabla\cdot\mathbf{u}) + \mathscr{N}(\mathbf{u},\mathbf{v},q,\eta) = \int_\Omega \mathbf{v}\cdot\mathbf{f} + \int_{\Gamma_i} \mathbf{v}\cdot\mathbf{g} \qquad (35) \end{aligned}$$

where $\eta = \gamma/h$ with γ the penalization constant. The weak form (35) now holds for all $\mathbf{v} \in H^1(\Omega) \times H^1(\Omega)$ ($\mathbf{v}$ does not have to vanish on Γ), and $q \in L^2(\Omega)$. Next, we express the problem in the discretized space V_h, defined by (34), and let both the velocity components and the pressure be elements of V_h. As this is known to violate the inf-sup condition [39] for the Stokes problem, leading to an unstable numerical scheme, we include an additional stabilization term, namely $c(\mathbf{u}_h, p_h, q_h)$, to stabilize the pressure[3]. The discretized problem then takes the form: find $(\mathbf{u}_h, p_h)$ such that

$$\begin{aligned} &a(\mathbf{u}_h,\mathbf{v}_h) + b(\mathbf{u}_h,q_h) + b(\mathbf{v}_h,p_h) \\ &+ j_\mathbf{u}(\mathbf{u}_h,\mathbf{v}_h) + j_p(q_h,p_h) + c(\mathbf{u}_h,p_h,q_h) = i(\mathbf{v}_h,q_h) \qquad (36) \end{aligned}$$

holds for all $\mathbf{v}_h \in V_h \times V_h$ and $q_h \in V_h$, with

$$\begin{aligned} a(\mathbf{u}_h,\mathbf{v}_h) &:= \int_\Omega \nabla\mathbf{v}_h : \nabla\mathbf{u}_h + \int_\Gamma -(\mathbf{v}_h\cdot\partial_\nu\mathbf{u}_h) \\ &\quad -(\partial_\nu\mathbf{v}_h\cdot\mathbf{u}_h) + \frac{\gamma}{h}(\mathbf{v}_h\cdot\mathbf{u}_h) && (37a) \\ b(\mathbf{u}_h,q_h) &:= -\int_\Omega q_h(\nabla\cdot\mathbf{u}_h) + \int_\Gamma (q_h\nu)\cdot\mathbf{u}_h && (37b) \\ j_\mathbf{u}(\mathbf{u}_h,\mathbf{v}_h) &:= \sum_{F\in\mathcal{F}_h^{\triangleright}} \int_F \gamma_1 h\,[\nabla\mathbf{u}_h][\nabla\mathbf{v}_h] && (37c) \\ j_p(q_h,p_h) &:= \sum_{F\in\mathcal{F}_h^{\triangleright}} \int_F \gamma_2 h^3\,[\nabla p_h][\nabla q_h] && (37d) \\ c(\mathbf{u}_h,p_h,q_h) &:= -\int_\Omega \beta h^2\,(-\Delta\mathbf{u}_h + \nabla p_h) : \nabla q_h && (37e) \\ i(\mathbf{v}_h,q_h) &:= \int_\Omega \mathbf{v}_h\cdot\mathbf{f} + \int_{\Gamma_i}\mathbf{v}_h\cdot\mathbf{g} - \int_\Omega \beta h^2\,(\mathbf{f}\cdot\nabla q_h) \quad . && (37f) \end{aligned}$$

We briefly explain the purpose of the above forms. First, a is the discretized Laplacian, supplemented by its boundary conditions, and b describes the coupling between velocity and pressure so that the incompressibility condition is satisfied. Thus, a, b and the first two integrals of the functional i, compose the weak form of the Stokes equation for the Nitsche problem, given by (35). Then, $j_\mathbf{u}, j_p$ are the cut-cell ghost-penalization terms [37,38,29], with the accompanying constants γ_1, γ_2, for the velocity and the pressure, respectively. Finally, c and the last integral appearing in i, are the pressure stabilization terms, with constant β. Note that $\Delta\mathbf{u}_h \equiv \mathbf{0}$ due to the bilinear finite elements $\mathcal{Q}_1$, which means that the pressure stabilization is not consistent (it would have been consistent for $\mathcal{Q}_2$ elements).

To compute the integrals we use standard Gaussian quadrature for cells $K \in \mathcal{T}_h^{\square}$, while for cut-cells $K \in \mathcal{T}_h^{\triangleright}$, where integration must be considered only for the intersection $K \cap \Omega$, we use Mirtich's approach [40]. Mirtich's approach relies on the divergence theorem and simply replaces the integral over $K \cap \Omega$ with an integral over $\partial\left(K\cap\Omega\right)$. The boundary integral on $\partial\left(K\cap\Omega\right)$ is then easily computed using one-dimensional Gaussian quadrature [41].

Lastly, we consider a basis $\left\{\varphi_i\right\}_{i=1}^n$ of V_h and write $\mathbf{u}_h = \sum_{j=1}^n \mathbf{u}_j\varphi_j$, $p_h = \sum_{i=1}^n p_j\varphi_j$, $\mathbf{v}_h = \sum_{i=1}^n \varphi_i$ and $q_h = \sum_{i=1}^n \varphi_i$ to find

[3] Known as pressure-stabilized Petrov–Galerkin (PSPG) method.

that the solution of the system

$$\begin{pmatrix} A & B \\ C & D \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ p \end{pmatrix} = \begin{pmatrix} F \\ G \end{pmatrix} \tag{38}$$

where

$$\begin{aligned} A_{ij} &:= a(\varphi_i, \varphi_j) + j_{\mathbf{u}}(\varphi_i, \varphi_j) & B_{ij} &:= b(\varphi_i, \varphi_{\mathbf{j}}) \\ C_{ij} &:= b(\varphi_j, \varphi_i) & D_{ij} &:= j_p(\varphi_i, \varphi_j) + c(\varphi_i, \varphi_j) \\ F_i &:= \langle \varphi_i, \mathbf{f} \rangle_\Omega + \langle \varphi_i, \mathbf{g} \rangle_{\Gamma_i} & G_i &:= -\langle \beta h^2 \mathbf{f}, \nabla \varphi_i \rangle_\Omega \end{aligned}$$

is a finite dimensional approximation of the problem (36). We solve the above system using the Schur complement; with an iterative solver (LGMRES) for the outer loops, and a direct sparse solver (UMFPACK) for the inner loops, both implemented in Python's SciPy library [42].

2.4 Shape Reconstruction with the Poisson Problem

We now test the shape inference algorithm (algorithm 2) for the reduced problem of Poiseuille flow through a starfish-shaped pipe (figure 3a). This corresponds to a Poisson problem (instead of a Stokes problem) with zero-Dirichlet boundary conditions on the wall. For this test case we pick Ω_0 (the initial guess) to be far enough from $\Omega^\bullet$ (the ground truth) to test the robustness of the algorithm. However, for real applications, it would be more practical to use a general image segmentation algorithm for Ω_0. The rest of the parameters needed to execute the algorithm and to solve the numerical problems, are summarized in table 1. We create a synthetic MRV image by first solving the Poisson problem in the ground truth domain $\Omega^\bullet$, in order to obtain the ground truth velocity field $u^\bullet$, and then corrupt the image by adding Gaussian white noise. We define the signal-to-noise ratio (SNR) as

$$\mathrm{SNR} := \frac{\mu^2}{\sigma^2}, \quad \text{with} \quad \mu = \frac{1}{|\Omega^\bullet|} \int_{\Omega^\bullet} u^\bullet$$

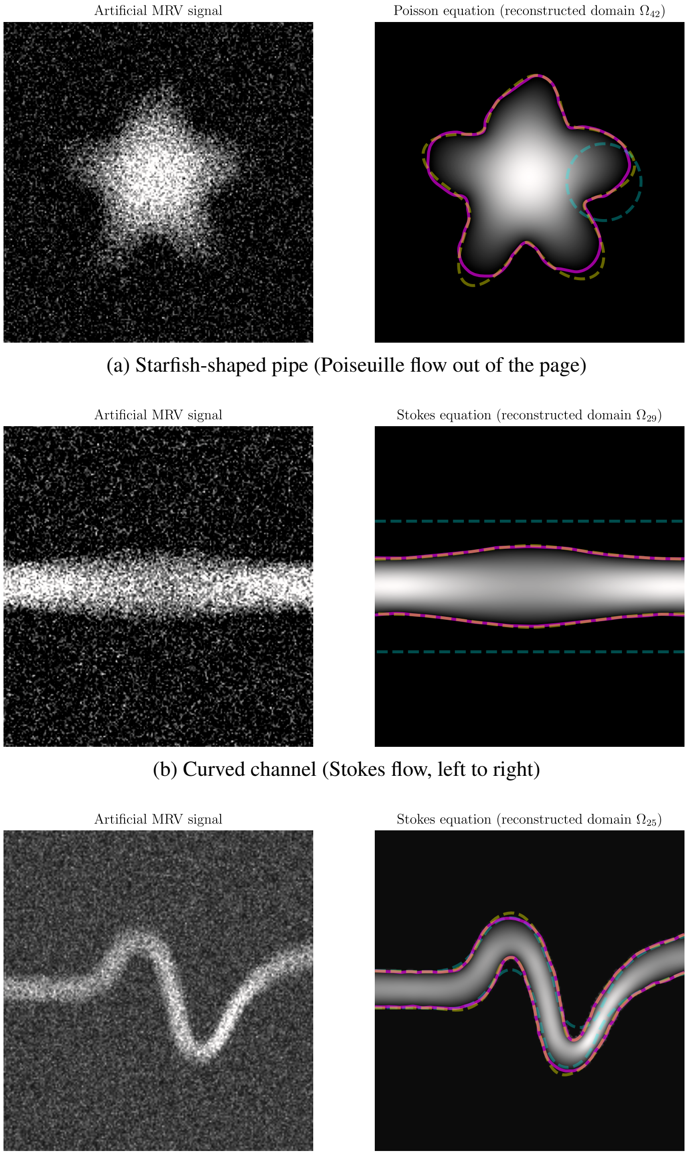
where σ^2 is the variance of the Gaussian white noise, and $|\Omega^\bullet|$ the area of $\Omega^\bullet$. When the ground truth is known, we also define the L^2-norm reconstruction error as

$$\mathscr{E} := \frac{\|u - u^\bullet\|_{L^2(I)}}{\|u^\bullet\|_{L^2(I)}} \quad .$$

The synthetic MRV image has an SNR of 2.5, and the geometric flow converges in 42 iterations with $\mathscr{E} \simeq 4.0\%$ (figure 3a).

2.5 Shape Reconstruction with the Stokes Problem

Next, we test the algorithm for 2D steady Stokes flow (figures 3b, 3c) in two different domains: i) a curved channel, and ii) a blood vessel dummy. We add Gaussian white noise such that SNR $= 2.5$. For the curved channel, we start the geometric flow from a rectangular channel, and after 29 iterations we recover a reconstruction closely resembling the ground truth with $\mathscr{E} \simeq 3.5\%$ (figure 3b). For the dummy blood vessel, we start with a domain Ω_0 that resembles $\Omega^\bullet$, but perturbed so that it is visibly narrower before the outlet (as if there was an image segmentation error), and obtain the reconstructed shape after 25 iterations with $\mathscr{E} \simeq 7.1\%$ (figure 3c). We observe that the reconstruction deteriorates near neighbourhoods of Γ (the walls) where shape perturbations have a minor effect on $\mathscr{J}$ (where velocity and its gradients vanish rapidly). This is due to the local stiffness of the optimization problem and the considerable loss of information due to noise (the local SNR is very low). Fortunately, this also means that these local geometric errors will have a small effect on the local reconstructed velocity field and its gradients.


Artificial MRV signal
Poisson equation (reconstructed domain Ω_{42})
Artificial MRV signal
Stokes equation (reconstructed domain Ω_{29})
Artificial MRV signal
Stokes equation (reconstructed domain Ω_{25})

(a) Starfish-shaped pipe (Poiseuille flow out of the page)

(b) Curved channel (Stokes flow, left to right)

(c) Dummy blood vessel (Stokes flow, left to right)

FIGURE 3: Inferring the shape of an object $\Omega^\bullet$ from synthetic MRV data (light intensity denotes velocity magnitude). The dashed cyan line denotes the initial guess Ω_0, the solid purple line the reconstructed domain, and the dashed yellow line the ground truth (where the lines overlap they seem orange).

TABLE 1: Input parameters for the Stokes shape inference algorithm (algorithm 2).

Model Parameters				Optimization Parameters				Numerical Parameters				
$\mathbf{f}$	$\mathbf{g}$	ε	τ_ζ	$\tau_{i=0}$	ε_{tol}	$i_{\max}$	$\ell_{\max}$	Res. (n)	γ	γ_1	γ_2	β
Starfish-shaped pipe (Poiseuille flow)												
$(1, \cdot)$	$\cdot$	0.0125	10^6	0.1	10^{-6}	100	10	200^2 px	10	1	$\cdot$	$\cdot$
Curved channel (Stokes flow)												
(0, 0)	$(2.5{\cdot}10^4, 0)$	0.125	10^6	0.05	10^{-6}	100	10	200^2 px	10	1	0.05	0.2
Dummy blood vessel (Stokes flow)												
(0, 0)	$(2.5{\cdot}10^4, 0)$	0.0125	10^6	0.005	10^{-6}	100	10	200^2 px	10	1	0.05	0.2

2.6 Shape and Forcing Reconstruction with the Poisson Problem

In this case, the MRV data [43] $u^\star$ (128^2 pixels) correspond to a real experiment of pipe flow. We use the regularized objective functional

$$\mathscr{J} = \frac{1}{2\sigma_u^2}\|u - u^\star\|^2_{L^2(I)} + \frac{1}{2\sigma_f^2}|f - f_0|^2_{H_0^1(I)} \tag{39}$$

with $\sigma_u = 0.001$, $\sigma_f = 10$, and $f_0 \equiv 0.1$. We use again algorithm 2 and include f (its gradient given by (27a)) in the same line search that we use for $\phi_\pm$ (the domain), and under the same pseudo-timestep τ. Figure 5a presents a flattened and non-axisymmetric velocity profile, characterising undeveloped flow. For Poiseuille flow, f would correspond to $-\mu^{-1}\,\partial p/\partial x$, with μ the dynamic viscosity, $\partial p/\partial x$ the pressure gradient, and the flow would be axisymmetric and developed. As undeveloped flow cannot be described by a Poisson equation, the forcing term f should be considered as a model-error parameter, instead of a quantity with physical meaning. The domain and the forcing term are reconstructed after 40 iterations.

2.7 Shape and Boundary Condition Reconstruction with the Stokes Problem

Finally, we solve the same problem as that in section 2.5 (curved channel), but now we also treat the inlet boundary condition $\mathbf{g}$ as an unknown. To be precise, we seek the pair $(\Omega, \mathbf{g})$ that minimizes

$$\mathscr{J} = \frac{1}{2\sigma_u^2}\|\mathbf{u} - \mathbf{u}^\star\|^2_{L^2(I)} + \frac{1}{2\sigma_g^2}|g_x - (g_x)_0|^2_{\mathbb{R}}$$

where $\mathbf{g} = (g_x, g_y)$ is the natural boundary condition at the inlet, with $g_x, g_y \in \mathbb{R}$. The starting domain Ω_0 is taken as a rectangle, the initial guess for $\mathbf{g}$ is $\mathbf{g}_0 = (-15, 0)$, and the ground truth is $\mathbf{g}^\star = (-25, 0)$. The combined reconstruction for $(\Omega, \mathbf{g})$ converges to the ground truth after 32 iterations (figures 4,5b).

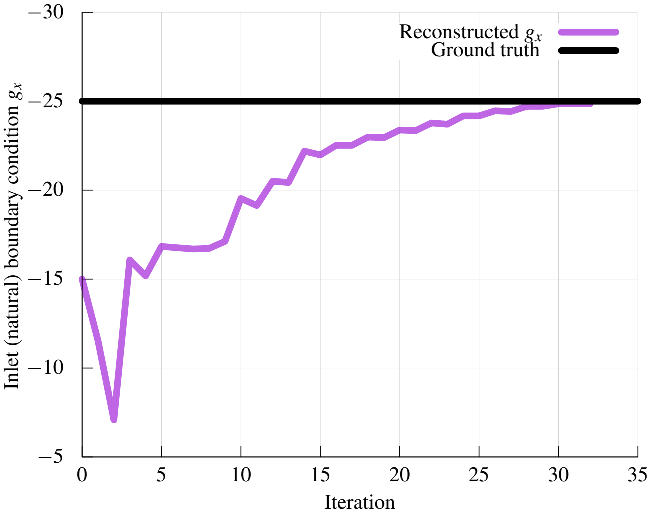
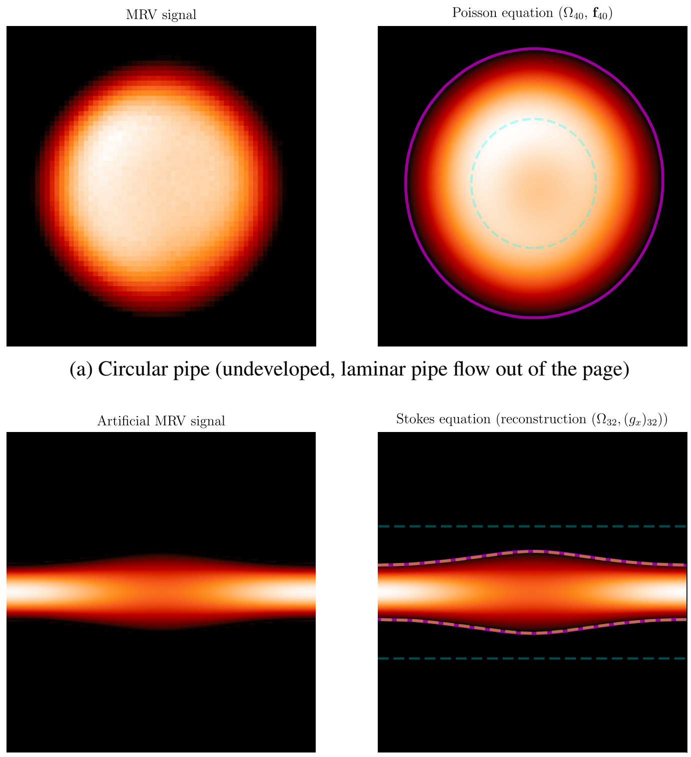


FIGURE 4: Iterates of the pressure inlet boundary condition g_x.

3 CONCLUSIONS

We have developed a prototype method to infer the shape of objects and the flows inside them from magnetic resonance velocimetry data. The method relies on (inverse) shape optimization techniques and is formulated in a natural framework for problems with moving domains. In particular, we implicitly define the boundaries in terms of signed distance functions and use Nitsche's method to weakly enforce the Dirichlet boundary condition on the moving front. The moving of the domain is expressed by a convection-diffusion equation for the signed distance function, which allows topological changes. We introduce additional unknown model parameters to the inverse problem, such as the forcing term $\mathbf{f}$ and the inlet boundary condition $\mathbf{g}$. We provide two alternative perspectives for the type of the regularization $\mathscr{R}$ one should choose: i) a purely variational perspective in function spaces, and ii) an approach based on Gaussian measures and random functions. However, the choice of $\mathscr{R}$ largely depends on the nature of the *a priori* knowledge. Finally, we test the methods for the Poisson and the Stokes problems using both real and synthetic MRV measurements. The present method



(a) Circular pipe (undeveloped, laminar pipe flow out of the page)

(b) Curved channel (Stokes flow, left to right)

FIGURE 5: Inverse problems for a) (Ω, f) using real MRV data [43], and b) $(\Omega, \mathbf{g})$ using synthetic MRV data (light intensity denotes velocity magnitude). The dashed cyan line denotes the initial guess Ω_0, the solid purple line the reconstructed domain, and the dashed yellow line the ground truth (where the lines overlap they seem orange).

shows several advantages over general image segmentation algorithms (which do not respect the underlying physics and the boundary conditions), and numerical experiments demonstrate its capability to reconstruct very noisy (SNR = 2.5) images. Consequently, our method provides a consistent treatment to reconstruct and filter velocity fields from MRV data, when the geometry of the domain, the boundary conditions, and the forcing term are not known beforehand. The extension of the above methods to the Navier–Stokes problem is under development.

REFERENCES

[1] Hoang, V. H., Law, K. J., and Stuart, A. M., 2014. "Determining white noise forcing from Eulerian observations in the Navier–Stokes equation". *Stochastics and Partial Differential Equations: Analysis and Computations,* ***2***(2), pp. 233–261.

[2] Gillissen, J. J. J., Bouffanais, R., and Yue, D. K. P., 2019. "Data assimilation method to de-noise and de-filter particle image velocimetry data". *Journal of Fluid Mechanics,* ***877***, pp. 196–213.

[3] Mons, V., Chassaing, J. C., and Sagaut, P., 2017. "Optimal sensor placement for variational data assimilation of unsteady flows past a rotationally oscillating cylinder". *Journal of Fluid Mechanics,* ***823***, pp. 230–277.

[4] Verma, S., Papadimitriou, C., Lüthen, N., Arampatzis, G., and Koumoutsakos, P., 2019. "Optimal sensor placement for artificial swimmers". *Journal of Fluid Mechanics,* ***884***.

[5] Stuart, A. M., 2010. "Inverse problems: A Bayesian perspective". *Acta Numerica,* ***19***(2010), pp. 451–459.

[6] Benning, M., and Burger, M., 2018. "Modern regularization methods for inverse problems". *Acta Numerica,* ***27***, pp. 1–111.

[7] Cotter, S. L., Dashti, M., Robinson, J. C., and Stuart, A. M., 2009. "Bayesian inverse problems for functions and applications to fluid mechanics". *Inverse Problems,* ***25***(11).

[8] Koltukluoğlu, T. S., and Blanco, P. J., 2018. "Boundary control in computational haemodynamics". *Journal of Fluid Mechanics,* ***847***, pp. 329–364.

[9] Koltukluoğlu, T. S., 2019. Fourier Spectral Dynamic Data Assimilation : Interlacing CFD with 4D Flow MRI. Tech. rep., Research Report No. 2019-56, ETH Zurich.

[10] Koltukluoğlu, T. S., Cvijetić, G., and Hiptmair, R., 2019. "Harmonic balance techniques in cardiovascular fluid mechanics". In Medical Image Computing and Computer Assisted Intervention – MICCAI 2019, Springer International Publishing, pp. 486–494.

[11] Funke, S. W., Nordaas, M., Evju, Ø., Alnæs, M. S., and Mardal, K. A., 2019. "Variational data assimilation for transient blood flow simulations: Cerebral aneurysms as an illustrative example". *International Journal for Numerical Methods in Biomedical Engineering,* ***35***(1), pp. 1–27.

[12] Morris, P. D., Narracott, A., Von Tengg-Kobligk, H., Soto, D. A. S., Hsiao, S., Lungu, A., Evans, P., Bressloff, N. W., Lawford, P. V., Rodney Hose, D., and Gunn, J. P., 2016. "Computational fluid dynamics modelling in cardiovascular medicine". *Heart,* ***102***(1), pp. 18–28.

[13] Sankaran, S., Kim, H. J., Choi, G., and Taylor, C. A., 2016. "Uncertainty quantification in coronary blood flow simulations: Impact of geometry, boundary conditions and blood viscosity". *Journal of Biomechanics,* ***49***(12), pp. 2540–2547.

[14] Katritsis, D., Kaiktsis, L., Chaniotis, A., Pantos, J., Efstathopoulos, E. P., and Marmarelis, V., 2007. "Wall Shear Stress: Theoretical Considerations and Methods of Measurement". *Progress in Cardiovascular Diseases,* ***49***(5), pp. 307–329.

[15] Sokolowski, J., and Zolesio, J.-P., 1992. *Introduction to Shape Optimization.* Springer Series in Computational Mathematics, Springer-Verlag, Berlin, Germany.

[16] Delfour, M., and Zolesio, J.-P., 2011. *Shapes and Geometries: Analysis, Differential Calculus, and Optimization.* 2nd ed., Advances in Design and Control 22, SIAM, Philadelphia, PA.

[17] Walker, S. W., 2015. *The Shapes Of Things: A Practical Guide To Differential Geometry And The Shape Derivative.* Advances in Design and Control, SIAM, Philadelphia, PA.

[18] Osher, S., and Sethian, J. A., 1988. "Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations". *Journal of Computational Physics,* ***79***(1), pp. 12–49.

[19] Burger, M., 2001. "A level set method for inverse problems". *Inverse Problems,* ***17***(5), pp. 1327–1355.

[20] Burger, M., 2003. "A framework for the construction of level set methods for shape optimization and reconstruction". *Interfaces and Free Boundaries,* ***5***(3), pp. 301–329.

[21] Burger, M., and Osher, S. J., 2005. "A Survey in Mathematics for Industry: A survey on level set methods for inverse problems and optimal design". *European Journal of Applied Mathematics,* ***16***(2), pp. 263–301.

[22] Yu, H., Juniper, M. P., and Magri, L., 2019. "Combined state and parameter estimation in level-set methods". *Journal of Computational Physics,* ***399***, p. 108950.

[23] Sethian, J. A., 1996. "A fast marching level set method for monotonically advancing fronts". *Proceedings of the National Academy of Sciences of the United States of America,* ***93***(4), pp. 1591–1595.

[24] Sethian, J. A., and Adalsteinsson, D., 1999. "The fast construction of extension velocities in level set methods". *Journal of Computational Physics,* ***148***, pp. 2–22.

[25] Crane, K., Weischedel, C., and Wardetzky, M., 2017. "The heat method for distance computation". *Communications of the ACM,* ***60***(11), pp. 90–99.

[26] Varadhan, S. R., 1967. "On the behavior of the fundamental solution of the heat equation with variable coefficients". *Communications on Pure and Applied Mathematics,* ***20***(2), pp. 431–455.

[27] Varadhan, S. R., 1967. "Diffusion processes in a small time interval". *Communications on Pure and Applied Mathematics,* ***20***(4), pp. 659–685.

[28] Nitsche, J., 1971. "Über ein Variationsprinzip zur Lösung von Dirichlet-Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind". *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg,* ***36***(1), pp. 9–15.

[29] Massing, A., Larson, M. G., Logg, A., and Rognes, M. E., 2014. "A Stabilized Nitsche Fictitious Domain Method for the Stokes Problem". *Journal of Scientific Computing,* ***61***(3), pp. 604–628.

[30] Hairer, M., 2009. An Introduction to Stochastic PDEs. Lecture Notes.

[31] Bogachev, V. I., 1998. *Gaussian measures*. American Mathematical Society.

[32] Lodhia, A., Sheffield, S., Sun, X., and Watson, S. S., 2016. "Fractional gaussian fields: A survey". *Probability Surveys,* ***13***(2016), pp. 1–56.

[33] Lischke, A., Pang, G., Gulian, M., Song, F., Glusa, C., Zheng, X., Mao, Z., Cai, W., Meerschaert, M. M., Ainsworth, M., and Karniadakis, G. E., 2020. "What is the fractional Laplacian? A comparative review with new results". *Journal of Computational Physics,* ***404***, p. 109009.

[34] Ramussen, C. E., and Williams, C. K. I., 2006. *Gaussian processes for machine learning.* The MIT Press.

[35] Tarantola, A., 2005. *Inverse Problem Theory and Methods for Model Parameter Estimation*. SIAM.

[36] Babuška, I., Banerjee, U., and Osborn, J. E., 2003. "Survey of meshless and generalized finite element methods: A unified approach". *Acta Numerica,* ***12***(2003), pp. 1–125.

[37] Burman, E., 2010. "La pénalisation fantôme". *Comptes Rendus Mathematique,* ***348***(21-22), pp. 1217–1220.

[38] Burman, E., and Hansbo, P., 2012. "Fictitious domain finite element methods using cut elements: II. A stabilized Nitsche method". *Applied Numerical Mathematics,* ***62***(4), pp. 328–341.

[39] Brenner, Susanne, Scott, R., 2008. *The Mathematical theory of finite element methods*, Vol. 15. Texts in Applied Mathematics, Springer.

[40] Mirtich, B., 1996. "Fast and Accurate Computation of Polyhedral Mass Properties". *Journal of Graphics Tools,* ***1***(2), pp. 31–50.

[41] Massing, A., Larson, M. G., and Logg, A., 2013. "Efficient implementation of finite element methods on nonmatching and overlapping meshes in three dimensions". *SIAM Journal on Scientific Computing,* ***35***(1).

[42] Virtanen, P., et al., 2020. "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python". *Nature Methods,* ***17***, pp. 261–272.

[43] Reci, A., 2019. "Signal sampling and processing in magnetic resonance applications". PhD thesis, University of Cambridge.