# Tracking of blood vessels motion from 4D-flow MRI data

Mocia Agbalessi, Alain Lalande, Olivier Bouchot, Toshiyuki Hayase, Jean-Joseph Christophe, Miguel Angel Fernández, Damiano Lombardi

## ▶ To cite this version:

Mocia Agbalessi, Alain Lalande, Olivier Bouchot, Toshiyuki Hayase, Jean-Joseph Christophe, et al.. Tracking of blood vessels motion from 4D-flow MRI data. Cardiovascular Engineering and Technology, 2023, 14, pp.577-604. 10.1007/s13239-023-00677-z . hal-03349442v3

**HAL Id: hal-03349442**

**https://inria.hal.science/hal-03349442v3**

Submitted on 5 Sep 2023

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution 4.0 International License

# Tracking of blood vessels motion from 4D-flow MRI data

Mocia Agbalessi[1,2], Alain Lalande[2,3], Olivier Bouchot[4], Toshiyuki Hayase[5], Jean-Joseph Christophe[6], Miguel Angel Fernández[1] and Damiano Lombardi[1*]

[1] Sorbonne Université, CNRS, Inria, Paris, France.
[2]ImVia Laboratory, University of Burgundy, Dijon, France.
[3]Department of Magnetic Resonance Imaging, Dijon University Hospital, Dijon, France.
[4]Department of Cardiovascular and Thoracic Surgery, Dijon University Hospital, Dijon, France.
[5]Institute of fluid science, Tohoku University, Sendai, Japan.
[6]CASIS - CArdiac Similation & Imaging Software, Dijon, France.

*Corresponding author(s). E-mail(s): damiano.lombardi@inria.fr;

**Abstract**

This paper presents a novel approach to track objects from 4D-flow MRI data. A salient feature of the proposed method is that it fully exploits the geometrical and dynamical nature of the information provided by this imaging modality. The underlying idea consists in formulating the tracking problem as a data assimilation problem, in which both position and velocity observations are extracted from the 4D-flow MRI data series. Optimal state estimation is then performed in a sequential fashion via Kalman filtering. The capabilities of the method are extensively assessed in a numerical study involving synthetic and clinical data.

**Keywords:** 4D-flow MRI, aortic wall tracking, data assimilation, Kalman filter.

# Contents

# Introduction

Among cardiovascular diseases, arterial aneurysms (i.e., abnormal enlargement of arteries), depict a high risk of mortality [1, 2]. To prevent complications, surgery is proposed in current clinical practice, consisting in the removal of the aneurysm and the replacement of the aortic section by a synthetic tube in Dacron. The surgical decision follows guidelines [3–5] which are solely based on the diameter of the aneurysm [3, 4]. Nevertheless, recent studies [6] showed that dissection and rupture can appear even when the aortic diameter is lower than current thresholds, in case of ascending aorta aneurysms and for patients without any specific medical history. Consequently, there is a strong clinical demand for more robust predictive parameters of aortic aneurysm risk. Biomarkers derived from mechanistic parameters such as wall stiffness, wall shear stress (WSS), inlet flow jet angle, etc. are being increasingly investigated to assess the risk of rupture of aneurysm. In *Hope et. al* [7], a correlation has been established between high growth rate of aortic aneurysm and abnormal flow displacement for patients with biscuspid aortic valve (BAV).

The most convenient way to obtain those parameters is based on Phase Contrast Magnetic Resonance Imaging (PC-MRI) [8], a medical imaging modality developed in recent decades with the aim of visualizing and quantifying the blood flow in the vasculature. In the literature, this imaging technique is also called 4D PC-MRI or 4D-flow MRI, because it includes time-resolved slice images covering a volume of acquisition, with flow encoding. It provides both anatomical (magnitude images) and functional (phase images) information that enables the non invasive analysis of blood vessel hemodynamics, and consequently the assessment of flow driven parameters of interest. *Hope et. al* [7] demonstrate a good reproducibility of biomarkers computed through 4D-flow MRI. However, 4D-flow analysis requires extensive pre-processing. In particular, the extraction of the aortic geometry at each time step of the cardiac cycle may be needed. The manual contouring of the aortic lumen takes 20-30 minutes for each time step[1] and cannot be considered as viable part in any clinical routine. It is thus necessary to develop automatic or at least semi-automatic segmentation algorithms of the aortic wall.

We distinguish segmentation from reconstruction, the first aims at enhancing in the images (for example with a binary mask or contours) pixels belonging to an organ of interest (aorta), whereas with the latter, the geometrical configurations of this organ are extracted. In the practice, 4D-flow MRI visualization and quantification require a 3D reconstruction of the geometry. This can be achieved by volume rendering followed by segmentation or by any vessel reconstruction technique using a contrast enhanced Magnetic Resonance angiography (MRA) sequence derived the 4D Flow MRI data[9]. Concerning the latter option, the purpose of using a MRA derived sequence is to benefit from a better image quality (contrast, spatial resolution) than 4D Flow MRI. Using this sequence, different techniques adapted to 2D image segmentation can be applied on the stack images of the MRA sequences, such as model based approaches, tracking techniques [10] or supervised/unsupervised image contour segmentations as referred in the review [11]. The common practice is thus to extract one 3D patient specific aortic reconstruction from 4D Flow derived image sequences, and all biomarkers are evaluated based on this configuration. However, more advanced parameters such as time-averaged WSS are often computed through numerical simulation, due to the limited spatial and temporal resolution of 4D MRI. WSS related biomarkers are increasingly investigated first to quantify the effect of abnormal flows near the aortic wall, and then to predict the location of dissection or rupture. Ideally, one would like to evaluate the biomarkers from patient specific fluid-structure interaction simulations of the aorta. This would require to reconstruct different aortic wall configurations from 4D-flow MRI. Furthermore, the extraction of Lagrangian information (i.e., point-to-point match between configurations) from 4D-flow MRI would also facilitate this calibration. Robust and accurate techniques have been proposed for the time coherent segmentation of 4D MRI image sequences (anatomic series) of the aorta with minimal human intervention [12]. The surface of the aorta can be reconstructed afterwards from 4D segmentation

---

[1] There is no standard time, but it depends on the expert.

resulting in a series of mesh representing different time configurations. Point-to-point correspondence is not generally required in these meshes. Notable results have been achieved in the reconstruction of time configurations with point-to-point matching, specifically for the left ventricle [13]. 2D contours lines are pre-segmented from time series of grayscale 3D images prior to applying the reconstruction. A patient specific template is reconstructed and deformed to fit the segmented contours, using a tree-based structure. The point-to-point matching is obtained by relying on geometrical assumptions. Note that the twist of the ventricle is not captured. Another alternative is to use 4D Tagged MRI data. The local deformation of the left ventricle endocardium is integrated in the reconstruction in terms of Lagrangian tracking [14]. Tagged MRI is however limited to organs with sufficient thickness, so that assessing the local deformation of the aortic wall from Tagged MRI might be delicate.

In this work, we address the tracking of the aortic wall motion from 4D-flow MRI sequences. The velocity data serves as a kinematical a priori for the estimation of the Lagrangian wall deformation. Several corrections are proposed in the literature to the near-wall velocity, due to the so-called partial volume effect [15, 16]. In the proposed method, we model this effect as an additive noise in the 4D-flow velocity field and the subsequent uncertainty is treated via Kalman filtering estimator. This provides both an estimate of the state of the system and of the quality of the estimation. A position observation operator is introduced to extract geometrical information from the grayscale images. It mimics active contours approaches [17, 18], which minimize the sum of an internal energy (the deformation energy which ensures surface regularity) and of an external energy (often build from the gradient of image intensity). Since the image gradient can be noisy or have some irregularities, additional features are added to smooth the image gradient as in [19], where the external energy is expressed with a generalized gradient vector flow.

In the literature, active contour approaches have been combined with Kalman filtering in order to achieve robust edge detection in 3D CTA segmentation, by tracking 2D contours from slice to slice [10]. Kalman filter has also been combined with active contours to perform 4D reconstruction of vessels from ultrasound [20], by tracking the centerline of the vessel at specific time steps. In the present paper, Kalman filtering is considered with a model-based prediction and correction from both time resolved anatomic images and flow series. Alternative algorithms that address the time reconstruction of moving parts of the heart from time-resolved image sequences can be found in [13, 14, 21]. They mainly rely on pre-processed or pre-segmented images, and do not exploit the flow information (Eulerian velocity field) provided by 4D MRI. The mesh point tracking, when considered, is performed through morphological hypotheses, image intensity or velocity approximations from complementary images with enhanced temporal resolution.

The rest of this paper is organized as follows. In the next section, the tracking problem is presented in both the continuous and the discrete settings. The position observation operator is presented in Section 1.3.2. After a brief presentation of Kalman filter, the proposed tracking algorithm is detailed in Section 2.2. Section 3 provides a numerical assessment of the proposed approach using synthetic and clinical data. Finally, a summary of the results and some conclusion with perspectives are drawn in Section 4.

# 1 Continuous problem and discretization

At the continuous level, we assume that the Lagrangian tracking of the aorta can be approximated by the tracking of the local deformation of an elastic surface $\Gamma$, which is transported by some regular velocity field $\boldsymbol{u}$. The deformation of $\Gamma$ is parametrized by means of a mapping $\boldsymbol{\phi}\colon \Gamma_{\mathrm{R}} \times [0,T] \longrightarrow \mathbb{R}^d$, for a given final time instant $T > 0$, $\Gamma_{\mathrm{R}}$ being a reference configuration. The transport equation is given by :

$$\begin{cases} \partial_t \boldsymbol{\phi}(\boldsymbol{\xi},t) = \boldsymbol{u}(\boldsymbol{\phi}(\boldsymbol{\xi},t),t), \quad (\boldsymbol{\xi},t) \in \Gamma_{\mathrm{R}} \times ]0,T], \\ \boldsymbol{\phi}(\boldsymbol{\xi},0) = \boldsymbol{\phi}_0(\boldsymbol{\xi}), \quad \boldsymbol{\xi} \in \Gamma_{\mathrm{R}}, \end{cases} \tag{1}$$

where $\boldsymbol{\phi}_0\colon \Gamma_{\mathrm{R}} \to \mathbb{R}^d$ stands for initial configuration of the surface (see Figure 1).

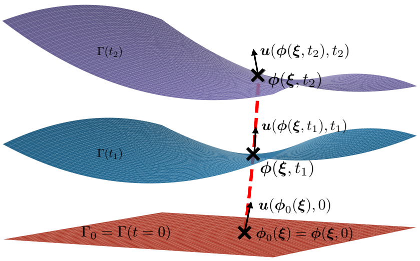


**Fig. 1** Illustration of the transport of $\Gamma(t)$ through the velocity field $\boldsymbol{u}$.

The problem considered in this paper consists in estimating the motion $\boldsymbol{\phi}$ from the following information (see Figure 2):

- the time-resolved grayscale image sequence described in terms of the scalar function $F_{\mathrm{M}}\colon \Omega \times [0,T] \to \mathbb{R}$;
- the measurement $\boldsymbol{u}_{\mathrm{M}}\colon \Omega \times [0,T] \to \mathbb{R}^d$ of the Eulerian velocity field $\boldsymbol{u}$ in $\Omega$. The measured field $\boldsymbol{u}_{\mathrm{M}}$ is presumably less regular than $\boldsymbol{u}$;
- a prior mapping $\widetilde{\boldsymbol{\phi}}\colon \Gamma_{\mathrm{R}} \times [0,T] \to \mathbb{R}^d$ such that the surface $\widetilde{\boldsymbol{\phi}}(\Gamma_R,\cdot)$ has a similar topology and close geometrical position as $\boldsymbol{\phi}(\Gamma_R,\cdot)$.

The first two points correspond to the data traditionally provided by 4D-flow MRI image sequences [22, 23].

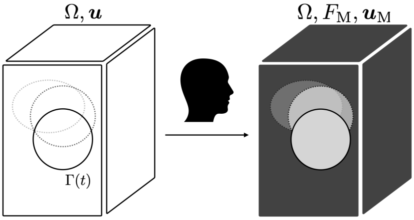


**Fig. 2** The surface $\Gamma(t)$ is transported by $\boldsymbol{u}$ within the domain $\Omega$. Due to contrast we can detect $\Gamma(t)$ on the image sequences $F_{\mathrm{M}}$ and $\boldsymbol{u}_{\mathrm{M}}$ is a measurement of the velocity field $\boldsymbol{u}$.

## 1.1 State estimation

We first introduce a variational formulation of the state estimation problem as follows: Find

- a mapping $\hat{\boldsymbol{\phi}}\colon \Gamma_{\mathrm{R}} \times [0,T] \to \mathbb{R}^d$;
- and a velocity field $\hat{\boldsymbol{v}}\colon \Gamma_{\mathrm{R}} \times [0,T] \to \mathbb{R}^d$,

which minimize the misfit functional

$$(\boldsymbol{\varphi}, \boldsymbol{v}) \mapsto \frac{1}{2}\int_0^T \int_{\Gamma_{\mathrm{R}}} \Big( \|\boldsymbol{\psi}_{\mathrm{M}} - \boldsymbol{\varphi}\|^2_{\mathsf{B}_{\mathrm{pos}}} + \|\boldsymbol{u}_{\mathrm{M}}(\boldsymbol{\psi}_{\mathrm{M}}) - \boldsymbol{v}\|^2_{\mathsf{B}_{\mathrm{vel}}} \Big)\, \mathrm{d}\boldsymbol{\xi}\, \mathrm{d}t, \tag{2}$$

where $\boldsymbol{\psi}_{\mathrm{M}}\colon \Gamma_{\mathrm{R}} \times [0,T] \to \mathbb{R}^d$ is an observation of the unknown mapping $\boldsymbol{\phi}$. For the moment let assume that $\boldsymbol{\psi}_{\mathrm{M}}$ is given and depends on both the time-dependent images $F^n_{\mathrm{M}}$ and the prior $\tilde{\boldsymbol{\phi}}$. The procedure to compute this term will be described in the next section. The symmetric positive definite matrices $\mathsf{B}_{\mathrm{pos}}$, $\mathsf{B}_{\mathrm{vel}}$ have two purposes. First, they ensure dimensionless comparison between the misfit to the position observation and the misfit to the velocity observation. Second, $\mathsf{B}_{\mathrm{pos}}$ and $\mathsf{B}_{\mathrm{vel}}$ quantify the confidence in the observed mapping $\boldsymbol{\psi}_{\mathrm{M}}$ and in the observed velocity $\boldsymbol{u}_{\mathrm{M}}(\boldsymbol{\psi}_{\mathrm{M}})$, respectively.

An additional constraint is added to the transport equation (1) to impose some regularity on the solution $\hat{\boldsymbol{v}}$. Here we chose the following system of equations:

$$\begin{cases} \partial_t \boldsymbol{\varphi}(\boldsymbol{\xi}, t) = \boldsymbol{v}(\boldsymbol{\xi}, t), \quad (\boldsymbol{\xi}, t) \in \Gamma_{\mathrm{R}} \times ]0,T], \\ \partial_t \boldsymbol{v}(\boldsymbol{\xi}, t) = \kappa \Delta_{\boldsymbol{\xi}} \big( \boldsymbol{\varphi}(\boldsymbol{\xi}, t) - \widetilde{\boldsymbol{\phi}}_0(\boldsymbol{\xi}) \big), \quad (\boldsymbol{\xi}, t) \in \Gamma_{\mathrm{R}} \times ]0,T], \\ \boldsymbol{\varphi}(\boldsymbol{\xi}, 0) = \widetilde{\boldsymbol{\phi}}_0(\boldsymbol{\xi}), \quad \boldsymbol{v}(\boldsymbol{\xi}, 0) = \widetilde{\boldsymbol{v}}_0(\boldsymbol{\xi}), \quad \boldsymbol{\xi} \in \Gamma_{\mathrm{R}}, \end{cases} \tag{3}$$

where $\widetilde{\boldsymbol{\phi}}_0\colon \Gamma_{\mathrm{R}} \to \mathbb{R}^d$ and $\widetilde{\boldsymbol{v}}_0\colon \Gamma_{\mathrm{R}} \to \mathbb{R}^d$ denote the initial conditions and play the role of a control on the $\boldsymbol{\varphi}$ and $\boldsymbol{v}$, in order to reach the optimum in (2) and $\Delta_\xi$ is the Laplace-Beltrami operator on the reference surface. Consequently, the misfit functional (2) could be written as a function the initial condition

$(\widetilde{\boldsymbol{\phi}}_0, \widetilde{\boldsymbol{v}}_0)$ through (3). Hence, the optimization problem reads as:

$$\min_{\widetilde{\boldsymbol{\phi}}_0, \widetilde{\boldsymbol{v}}_0} \frac{1}{2} \int_0^T \int_{\Gamma_{\mathrm{R}}} \left( \|\boldsymbol{\psi}_{\mathrm{M}} - \boldsymbol{\varphi}\|^2_{\mathsf{B}_{\mathrm{pos}}} + \|\boldsymbol{u}_{\mathrm{M}}(\boldsymbol{\psi}_{\mathrm{M}}) - \boldsymbol{v}\|^2_{\mathsf{B}_{\mathrm{vel}}} \right) \, \mathrm{d}\boldsymbol{\xi} \, \mathrm{d}t, \tag{4}$$

such that $\boldsymbol{\varphi}$ and $\boldsymbol{v}$ depend on $\widetilde{\boldsymbol{\phi}}_0$ and on $\widetilde{\boldsymbol{v}}_0$ as follows

$$\begin{cases} \partial_t \boldsymbol{\varphi} = \boldsymbol{v} \quad \text{in} \quad \Gamma_{\mathrm{R}} \times ]0, T], \\ \partial_t \boldsymbol{v} = \mathcal{F}(\boldsymbol{\varphi}) \quad \text{in} \quad \Gamma_{\mathrm{R}} \times ]0, T], \\ \boldsymbol{\varphi}(\cdot, 0) = \widetilde{\boldsymbol{\phi}}_0(\cdot) \quad \boldsymbol{v}(\cdot, 0) = \widetilde{\boldsymbol{v}}_0(\cdot) \quad \text{in} \quad \Gamma_{\mathrm{R}}, \end{cases} \tag{5}$$

and with $\mathcal{F}(\boldsymbol{\varphi}) := \kappa \Delta_{\boldsymbol{\xi}} \big(\boldsymbol{\varphi} - \widetilde{\boldsymbol{\phi}}_0\big)$. The state estimation is formulated as a space and time continuous optimization problem controlled by the initial condition.

## 1.2 The position observation operator from grayscale images

In this section, we describe the observation operator of the unknown mapping $\boldsymbol{\phi}$ from the images $F_{\mathrm{M}}$. It corresponds to the mapping $\boldsymbol{\psi}_{\mathrm{M}}$ introduced in the minimization problems (2) and (4). Grayscale intensity images of the domain $\Omega$ provide geometrical information on the location of the surface $\Gamma(t)$. In fact, moving objects on images are usually visible and trackable to the naked eyes, at least in cases in which there is sufficient intensity contrast, and provided that there is reasonable prior information on the position and the shape. As shown on Figure 2, the mapping describing the motion of the surface is unknown, but we can see the surface moving due to the contrast of the image. The ability to track the surface motion is a consequence of the fact that the contour, which identifies $\Gamma(t)$, is associated with a *local* extremal value of the norm of the gradient of the image intensity. This mere property is widely used in basic edge detection techniques. Different image filters have been developed in order to quantify the gradient of the image intensity [24, 25] and thus facilitate contour segmentation. Among them, the Canny edge detection [26] proposes contour identification, since it selects candidate contours given ribbons of extremal gradient. This approach can be impacted by the image contrast and local inhomogeneities which might result in artefacts in the segmentation and non-smooth geometry reconstruction. This shows that maximizing the gradient of the image intensity does not guarantee the regularity of the boundary, even if the boundary of the real object is smooth. For this reason, a regularizing constraint is also added in the surface position operator that will be presented.

In what follows, the norm $\|\cdot\|$ is induced by the canonical euclidian scalar product $\langle \cdot, \cdot \rangle$. If the space of positive-definite symmetric matrices of dimension $d \times d$ is referred as $\mathcal{S}_d(\mathbb{R})$, then for each element $\mathrm{B} \in \mathcal{S}_d(\mathbb{R})$ we can define the scalar product $\langle \boldsymbol{x}, \boldsymbol{x} \rangle_{\mathrm{B}} = \boldsymbol{x}^{\mathrm{T}} \mathrm{B}^{-1} \boldsymbol{y}, \forall \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d$, and the induced norm $\|\cdot\|_{\mathrm{B}}$. At a given $t \in (0, T)$, the proposed position observation operator involves the minimization of the following functional $\mathcal{J}_t$, defined for sufficiently regular deformation mapping $\boldsymbol{\psi}\colon \Gamma_\mathrm{R} \to \mathbb{R}^d$:

$$\mathcal{J}_t(\boldsymbol{\psi}) = \frac{1}{2}\int_{\Gamma_\mathrm{R}} \Big( - \left\|\boldsymbol{\nabla}_{\boldsymbol{x}} F_\mathrm{M}\big(\boldsymbol{\psi}(\boldsymbol{\xi}), t\big)\right\|^2_{\mathsf{B}_\mathrm{img}} + \left\|\boldsymbol{\psi}(\boldsymbol{\xi}) - \widetilde{\boldsymbol{\phi}}(\boldsymbol{\xi}, t)\right\|^2_{\mathsf{B}_\mathrm{prior}} \\ + \delta \left\| \boldsymbol{\nabla}_{\boldsymbol{\xi}} \big(\boldsymbol{\psi} - \widetilde{\boldsymbol{\phi}}_0\big)(\boldsymbol{\xi})\right\|^2 \Big) \,\mathrm{d}\boldsymbol{\xi}. \tag{6}$$

Here, the symbols $\boldsymbol{\nabla}_{\boldsymbol{x}}$, $\boldsymbol{\nabla}_{\boldsymbol{\xi}}$ denote the gradient with respect to the cartesian and the curvilinear coordinates, respectively. The mapping $\widetilde{\boldsymbol{\phi}}$ represents a prior on the unknown mapping $\boldsymbol{\phi}$, so that the second term can be interpreted as a trust region. The matrices $\mathsf{B}_\mathrm{img}$ and $\mathsf{B}_\mathrm{prior}$, both in $\mathcal{S}_d(\mathbb{R})$, weigh the confidence in the image gradient term, and the confidence in the geometrical prior, respectively. In practice, we have

$$\mathsf{B}_\mathrm{img} := \frac{\gamma}{\|\boldsymbol{\nabla}_{\boldsymbol{x}} F_\mathrm{M}(\cdot, t)\|^2_\infty} \mathsf{I}_d,$$

with $\mathsf{I}_d$ the identity matrix in dimension $d$, $\gamma > 0$ a dimensionless user-defined parameter. $\mathsf{B}_\mathrm{prior}$ is a diagonal matrix whose coefficients are given by the spacings of the image in its three spatial directions. The larger $\gamma$, the more confidence is given to the image gradient maximization. The third term of (6) is a regularization term penalizing the norm of the derivatives of the mapping $\boldsymbol{\psi} - \widetilde{\boldsymbol{\phi}}_0$. The position observation is thus defined as $\boldsymbol{\psi}_\mathrm{M}\colon \Gamma_\mathrm{R} \times [0, T] \to \mathbb{R}^d$ such that

$$\boldsymbol{\psi}_\mathrm{M}(\cdot, t) := \arg\inf_{\boldsymbol{\psi}} \mathcal{J}_t(\boldsymbol{\psi}(\cdot)) \tag{7}$$

for all $t \in ]0, T]$. Minimizing $\mathcal{J}_t$ is equivalent to selecting a smooth surface among all possible surfaces resulting from the local maximization of the image gradient around $\widetilde{\boldsymbol{\phi}}$, see Figure 3.

The Euler-Lagrange equations yield the following necessary condition satisfied by $\boldsymbol{\psi}_\mathrm{M}$:

$$-\boldsymbol{\nabla}_{\boldsymbol{xx}} F_\mathrm{M}(\boldsymbol{\psi}_\mathrm{M}) \mathsf{B}^{-1}_\mathrm{img} \boldsymbol{\nabla}_{\boldsymbol{x}} F_\mathrm{M}(\boldsymbol{\psi}_\mathrm{M}) + \mathsf{B}^{-1}_\mathrm{prior}(\boldsymbol{\psi}_\mathrm{M} - \widetilde{\boldsymbol{\phi}}) - \delta \Delta_{\boldsymbol{\xi}}\big(\boldsymbol{\psi}_\mathrm{M} - \widetilde{\boldsymbol{\phi}}_0\big) = \mathbf{0}, \tag{8}$$

where the symbol $\nabla_{\boldsymbol{xx}} F_\mathrm{M}$ denotes the Hessian matrix of $F_\mathrm{M}$.

*Remark 1* As stated, the position observation operator behaves like an active contour method. The functional $\mathcal{J}_t$ can be decomposed into an image-based energy, given by the two first terms, and an internal energy with the norm of the surface gradient. ☐

Obtaining the observed mapping $\boldsymbol{\psi}_\mathrm{M}$ directly from (8) would require to treat the non-linearity induced by the image gradient and Hessian term. In what follows, we propose a splitting procedure which mitigates the difficulties

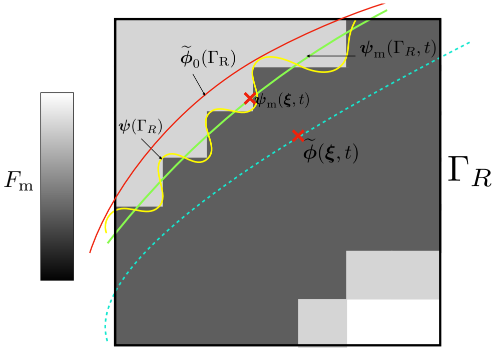


**Fig. 3** The position observer. On the image at $t \in [0, T]$, the maximization of the norm of the image gradient would result in the mapping $\boldsymbol{\psi}$ with high local curvature. Among all those maps, the regularizing constraint selects $\boldsymbol{\psi}_{\mathrm{M}}$. Having the trust region around $\widetilde{\boldsymbol{\phi}}$ enforces a selection of a specific extremal area.

encountered either in a direct minimization of $\mathcal{J}_t$ or in the resolution of the Euler-Lagrange equations. Note that the two first terms of $\mathcal{J}_t$ in (6) treat independently points on $\boldsymbol{\psi}(\Gamma_{\mathrm{R}})$ while the third couples nearby points on the surface. The proposed two-step approach reads as follows:

1. Minimization of a simplified cost functional $\widetilde{\mathcal{J}}_t$ to generate an intermediate observation $\widetilde{\boldsymbol{\psi}}_{\mathrm{M}}$:
$$\widetilde{\boldsymbol{\psi}}_{\mathrm{M}}(\cdot, t) := \arg\inf_{\boldsymbol{\psi}} \widetilde{\mathcal{J}}_t(\boldsymbol{\psi}(\cdot)), \tag{9}$$
with
$$\widetilde{\mathcal{J}}_t \colon \boldsymbol{\psi} \to \frac{1}{2} \int_{\Gamma_{\mathrm{R}}} \left( -\left\| \boldsymbol{\nabla}_{\boldsymbol{x}} F_{\mathrm{M}}(\boldsymbol{\psi}(\boldsymbol{\xi}), t) \right\|^2_{\mathsf{B}_{\mathrm{img}}} + \left\| \boldsymbol{\psi}(\boldsymbol{\xi}) - \widetilde{\boldsymbol{\phi}}(\boldsymbol{\xi}, t) \right\|^2_{\mathsf{B}_{\mathrm{prior}}} \right) \, \mathrm{d}\boldsymbol{\xi}. \tag{10}$$
2. Surface regularization that provides the end-of-step position observation $\boldsymbol{\psi}_{\mathrm{M}}$:
$$\boldsymbol{\psi}_{\mathrm{M}} - \widetilde{\boldsymbol{\psi}}_{\mathrm{M}} - \delta \Delta_{\boldsymbol{\xi}} \big( \boldsymbol{\psi}_{\mathrm{M}} - \widetilde{\boldsymbol{\phi}}_0 \big) = \mathbf{0}. \tag{11}$$

Some remarks are in order to motivate the above splitting approach as an alternative to (7).

We have $\mathcal{J}_t = \widetilde{\mathcal{J}}_t + \mathcal{G}$, with

$$\mathcal{G} \colon \boldsymbol{\psi} \to \frac{1}{2} \int_{\Gamma_{\mathrm{R}}} \delta \left\| \boldsymbol{\nabla}_{\boldsymbol{\xi}} \big( \boldsymbol{\psi} - \widetilde{\boldsymbol{\phi}}_0 \big)(\boldsymbol{\xi}) \right\|^2 \, \mathrm{d}\boldsymbol{\xi}.$$

The intermediate position observation $\widetilde{\boldsymbol{\psi}}_{\mathrm{M}}$ returned by (9) does not satisfy *a priori* the optimality condition for $\mathcal{J}_t$, that is, $\partial_{\boldsymbol{\psi}} \mathcal{J}_t(\widetilde{\boldsymbol{\psi}}_{\mathrm{M}}) \neq \mathbf{0}$. With a slight abuse of notation we have denoted here with $\partial_\psi$ the Fréchet derivative of $\mathcal{J}_t$ with respect to the function $\psi$. Consequently, a descent gradient method would suggest to move in the opposite direction of the gradient of $\mathcal{J}_t$, namely, $-\delta(\partial_{\boldsymbol{\psi}}\widetilde{\mathcal{J}}_t(\widetilde{\boldsymbol{\psi}}_{\mathrm{M}})+\partial_{\boldsymbol{\psi}}\mathcal{G}(\widetilde{\boldsymbol{\psi}}_{\mathrm{M}})) = -\partial_{\boldsymbol{\psi}}\mathcal{G}(\widetilde{\boldsymbol{\psi}}_{\mathrm{M}})$. The second step (11) performs such a gradient descent step from $\widetilde{\boldsymbol{\psi}}_{\mathrm{M}}$, but with an implicit treatment of $\boldsymbol{\psi}_{\mathrm{M}}$ in the Fréchet derivate term $\partial_{\boldsymbol{\psi}}\mathcal{G}$.

## 1.3 Fully discrete formulation

In this section, we present the fully discrete counterpart of the continuous estimation problem introduced in the previous section, after temporal and spatial discretization.

### 1.3.1 Spatial and temporal discretization

The surface $\Gamma(t)$ is discretized in space by means of a triangular mesh with $N_{\mathrm{P}}$ points. The mesh points are uniquely identified by their indices $i \in \{0, \cdots, N_{\mathrm{P}}-1\}$ and the motion of the surface $\Gamma(t)$ is assessed through the time-dependent positions $\left\{\boldsymbol{q}_i\colon [0,T] \to \mathbb{R}^d\right\}_{i=0,\cdots,N_{\mathrm{P}}-1}$ of of its points. Let introduce the position $\mathbf{Q}(t) = \left(\boldsymbol{q}_0(t)^T \ \cdots \ \boldsymbol{q}_{N_{\mathrm{P}}-1}(t)^T\right) \in \mathbb{R}^{dN_{\mathrm{P}}}$ and velocity $\mathbf{U}(t) = \left(\boldsymbol{u}_0(t)^T \ \cdots \ \boldsymbol{u}_{N_{\mathrm{P}}-1}(t)^T\right) \in \mathbb{R}^{dN_{\mathrm{P}}}$ vectors. After discretization in space, the dynamics (5) yield the following linear ODE system:

$$\begin{cases} \dot{\mathbf{Q}} = \mathbf{U} \quad \text{in} \quad [0,T], \\ \dot{\mathbf{U}} = \kappa \mathsf{K}\left(\mathbf{Q} - \widetilde{\mathbf{Q}}^0\right) \quad \text{in} \quad [0,T], \\ \mathbf{Q}(0) = \widetilde{\mathbf{Q}}^0, \\ \mathbf{U}(0) = \widetilde{\mathbf{U}}^0. \end{cases} \tag{12}$$

where $\widetilde{\mathbf{Q}}^0, \widetilde{\mathbf{U}}^0 \in \mathbb{R}^{dN_{\mathrm{P}}}$ stands for the (unknown) initial location and velocity of the surface points and $\mathsf{K}$ denotes the stiffness matrix of the surface Laplacian (which can be derived from a finite element approximation, or using spring-mass models [27]). By introducing the full state of the system $\mathbf{X} := (\mathbf{Q}^T \, \mathbf{U}^T) \in \mathbb{R}^{2dN_{\mathrm{P}}}$, system (12) can be recast into the general form

$$\dot{\mathbf{X}}(t) = \mathsf{A}\mathbf{X}(t) + \mathbf{R} \quad \text{in} \quad [0,T], \tag{13}$$

with the notations

$$\mathsf{A} := \begin{pmatrix} 0 & I_{dN_p} \\ -\kappa\mathsf{K} & 0 \end{pmatrix}, \quad \widetilde{\mathbf{X}}^0 := \begin{pmatrix} \widetilde{\mathbf{Q}}^0 \\ \widetilde{\mathbf{U}}^0 \end{pmatrix}, \quad \mathbf{R} := \begin{pmatrix} \mathbf{0} \\ \kappa\mathsf{K}\widetilde{\mathbf{Q}}^0 \end{pmatrix}.$$

In the present work, we consider one-step time discretizations of (13) by means of implicit methods (Backward Euler or Crank-Nicolson), which yield the following time-marching scheme:

$$\begin{cases} \mathbf{X}^n = \mathsf{D}_{n|n-1}\mathbf{X}^{n-1} + \mathbf{R}^n, \quad 1 \leq n \leq N_T, \\ \mathbf{X}^0 = \widetilde{\mathbf{X}}^0, \end{cases} \tag{14}$$

where, for instance, in the case of the Backward Euler scheme we have

$$\mathsf{D}_{n|n-1} = (\mathsf{I} - \Delta T \mathsf{A})^{-1}, \quad \mathbf{R}^n = (\mathsf{I} - \Delta T \mathsf{A})^{-1}\mathbf{R}.$$

Here, $\Delta T := T/N_T$ denotes the time-step length, with $N_T + 1$ the number of data samples available.

### 1.3.2 Discrete observation operator

The continuous position operator (9) and (11) is also discretized, so as to provide the observation of the mesh points location. For $n = 0, \cdots N_T$, the 4D MRI data is given in terms of the image sequences $F^n_{\mathrm{M}}$ and the Eulerian measured velocity $\boldsymbol{u}^n_{\mathrm{M}}$ as piecewise constant (scalar and vector, respectively) fields on a voxel grid of size $N_1 \times \cdots \times N_d$. In what follows, the voxels are denoted by $\Omega_{\boldsymbol{\alpha}}$ and are uniquely identified by the multi-index $\boldsymbol{\alpha} = (\alpha_1, \cdots, \alpha_d) \in [1, N_1] \times \cdots \times [1, N_d]$, as in the blue annotations of Figure 4 ($d = 2$). The gradient of $F^n_{\mathrm{M}}$, noted $\nabla_{\mathbf{x}} F^n_{\mathrm{M}}$, is approximated voxel-wise by means of a centered finite differences scheme. Both $F^n_{\mathrm{M}}$ and $\nabla_{\mathbf{x}} F^n_{\mathrm{M}}$ are taken constant per voxel. The voxel values of $\boldsymbol{u}^n_{\mathrm{M}}$ are assigned to voxel centers, they are defined on a generic point of $\Omega$ via trilinear interpolation.

The position observations are evaluated by means of the two-step approach, as in Section 1.2:

1. The discrete image-gradient based cost functional counterpart of (10), noted $J_n(\widetilde{\boldsymbol{q}}^n_i;\ \cdot)$ is minimized for voxel centers within a local window search for size $2\Delta N + 1$ for each mesh point $i$ to generate an intermediate observed position $\boldsymbol{q}^{n-}_{i,\mathrm{M}}$. Note that the mesh points can be treated independently. The voxel window search for the minimization is introduced through the compact notation

$$B(\boldsymbol{k}, \Delta N) = \{\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_d),\ \text{s.t. } |k_j - \alpha_j| \leq \Delta N,\ \forall j\}, \tag{15}$$

with $\Delta N < \min_{k_d} \{(N_{k_d} - 1)/2\}$, and represents the set of voxel indices within the box centered in the voxel $\Omega_{\boldsymbol{k}}$, of size $(2\Delta N + 1)$ in each direction. Let denote $\boldsymbol{\pi} : \Omega \to [1, N_1] \times \cdots \times [1, N_d]$ the function which associates to each point $\boldsymbol{x} \in \Omega$, the multi-index $\boldsymbol{k'}$ of the voxel strictly containing $\boldsymbol{x}$. The minimization is performed by evaluating the cost functional on the centers $\boldsymbol{c}_{\boldsymbol{\alpha}}$ of the voxels $\Omega_{\boldsymbol{\alpha}}$. Hence, the intermediate observed position is $\boldsymbol{q}^{n-}_{i,\mathrm{M}}(t) = \boldsymbol{c}_{\boldsymbol{k}^*}$ with

$$\boldsymbol{k}^* := \underset{\boldsymbol{k} \in B(\pi(\widetilde{\boldsymbol{q}}^n_i), \Delta N)}{\arg\min} \quad J_n(\widetilde{\boldsymbol{q}}^n_i;\ \boldsymbol{c}_{\boldsymbol{k}}), \tag{16}$$

and

$$\begin{aligned} J_n(\widetilde{\boldsymbol{q}}^n_i;\ \cdot) \colon \Omega &\longrightarrow \mathbb{R} \\ \boldsymbol{x} &\longmapsto -\gamma \frac{\|\boldsymbol{\nabla}_{\boldsymbol{x}} F^n_{\mathrm{M}}(\boldsymbol{x})\|^2}{\|\boldsymbol{\nabla}_{\boldsymbol{x}} F^n_{\mathrm{M}}\|^2_\infty} + \|\boldsymbol{x} - \widetilde{\boldsymbol{q}}^n_i\|^2_{\mathsf{B}_{\mathrm{prior}}}. \end{aligned} \tag{17}$$

$\mathsf{B}_{\text{prior}} \in \mathcal{S}_d(\mathbb{R})$ is the same as in the space continuous problem, see Section 1.2.

2. The end-of-step observation $\boldsymbol{q}^n_{i,\mathrm{M}}$ is computed by applying the discrete regularization:
$$\mathbf{Q}^n_{\mathrm{M}} - \mathbf{Q}^{n-}_{\mathrm{M}} = -\delta \mathsf{K}_{\text{obs}} \cdot (\mathbf{Q}^n_{\mathrm{M}} - \widetilde{\mathbf{Q}}^0), \tag{18}$$
with $\mathbf{Q}^n_{\mathrm{M}} = \big((\boldsymbol{q}^n_{0,\mathrm{M}})^T \cdots (\boldsymbol{q}^n_{N_{\mathrm{P}}-1,\mathrm{M}})^T\big) \in \mathbb{R}^{dN_{\mathrm{P}}}$, $\mathbf{Q}^{n-}_{\mathrm{M}} = \big((\boldsymbol{q}^{n-}_{0,\mathrm{M}})^T \cdots (\boldsymbol{q}^{n-}_{N_{\mathrm{P}}-1,\mathrm{M}})^T\big) \in \mathbb{R}^{dN_{\mathrm{P}}}$ and $\mathsf{K}_{\text{obs}} := \kappa_{\text{obs}}\mathsf{K} \in \mathbb{R}^{dN_{\mathrm{P}} \times dN_{\mathrm{P}}}$. The diffusion step (18) is performed with the same discretization as introduced in Section 1.3.1.

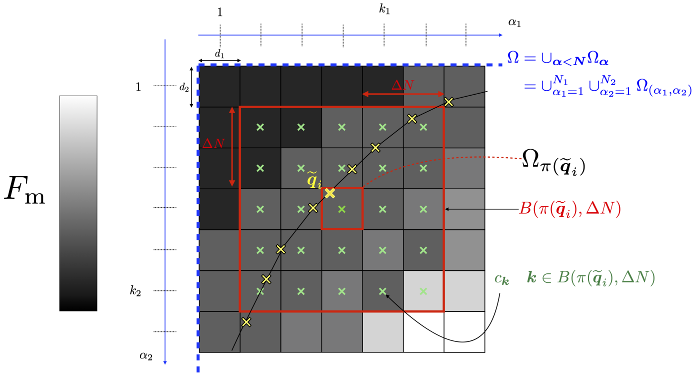


**Fig. 4** Two-dimensional illustration of optimization step parameters in the position observation operator ($\Delta N = 2$). The landmark (mesh point) $i$ is in $\widetilde{\boldsymbol{q}}_i(t)$ in the prior mesh (yellow). It is inside the voxel $\Omega_{\pi(\widetilde{\boldsymbol{q}}_i)}$ (in red) which defines the local window search $B(\pi(\widetilde{\boldsymbol{q}}_i(t)), \Delta N)$ for the optimum. The functional $J_t(\widetilde{\boldsymbol{q}}_i;\ \cdot)$ will be evaluated in the voxel centers $c_k$ for all $k \in B(\pi(\widetilde{\boldsymbol{q}}_i(t)), \Delta N)$.

The complete state observation $\mathbf{Z}^n \in \mathbb{R}^{2dN_{\mathrm{P}}}$ is thus given by
$$\mathbf{Z}^n = \begin{pmatrix} \mathbf{Q}^n_{\mathrm{M}} \\ \mathbf{U}^n_{\mathrm{M}} \end{pmatrix},$$
with $\mathbf{U}^{\mathrm{M}} = \big(\boldsymbol{u}^n_{\mathrm{M}}(\boldsymbol{q}^n_{0,\mathrm{M}})^T \cdots \boldsymbol{u}^n_{\mathrm{M}}(\boldsymbol{q}^n_{N_{\mathrm{P}}-1,\mathrm{M}})^T\big) \in \mathbb{R}^{dN_{\mathrm{P}}}$.

### 1.3.3 Discrete estimation problem

After spatial and temporal discretization, the discrete counterpart of (4)-(5) takes the form
$$\min_{\mathbf{X}_0 = (\mathbf{Q}^0, \mathbf{U}^0)^T} \frac{1}{2} \sum_{i=1}^{N_T} \|\mathbf{Z}^n - \mathbf{X}^n\|^2_{\mathsf{B}}, \tag{19}$$
such that
$$\begin{cases} \mathbf{X}^n = \mathsf{D}_{n|n-1}\mathbf{X}^{n-1} + \mathbf{R}^n, \\ \mathbf{X}^0 = \mathbf{X}_0, \end{cases} \tag{20}$$

with $\mathbf{X}_0$ the control representing the unknown initial condition and we denote

$$\mathbf{Z}^n = \mathcal{F}_{\mathrm{obs}}\left(F_{\mathrm{M}}^n, \boldsymbol{u}_{\mathrm{M}}^n, \widetilde{\mathbf{Q}}^n, \gamma, \Delta N, \delta, \kappa_{\mathrm{obs}}\right). \tag{21}$$

the discrete observation operator. This associates to the data an estimation of position and velocity by solving the optimisation problem introduced in Section 1.3.2.

In (19), the symbol $\mathsf{B} \in \mathbb{R}^{2dN_{\mathrm{P}} \times 2dN_{\mathrm{P}}}$ denotes he diagonal block matrix, whose $N_{\mathrm{P}}$ first and $N_{\mathrm{P}}$ other blocks of size $d \times d$ are identically equal to $\mathsf{B}_{\mathrm{pos}}$ and $\mathsf{B}_{\mathrm{vel}}$, respectively:

$$\mathsf{B} = \begin{pmatrix} \mathsf{B}_{\mathrm{pos}} & & & & & \\ & \ddots & & & & \\ & & \mathsf{B}_{\mathrm{pos}} & & & \\ & & & \mathsf{B}_{\mathrm{vel}} & & \\ & & & & \ddots & \\ & & & & & \mathsf{B}_{\mathrm{vel}} \end{pmatrix}.$$

Problem (19)-(20) can be solved using either variational or sequential approaches (see, e.g., [28]). On the one hand, variational methods (and more specifically 4D-VAR approaches) search for the optimal control (initial condition) such that the resulting estimation best fits all the observations. To this purpose, one may solve the associated Euler-Lagrange equations, or apply an optimization algorithm to find the saddle point of the associated Lagrangian. On the another hand, sequential estimation techniques such the Kalman filter improve the current state estimation as observations are provided. In case where the model and the observations are both linear, the estimation returned at a certain time by a (linear) Kalman filter is the best linear estimation given the observations provided up to that time. All observations are thus integrated in the estimation only at the final time for Kalman filter. Linear Kalman filter and 4D-VAR approaches return the same state estimation, only at final time [29].

# 2 State estimation via Kalman filter

In this section, we briefly review the main features of the Kalman filter [30, 31] then we build on it to formulate the sequential estimation counterpart of (19)-(20).

## 2.1 Kalman filter

When the model describing the evolution of the state is linear, as Eq. (20), and the observations $\mathbf{Z}^n$ are linear in the state, the Kalman filter provides an optimal estimation of the state.

**Algorithm 1** (Simplified) Linear Kalman Filter

```
Require: $\left(\hat{\mathbf{X}}^0;\ \mathsf{P}_0\right)$, $(\mathsf{A}_{n|n-1}, \mathbf{R}_{n=0,\cdots,N_T};\ \mathsf{W}_{\text{model}})$, $(\mathbf{Z}^n;\ \mathsf{W}_{\text{obs}})$
  for $n = 1, \cdots, N_T$ do
    Prediction: $(\hat{\mathbf{X}}^{n-1}, \mathsf{P}_{n-1}) \to (\hat{\mathbf{X}}^{n^-}, \mathsf{P}_{n^-})$
    $\hat{\mathbf{X}}^{n^-} \leftarrow \mathsf{D}_{n|n-1}\hat{\mathbf{X}}^{n-1} + \mathbf{R}^n$
    $\mathsf{P}_{n^-} \leftarrow \mathsf{D}_{n|n-1}\mathsf{P}_{n-1}\mathsf{D}^{\mathrm{T}}_{n|n-1} + \mathsf{W}_{\text{model}}$
    Correction: $(\mathbf{Z}^n, \mathsf{W}_{\text{obs}}, \hat{\mathbf{X}}^{n^-}, \mathsf{P}_{n^-}) \to \left(\hat{\mathbf{X}}^n, \mathsf{P}_n\right)$
    $\mathsf{S}_n \leftarrow \mathsf{P}_{n^-} + \mathbb{W}_{\text{obs}}$
    $\mathsf{K}_n^{\mathrm{T}}$ is solution of $\mathsf{S}_n^{\mathrm{T}}\chi = \mathsf{P}_{n^-}^{\mathrm{T}}$
    $\hat{\mathbf{X}}^n \leftarrow \hat{\mathbf{X}}^{n^-} + \mathsf{K}_n(\mathbf{Z}^n - \hat{\mathbf{X}}^{n^-})$
    $\mathsf{P}_n \leftarrow (\mathsf{I} - \mathsf{K}_n)\mathsf{P}_{n^-}$
  end for
  return $\left(\hat{\mathbf{X}}^n, \mathsf{P}_n\right), n = 1, \cdots, N_T$
```

The Kalman filter presented in Algorithm 1, and whose main steps are described below, will be directly applied to our specific case.

### *Initialization*

To initialise the method, we need to provide an initial prior state $\hat{\mathbf{X}}^0 = ((\hat{\mathbf{Q}}^0)^T\ \ (\hat{\mathbf{U}}^0)^T)^T \in \mathbb{R}^{2dN_{\mathrm{P}}}$ and its associated covariance matrix $\mathsf{P}_0 \in \mathbb{R}^{2dN_p \times 2dN_p}$. Its interpretation is the following: the prior can be written as $\mathbf{X}^0 + \eta^0$, where $\mathbf{X}^0 = \mathbf{X}(0)$ is the real (unknown) initial state and $\eta^0$ is a random variable such that $\mathbb{E}[\eta^0] = \mathbf{0}$ and of $\mathbb{E}[(\eta^0)^T(\eta^0)] = \mathsf{P}_0$. In the present context, $\hat{\mathbf{Q}}^0$ defines the prior on the initial position the mesh points and $\hat{\mathbf{U}}^0$ their associated initial velocity.

### *Prediction*

At time $t_n$, given the best state estimation at previous time as well as its associated covariance, $\hat{\mathbf{X}}^{n-1} = \left((\hat{\mathbf{Q}}^{n-1})^T\ \ (\hat{\mathbf{U}}^{n-1})^T\right)^T$, we compute a prediction, denoted by $\hat{\mathbf{X}}^{n-}$ and the so called *a priori* covariance matrix $\mathsf{P}_n^-$ by exploiting the model:

$$\begin{cases} \hat{\mathbf{X}}^{n-} &= \mathsf{D}_{n|n-1}\hat{\mathbf{X}}^{n-1} + \mathbf{R}^n, \\ \mathsf{P}_{n^-} &= \mathsf{D}_{n|n-1}\mathsf{P}_{n-1}\mathsf{D}^{\mathrm{T}}_{n|n-1} + \mathsf{W}_{\text{model}}, \end{cases}$$

where the symbol $\mathsf{W}_{\text{model}} \in \mathbb{R}^{2dN_p \times 2dN_p}$ stands for the model error covariance matrix.

### *Correction*

In its classical formulation, we write the estimation of the state at time instant $t_n$, denoted by $\hat{\mathbf{X}}^n$, as a correction of the prediction provided by the model, $\hat{\mathbf{X}}^{n-}$. The correction term depends upon the discrepancy, namely $(\mathbf{Z}^n - \hat{\mathbf{X}}^{n-})$.

The correction step takes the form:

$$\hat{\mathbf{X}}^n = \hat{\mathbf{X}}^{n-} + \mathsf{K}_n(\mathbf{Z}^n - \hat{\mathbf{X}}^{n-}), \tag{22}$$

where $\mathsf{K}_n$ is called the Kalman filter gain. Note that the general formulation of the linear Kalman filter considers the observation $\mathbf{Z} = \mathsf{H}\mathbf{X} + \eta$ as a linear function of the state $\mathbf{X}$, with $\mathsf{H}$ usually a rectangular matrix and $\eta$ a white noise. Here, $\mathsf{H}$ is the identity. The covariance matrix $\mathbb{W}_{\text{obs}} \in \mathbb{R}^{2dN_\text{P} \times 2dN_\text{P}}$ quantifies the uncertainty in the observation. Its expression is derived from the precision of the position observer, and the noise in the measured velocity. The most expensive part of the correction step is the calculation of $\mathsf{K}_n$ in Algorithm 1. In a vast majority of applications, $\mathsf{K}_n$ is of smaller size since the observation is of lower dimension than the state. Here, calculating $\mathsf{K}_n$ might require a significant amount of memory and computational time, depending on the size of the state vector.

## 2.2 Proposed estimation algorithm

***Implicit-Explicit model prediction.***

In the present work we discretised in time the model equations Eq.(13) by means of implicit schemes (Backward Euler or Crank-Nicolson). In the case of the Backward Euler scheme this would lead to a prediction step of the form:

$$\begin{aligned} \mathsf{D}_{n|n-1} &= (\mathsf{I} - \Delta T\mathsf{A})^{-1}, \\ \mathbf{R}^n &= (\mathsf{I} - \Delta T\mathsf{A})^{-1}\mathbf{R}, \\ \hat{\mathbf{X}}^{n-} &= \mathsf{D}_{n|n-1}\hat{\mathbf{X}}^{n-1} + \mathbf{R}^n. \end{aligned} \tag{23}$$

From a practical point of view, computing the update of the state just involves the resolution of a sparse linear system, which can be performed efficiently. The covariance update, however, will be computationally very expensive, as it would involve the resolution of $4d \cdot N_p$ linear systems. This is the reason why, in the present work, we propose to update the covariance by means of an explicit scheme. We will refer to this approximation as implicit-explicit prediction.

In particular, let $\mathsf{D}^{(exp)}_{n|n-1} = \mathsf{I} + \Delta T\mathsf{A}$, the update of the covariance matrix reads:

$$\mathsf{P}_{n^-} = \mathsf{D}^{(exp)}_{n|n-1}\mathsf{P}_{n-1}[\mathsf{D}^{(exp)}_{n|n-1}]^\mathrm{T} + \mathsf{W}_{\text{model}} \tag{24}$$

The covariance matrix of the model error $\mathsf{W}_{\text{model}}$ is calculated from the assumption that the model error results from a white noise in acceleration whose covariance matrix is $\sigma_a^2 I_{dN_p}$. Consequently we propose an approximation[2] of model error covariance matrix, given by:

$$\mathsf{W}_{\text{model}} = \sigma_a^2 \begin{pmatrix} \frac{\Delta T^4}{4}\mathsf{I}_{dN_\text{P}} & \frac{\Delta T^3}{2}\mathsf{I}_{dN_\text{P}} \\ \frac{\Delta T^3}{2}\mathsf{I}_{dN_\text{P}} & \Delta T^2\mathsf{I}_{dN_\text{P}} \end{pmatrix}. \tag{25}$$

---

[2] This approximation results from the propagation of a noise in acceleration for a zero-acceleration dynamical equation, that is to say $\ddot{\boldsymbol{q}} = a$, with $a$ a white noise.

*Remark 2* We propose a reasonable order of magnitude for $\sigma_a$ from the expression of $\mathsf{W}_{\text{model}}(\sigma_a)$. In fact, the noise in acceleration induces an error on the position $\Delta T^2 \sigma_a / 2$ which could be of the order of some voxels spacing $N\sqrt{h_1^2 + \cdots + h_d^2}$. In such a case, it would have a significant effect on the reconstruction compared to the observation error introduced by (16). This yields to the approximation

$$\sigma_a \sim 2N\sqrt{h_1^2 + \cdots + h_d^2}/\Delta T^2. \tag{26}$$

This heuristics provides a first evaluation of the order of magnitude, which has proven to work for the synthetic test case and the application to real data in Section 3.

### *Generating the prior* $\widetilde{\mathbf{Q}}^n$

As explained previously, the observation at iteration $n$ depends on $\widetilde{\mathbf{Q}}^n$ which is yet to be determined. Since $\widetilde{\mathbf{Q}}^n$ is a prior value for the state at iteration $n$, a reasonable guess could come from the model prediction. Given $\hat{\mathbf{X}}^{n-1}$ the prior for the local maximization of the image gradient is set to $\mathsf{B}_{n|n-1}^{-1}\left(\hat{\mathbf{X}}^{n-1} + \mathbf{R}^n\right)$ or equivalently $\mathsf{B}_{n|n-1}\widetilde{\mathbf{Q}}^n = \hat{\mathbf{X}}^{n-1} + \mathbf{R}^n$. Consequently, the observation is given by

$$\mathbf{Z}^n = \mathcal{F}_{\text{obs}}\left(F_{\text{M}}^n, \boldsymbol{u}_{\text{M}}^n, ''\mathsf{B}_{n|n-1}^{-1}\left(\hat{\mathbf{X}}^{n-1} + \mathbf{R}^n\right)'', \gamma, \Delta N, \delta, \kappa_{\text{obs}}\right),$$

taking up the notation introduced in Section 1.3.1. Note that $\mathsf{B}_{n|n-1}$ is not actually inverted, we solve the associated linear system instead.

**Algorithm 2** Full algorithm

```
Require: $\left(\hat{\mathbf{X}}^0;\ \mathsf{P}_0\right)$, $\left(\mathsf{A}_{n|n-1}, \mathbf{R}_{n=0,\cdots,N_T};\ \mathsf{W}_{\text{model}}\right)$, $\mathsf{W}_{\text{obs}}$
  for $n = 1, \cdots, N_T$ do
      Generate observation $\mathbf{Z}^n = \mathcal{F}_{\text{obs}}\left(F_{\text{M}}^n, \mathsf{B}_{n|n-1}^{-1}\left(\hat{\mathbf{X}}^{n-1} + \mathbf{R}^n\right), \gamma, \Delta N, \delta, \kappa_{\text{obs}}\right)$
      Apply Prediction: $(\hat{\mathbf{X}}^{n-1}, \mathsf{P}_{n-1}) \rightarrow (\hat{\mathbf{X}}^{n^-}, \mathsf{P}_{n^-})$ from Algorithm 1
      Apply Correction: $(\mathbf{Z}^n, \mathsf{W}_{\text{obs}}, \hat{\mathbf{X}}^{n^-}, \mathsf{P}_{n^-}) \rightarrow \left(\hat{\mathbf{X}}^n, \mathsf{P}_n\right)$ from Algorithm 1
  end for
  return $\left(\hat{\mathbf{X}}^n, \mathsf{P}_n\right), n = 1, \cdots, N_T$
```

The Kalman filter requires the user to specify the following parameters:

- For the model: $\sigma_a > 0$, directly associated to the norm of $\mathsf{W}_{\text{model}}$ from (25). Through (26), an order of magnitude of the $\sigma_a$ is provided.
- For the observation: $\sigma_q > 0$ and $\sigma_p > 0$, parameters for the covariance matrix $\mathsf{W}_{\text{obs}} \in \mathbb{R}^{2dN_p \times 2dN_p}$:

$$\mathsf{W}_{\text{obs}} = \mathsf{W}_{\text{obs}}(\sigma_q, \sigma_p) = \begin{pmatrix} (\sigma_q \mathsf{D}_{N_{\text{P}}})^2 & 0 \\ 0 & \sigma_p^2 \mathsf{I}_{dN_p} \end{pmatrix}.$$

$\mathsf{D}_{N_\mathrm{P}} = \mathrm{diag}(\mathsf{D}, \cdots, \mathsf{D}) \in \mathbb{R}^{dN_\mathrm{P} \times dN_\mathrm{P}}$ denotes the block diagonal matrix defined with the spacing matrix of the image data, $\mathsf{D} = \mathrm{diag}(\delta_0, \cdots, \delta_{d-1}) \in \mathbb{R}^{d \times d}$. For $i = 0, \cdots, d-1$, $\delta_i$ is the image spacing along the $i$-th axis.

- For the initial configuration: $\sigma_{P_0,q} > 0$ and $\sigma_{P_0,p} > 0$, parameters of the covariance matrix $\mathsf{P}_0 \in \mathbb{R}^{2dN_p \times 2dN_p}$:

$$\mathsf{P}_0 = \mathsf{P}_0(\sigma_{P_0,q}, \sigma_{P_0,p}) = \begin{pmatrix} (\sigma_{P_0,q}\mathsf{D}_{N_\mathrm{P}})^2 & 0 \\ 0 & \sigma^2_{P_0,p}\mathsf{I}_{dN_p} \end{pmatrix}.$$

Unlike $\sigma_p$ and $\sigma_{P_0,p}$, the user-defined parameters $\sigma_q$ and $\sigma_{P_0,q}$ are dimensionless and their values represent pixel radius of trust regions. In addition to the Kalman filter parameters, specific parameters must be specified:

- For the model (prediction): the elasticity parameter $\kappa \geq 0$.
- For the observation: $\gamma > 0$, $\Delta N \in \mathbb{N}^*$ the size of the local search window. Those parameters will be fixed. Intuitively the values prescribed should depend on the quality of the image. For the elasticity equation: the parameters $\kappa_\mathrm{obs} \geq 0$, or rather the product $\kappa_\mathrm{obs}\delta \geq 0$.

One can refer to Figure 5 for an overview of the parameters.

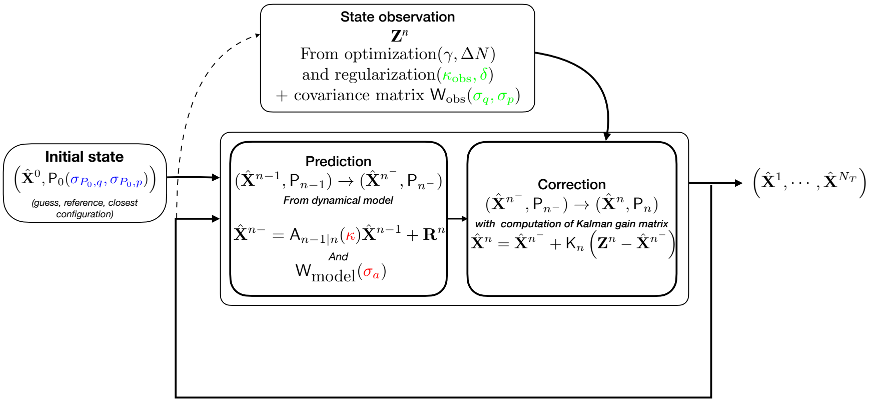


**Fig. 5** The parameters of the Kalman filter and the state observation.

Some of the parameters depends on the data quality.

## 2.3 Computational complexity.

The computational complexity of the proposed algorithm is discussed in this section. The different steps of the Kalman filter estimation are commented separately. Let $\mathcal{N} = 2dN_P$ and the state be a vector $X \in \mathbb{R}^{\mathcal{N}}$.

### Prediction.

In the prediction phase, the state is predicted by solving a linear system. The use of a direct solver would lead to a complexity of $\mathcal{O}(\mathcal{N}^3)$, whereas an iterative solver would make it possible to scale as $\mathcal{O}(\mathcal{N}^2)$. In order to perform the covariance update we have proposed an explicit update. This amounts to perform a matrix-matrix product. The system matrix $A$ is sparse. Let $c \in \mathbb{N}$ be the maximum number of non-zero elements in a row or column of $A$. Then, the cost scales as $\mathcal{O}(c\mathcal{N}^2)$.

### Observation.

The observation of the position requires the resolution of the optimisation problem defined by Eq.(6). This has been split into two sub-steps. In the first one, a non-linear optimisation problem is solved locally around every point of the mesh, by performing an extensive search inside a window. The cost is therefore linear in the number of points, *i.e.* $\mathcal{O}(N_P) \sim \mathcal{O}(\mathcal{N})$. The second step, which aims at regularising the shape, takes the form of a diffusion step, carried out implicitly. This implies the resolution of a linear system, which requires $\mathcal{O}(\mathcal{N}^3)$ or $\mathcal{O}(\mathcal{N}^2)$ depending on wether we use a direct or an iterative solver respectively. The velocity observation operation takes the form of a multi-linear interpolation. Its cost scales as $\mathcal{O}(\mathcal{N})$.

### Update.

The Kalman filter update requires matrix-matrix operations to update the covariance. This is in general the most costly step of the procedure, and it takes $\mathcal{O}(\mathcal{N}^4)$ or $\mathcal{O}(\mathcal{N}^3)$ operations depending on wether we compute the matrix $S$ in Algorithm 1 by using a direct or an iterative solver.

The cost is essentially related to the mesh size, and it is so also when considering real data coming from 4d-flow MRI. In a realistic setting, given the image resolution (a voxel of typically 2mm of size), it is reasonable to consider a mesh of around $10^3$ points. The proposed method is feasible when considering this number of points.

## 3 Numerical experiments

The purpose of this section is to illustrate the capabilities of Algorithm 2 by means of numerical experiments. The proposed test cases also aim to assess the sensitivity of the method to free parameters (see Tables 1 and 2) and noisy data. We consider two test cases. The first one (Section 3.1) is based on synthetic data and the second (Section 3.2) uses real data from 4D-flow MRI exam of a human aorta.

The precision of the surface segmentation is evaluated through the Dice coefficient, a metric used in image segmentation to compare the level of similarity of two segmentations. For $\mathcal{X}$ the ground truth segmentation, and $\mathcal{X}'$ another segmentation, both represented by collections of voxels, the Dice coefficient is given by the relation

$$\mathrm{Dice}(\mathcal{X}, \mathcal{X}') := \frac{\mathrm{card}(\mathcal{X} \cap \mathcal{X}')}{\mathrm{card}(\mathcal{X}) + \mathrm{card}(\mathcal{X}')},$$

where card stands for the cardinality of a set. In the synthetic test case, another complementary approach is proposed to account for the quality of the point-to-point tracking. It is essential to note that a good surface reconstruction (Dice coefficient close to 1) does not necessarily mean a good point tracking. If we consider the example of a circle rotating on itself, mesh points with a very fine resolution have a rotating motion while the surface would be apparently motionless on images. Standard surface reconstruction method are not able to detect the motion of the circle on itself, unless additional kinematic information is processed. The proposed tracking algorithms allows to estimate both the position and velocity of the surface. Algorithm 2 has been implemented using the standard C++ libraries, and the Eigen library for linear algebra operations [32]. The VTK library has also been intensively used for handling images and geometrical objects [33].

## 3.1 Tracking of a vesicle immersed in a lid-driven cavity flow

In this example, we consider the tracking of the motion of an elastic vesicle immersed in a lid-driven cavity flow (see Figure 6). The bi-dimensional motion of the vesicle is more complex than the deformation of the aorta under the pulsatile blood flow. Synthetic image data are generated from numerical simulations of the corresponding dynamical system. In particular, the simulated Eulerian velocity field is extracted so as to mimic 4D MRI flow series. Algorithm 2 is then used to retrieve the motion of the vesicle from the generated synthetic data. We investigate the sensitivity of the reconstruction considering the data quality (image, velocity measurements) and the regularization parameters.

### 3.1.1 Data generation

The considered dynamical system can be modelled by a non-linear fluid-structure interaction model (see, e.g., [34] for the details). The fluid is initially at rest and occupies the domain $\Omega := [0,1] \times [0,1]$. The vesicle is also initially at rest and has the shape of a circle centered at $(0.6, 0.5)$ with radius 0.2. All the units are given in the CGS unit system. The fluid is described by the incompressible Navier-Stokes equations, while a non-linear Timoshenko-Ehrenfest beam model in Lagrangian formalism is considered for the vesicle. The physical parameters for the fluid are: $\rho^{\mathrm{f}} = 100$, $\nu = 10$, and for the solid equation$\rho^{\mathrm{s}} = 100$, the Young Modulus $E = 5.6 \times 10^3$ and the Poisson's ratio $\nu = 0.4$. The boundary conditions on $\partial\Omega := \Gamma^f_{\mathrm{wall}} \cup \Gamma^f_{\mathrm{in}} \cup \Gamma^f_{\mathrm{out}} \cup \Gamma^f_{\mathrm{top}}$ (see Figure

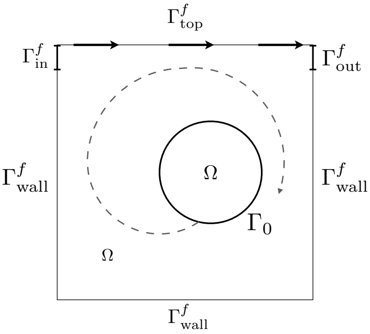
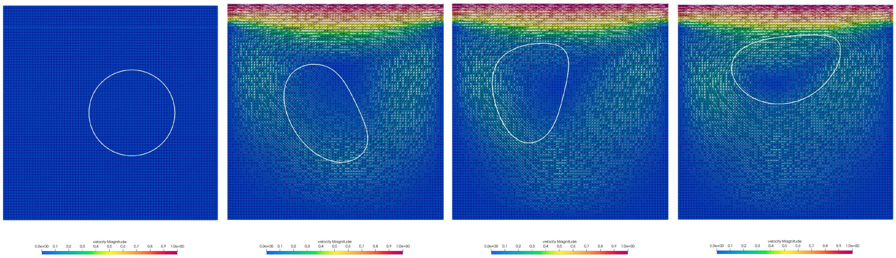


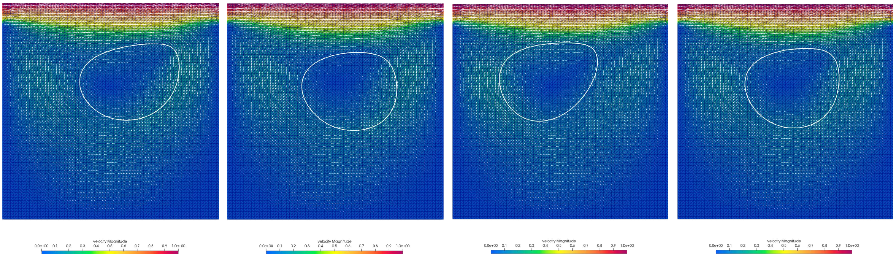
**Fig. 6** Geometrical configuration.

6) are the following: no-slip boundary conditions on $\Gamma^f_{\text{top}} \cup \Gamma^f_{\text{wall}}$, zero traction on $\Gamma^f_{\text{in}} \cup \Gamma^f_{\text{out}}$, constant velocity equal to 1 on $\Gamma^f_{\text{top}}$. Synthetic data has been generated by simulating 4000 time steps, of length $\Delta T_0 = 5.0 \times 10^{-3}$, using the numerical method proposed in [34] with a fluid mesh made of 12800 triangular elements and 19360 edges, and a solid mesh made of 160 edges.

Some snapshots of the fluid velocity magnitude and solid deformation are given in Figure 7. The vesicle deforms and rotates within the fluid cavity.



(a) $t = 0.0$. (b) $t = 2.0s$. (c) $t = 4.0s$. (d) $t = 7.0s$.



(e) $t = 9.0$. (f) $t = 10s$. (g) $t = 14s$. (h) $t = 20s$.

**Fig. 7** Data generation - Motion of the immersed vesicle, from FSI simulations.

From the solid displacement, the grayscale image frames $\{F_{\mathrm{M}}^{n}\}_{n=0,\cdots,N_T-1}$ of the vesicle in motion are generated, with a sampling time length $\Delta T = 20 \times \Delta T_0$, which represents the temporal resolution of the images. It results in $N_T = 200$ image frames. The pixel grid dimension is $80 \times 80$ with an isotropic spacing $h_1 = h_2 = 0.0125$. In fact , two adjacent triangles in the mesh of the cavity combine into one square, defining a pixel, as shown in Figure 8.This sets the image resolution. In order to generate the grayscale image at a time-step $n \in \{0, \cdots, N_T - 1\}$ ,we first assign one value among $\{0, 200\}$ to each pixel depending on its location with respect to the vesicle $\Gamma^n$. Pixels located inside the vesicle are assigned the maximum value 200, and those strictly outside have a value 0. The resulting binary image $F_{\mathrm{M},0}^{n}$ at the time step $n$ thus writes

$$\begin{aligned} F_{\mathrm{M},0}^{n}\colon \left\{\Omega_{\boldsymbol{\alpha}=(\alpha_1,\alpha_2)} \text{ s.t. } (\alpha_1,\alpha_2) \in [1,80]\times[1,80]\right\} &\longrightarrow \{0,200\} \\ (\Omega_{\boldsymbol{\alpha}}, t) &\longmapsto \begin{cases} 200, & \text{if } c_{\boldsymbol{\alpha}} \text{ is inside } \Gamma^n \\ 0 & \text{otherwise,} \end{cases} \end{aligned} \tag{27}$$

where $\{\Omega_{\boldsymbol{\alpha}}, \boldsymbol{\alpha} \in [1,80]\times[1,80]\}$ is the voxel grid covering $\Omega$, $c_{\boldsymbol{\alpha}}$ the center of a voxel $\Omega_{\boldsymbol{\alpha}}$.

$F_{\mathrm{M},0}^{n}$ is then blurred to generate a grayscale image $F_{\mathrm{M},1}^{n}$ using a Gaussian filter implemented in the VTK library `vtkImageGaussianSmooth`[3]. The filter consists in a convolution with a 2D Gaussian kernel[4] as follows:

$$\begin{aligned} K_{\mathrm{2D}}(i,j) :=& \frac{1}{2\pi\sigma_1\sigma_2}\exp\left[-\left(\frac{i^2}{2\sigma_1^2}+\frac{j^2}{2\sigma_2^2}\right)\right] \\ =& \underbrace{\frac{1}{\sqrt{2\pi}\sigma_1}\exp\left(-\frac{i^2}{2\sigma_2^2}\right)}_{:=K_{\mathrm{1D}}^{(1)}(i)}\underbrace{\frac{1}{\sqrt{2\pi}\sigma_2}\exp\left(-\frac{j^2}{2\sigma_2^2}\right)}_{K_{\mathrm{1D}}^{(2)}(j)}, \end{aligned}$$

for $i \in \{-R_1, -R_1+1, \cdots, R_1\}$ and $j \in \{-R_2, -R_2+1, \cdots, R_2\}$, $R_1, R_2 \in \mathbb{N}^*$. For a voxel labelled $\boldsymbol{\alpha} = (\alpha_1, \alpha_2)$, the evaluation of $F_{\mathrm{M},1}^{n}$ is given by :

$$\begin{aligned} F_{\mathrm{M},1}^{n}(\alpha_1,\alpha_2) &= [F_{\mathrm{M},0}^{n} * K_{\mathrm{2D}}](\alpha_1,\alpha_2) \\ &= \sum_{i=-R_1}^{R_1}\sum_{j=-R_2}^{R_2} F_{\mathrm{M},0}^{n}(\alpha_1 - i, \alpha_2 - j) K_{\mathrm{2D}}(i,j) \\ &= \sum_{i=-R_1}^{R_1} [F_{\mathrm{M},0}^{n} * K_{\mathrm{1D}}^{(2)}](\alpha_1 - i, \alpha_2) K_{\mathrm{1D}}^{(1)}(i) \\ &= \sum_{i=-R_2}^{R_2} [F_{\mathrm{M},0}^{n} * K_{\mathrm{1D}}^{(1)}](\alpha_1, \alpha_2 - j) K_{\mathrm{1D}}^{(2)}(j), \end{aligned} \tag{28}$$

---

[3] Detailed implementation on https://github.com/Kitware/VTK/blob/master/Imaging/General/vtkImageGau

[4] see `vtkImageGaussianSmooth`

where $*$ denotes the convolution product. Note the abuse of notation for $F^n_{\mathrm{M},0}, F^n_{\mathrm{M},1}$, since the voxel label is assimilated to the voxel itself. VTK adds a normalization to the Gaussian kernel. The kernel size parameters $R_k$, $k = 1, 2$ depend on the parameters $\sigma_k$ with the relation $R_k = 2\lfloor \sigma_k R' \rfloor + 1$, $R'$ being a radius factor. The Gaussian blur is performed with $\sigma_1 = \sigma_2 = \sigma$ and $R' = 1$. The standard deviation $\sigma$ is expressed in pixel units. The main purpose of the Gaussian kernel is to reduce the gradient gap across the contours, hence artificially degrading the performance of the image gradient maximization. The larger $\sigma$, the more challenging an accurate contour detection might be. Gradient artefacts are finally added to $F^n_{\mathrm{M},1}$, as it may occur with 4DMRI, using a Gaussian noise with null-expected value and a standard deviation of 5% of 200, providing the image $F^n_{\mathrm{M}}$.

The Eulerian velocity field $\boldsymbol{u}_{\mathrm{M}}$ is composed of the exact velocity inside the cavity $\Omega$, with an additive Gaussian noise $\boldsymbol{\eta}$ with $\mathbf{0}$-mean and $\sigma_v^2 \mathsf{I}_2$ covariance matrix. The evaluation of $\boldsymbol{u}_{\mathrm{M}}$ at any point inside the cavity is approximated by linear interpolation before adding the Gaussian noise.

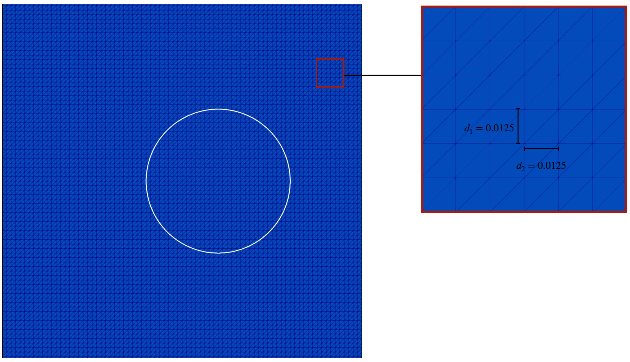


**Fig. 8** Data generation - The mesh of the cavity, with the initial geometrical configuration of the real vesicle.

### *Initial configuration.*

The initial configuration for the tracking is given by a mesh composed of $N_{\mathrm{P}} = 16$ equally distributed points $\left\{\hat{\boldsymbol{q}}_i^0\right\}_{i=0,\cdots,N_{\mathrm{P}}}$ along an ellipse centered at $(0.65, 0.42)$ of semi-major axis 0.22 and semi-minor axis 0.2, see Figure 9. The center of the ellipse has been translated by a vector $(0.05, -0.08)$ with respect of the center of the real initial configuration, such that the Dice coefficient at the initialization is 0.72. The time resolution is $\Delta T = 0.1$, $N_T = 200$. A finer time integration is used for the model (20), namely 0.001, as mentioned in Section 2.2.

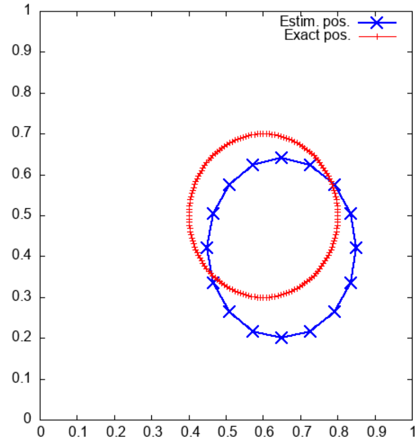


**Fig. 9** Initial configuration. In red, the initial position of the real vesicle. In blue, the initial mesh given to the filter for reconstruction.

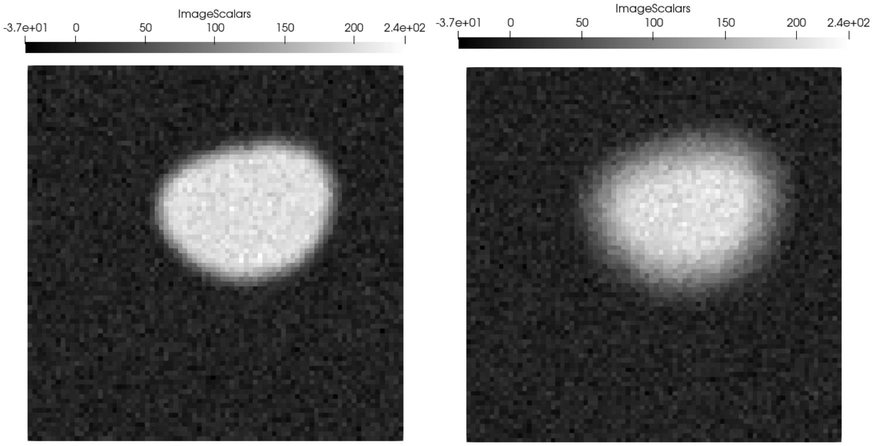
### 3.1.2 Sensitivity to data quality

The sensitivity of the filter with respect of the level of blur $\sigma$ in the images is investigated for different values of the velocity noise $\sigma_v$. The blur parameter $\sigma$ controls the diffusion of the gradient extremal values, so that large values of $\sigma$ makes the observation of the surface position more challenging. The data quality thus depends on $\sigma$ and $\sigma_v$. The parameters of the simulations are given in Table 1.

| Data quality | | Intialization | | Model | | Observation | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $\sigma$ | $\sigma_v$ | $\sigma_{P_0,q}$ | $\sigma_{P_0,p}$ | $\kappa$ | $\sigma_a$ | $\gamma$ | $\Delta N$ | $\kappa_{\text{obs}}\delta$ | $\sigma_q$ | $\sigma_p$ |
| $\{2.0, 6.0\}$ | {0.003, 0.03, 0.075, 0.105} | $10h_x$ | 0.1 | 10 | 5.0 | 100 | 30 | 0.1 | {1.5, 2.5, 3.0} | $\sigma_v$ |

**Table 1** Parameters for the first synthetic test case.

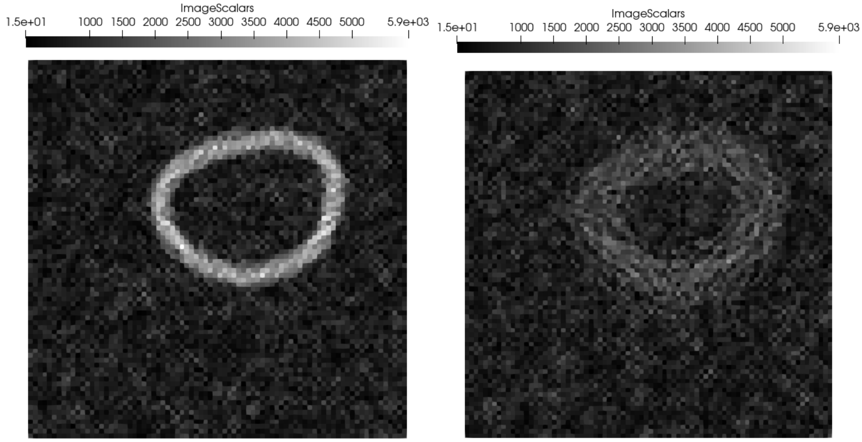
The simulations are performed with $\sigma = 2.0, 6.0$ (see Figures 10 and 11) and $\sigma_v = 0.003, 0.03, 0.075, 0.105$. The values of $\sigma_v$ simulate respectively 1%, 10%, 25% and 35% of the half range of variation of the highest component of $\boldsymbol{u}_{\text{M}}^n$ in an area around the vesicle, for all time steps. Here, that half range corresponds to 0.30. Those percentages deem as an appreciation of the estimated level of noise in the flow series, see Figures 12. Depending on the level of blur, the precision $\sigma_q$ of the position observation operator is adapted accordingly. For $\sigma = 2.0, 6.0, 8.0$, we respectively set $\sigma_q = 1.5, 2.5, 3.0$. Since every simulation is unique for a set of parameters $(\sigma, \sigma_q, \sigma_v)$, the Dice coefficient is averaged over 100 simulations. For a given triplet of parameters $(\sigma, \sigma_q, \sigma_v)$, every simulation is unique and provides a subsequent evolution of the Dice coefficient over the iterations. 100 simulations are performed for a each set of parameters and the Dice coefficient at each time step is averaged over all the simulations.



(a) $t_{100}, \sigma = 2.0$.

(b) $t_{100}, \sigma = 6.0$.

**Fig. 10** Noisy images with $\sigma = 2.0, 6.0$.

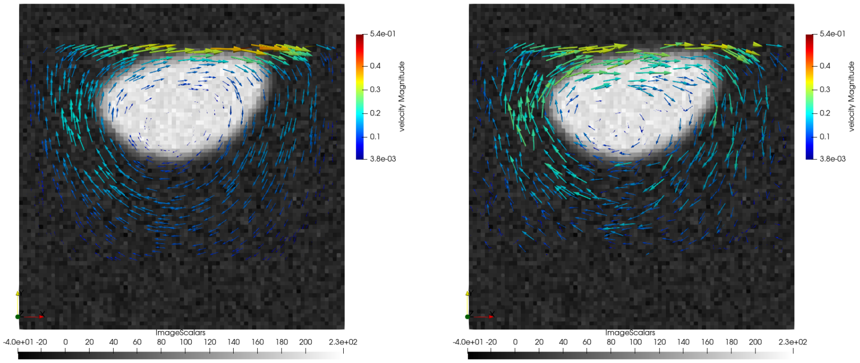


(a) $t_{100}, \sigma = 2.0$.

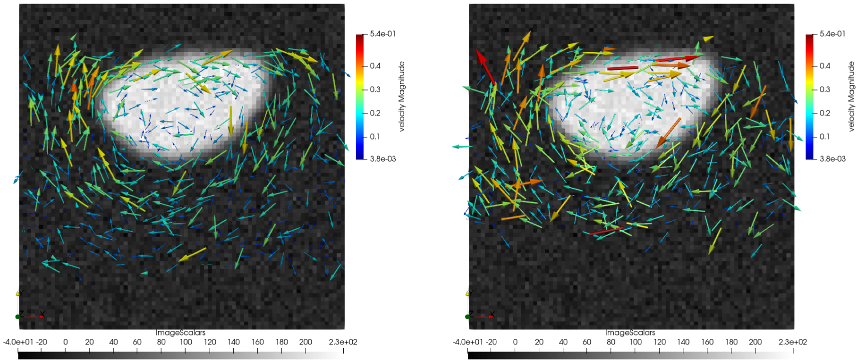
(b) $t_{100}, \sigma = 6.0$.

**Fig. 11** Gradient magnitude images with $\sigma = 2.0, 6.0$.

In addition to measuring the accuracy of the surface segmentation with the evolution of the dice coefficient, we also check the point-to-point tracking on single simulations. For $n \geq 1$, the exact position of the tracked surface $\Gamma^n$ is discretized in terms of a set of points $\{\boldsymbol{q}_j^n\}_{j=0,\cdots,N_E}$, and $\{\hat{\boldsymbol{q}}_i^n\}_{i=0,\cdots,N_{\mathrm{P}}}$ denote the estimation provided by Algorithm 2. The quality of the point-to-point tracking of $\Gamma^n$ by $\{\hat{\boldsymbol{q}}_i^n\}_{i=0,\cdots,N_{\mathrm{P}}}$ is measured by comparing the trajectories of two points whose labels $i_{\min}, i_{\max} \in \{0, \cdots, N_{\mathrm{P}}\}$ respectively to the trajectories of



(a) $t_{75}, \sigma = 2.0, \sigma_v = 0.003(1\%)$. (b) $t_{75}, \sigma = 2.0, \sigma_v = 0.030(10\%)$.



(c) $t_{75}, \sigma = 2.0, \sigma_v = 0.075(25\%)$. (d) $t_{75}, \sigma = 2.0, \sigma_v = 0.105(35\%)$.

**Fig. 12** The velocity field around the vesicle, images with $\sigma = 2.0$ and $\sigma_v = 0.003, 0.030, 0.075, 0.105$.

$j_{\min}, j_{\max} \in \{0, \cdots, N_E\}$. $i_{\min}, i_{\max}$ are defined as follows:

$$i_{\min} := \underset{i \in \{0,\cdots,N_{\mathrm{P}}\}}{\arg\min} \; d(\hat{\boldsymbol{q}}_i^{N_T}, \Gamma^{N_T}), \qquad i_{\max} := \underset{i \in \{0,\cdots,N_{\mathrm{P}}\}}{\arg\max} \; d(\hat{\boldsymbol{q}}_i^{N_T}, \Gamma^{N_T}),$$

where the symbol $d(\boldsymbol{q}, S)$ indicates the distance between the surface $S$ and the point $\boldsymbol{q}$. $j_{\min}$ and $j_{\max}$ are the labels of the closed-point projections on $\Gamma^{N_T}$ of the points $\hat{\boldsymbol{q}}_{i_{\min}}^{N_T}$, respectively $\hat{\boldsymbol{q}}_{i_{\max}}^{N_T}$. Note that, the discrete trajectories $\{\hat{\boldsymbol{q}}_{i_{\min}}^n\}_{n=1,\cdots,N_T}$ and $\{\hat{\boldsymbol{q}}_{i_{\max}}^n\}_{n=1,\cdots,N_T}$ respectively represent the best and worst choice for error quantification. When using Kalman filter, the presumably tracked point on $\Gamma^{N_T}$ by $\{\hat{\boldsymbol{q}}_i^n\}_{i=0,\cdots,N_{\mathrm{P}}}$ can only be assessed at the final iteration. In fact, the linear Kalman filter estimation improves as observations are provided, which makes it the most accurate at final iteration. In the result section, the plots of the discrete trajectories $\{\hat{\boldsymbol{q}}_{i_{\min}}^n\}_{n=1,\cdots,N_T}$ compared to $\{\boldsymbol{q}^n_{j_{\min}}\}_{n=1,\cdots,N_T}$ is referred as "min" trajectories, and $\{\hat{\boldsymbol{q}}^n_{i_{\max}}\}_{n=1,\cdots,N_T}$ compared to $\{\boldsymbol{q}^n_{j_{\max}}\}_{n=1,\cdots,N_T}$ as "max" trajectories.

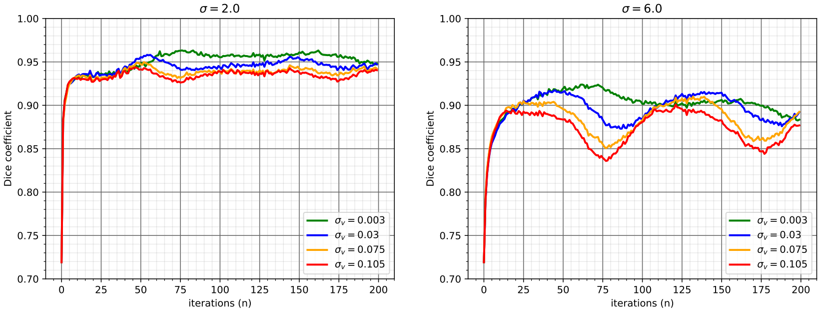
### *Choice of the free parameters*

The parameters $\sigma$ and $\sigma_v$ are used in this test case to control the data quality. However the user-defined parameter $\sigma_p$ is given to the Kalman filter as an expected level of noise in the velocity data. We assume that at each time, the noise in the each velocity component is Gaussian with a standard deviation $\sigma_p$. Here due to the specific synthetic case, we already have $\sigma_p = \sigma_v$, but in real application, this parameter has to be calculated by the user. Even if the noise of the velocity has a very complex origin and structure, a reasonable value for $\sigma_p$ can be proposed by calculating the standard deviation of the velocity field in a region of the image known to be a white noise area. The parameters to quantify the level of confidence that we have in the initialization are $\sigma_{P_0,q}$ and $\sigma_{P_0,p}$. $\sigma_{P_0,q}$ is expressed in voxel/pixel units. Depending on the origin of the initial configuration, t can be taken more or less close to 1. In the vesicle test case, it is taken quite large, because we force a bias of the initial configuration compared to the real one. When it comes to the aorta, we recommend a value close between 1-3 if the initial configuration has been segmented from 4D-flow MRI, or any derived image sequence (PC-MRA). $\sigma_{P_0,q}$ has to be taken very high, because the default initial value for the velocity of the aortic wall is null. We have no reasonable estimation of the initial velocity of the aortic wall. The critical parameters of the method are the model parameters. The parameter $\kappa$ models the vessel elasticity in a simplified way and contributes to regularize the prediction of the wall motion. $\sigma_a$ is proportional to the norm of the model error covariance matrix $\mathsf{W}_{\text{model}}$ and ponders the confidence in the model prediction compared to the confidence in the observation. The larger $\kappa$ is, the more rigid the predicted surface will be. Ultimately, compared to standard semi-automatic segmentation method, few parameters have an incidence on the quality of the tracking, and even for those parameters, it is possible to find through numerical experiments stable range of variable which results in equivalent quality of the reconstruction.

### *Results*

We present the results of the numerical tests on the method in this section. For every couple of parameters $(\sigma, \sigma_v)$, 100 simulations are performed and the Dice coefficient is systematically averaged per time iteration, given all the results of the simulations. The evolution of the averaged Dice coefficient from the initial iteration to the final iteration is plotted for $\sigma = 2.0, 6.0$ with different levels of velocity noise, see Figure 13.

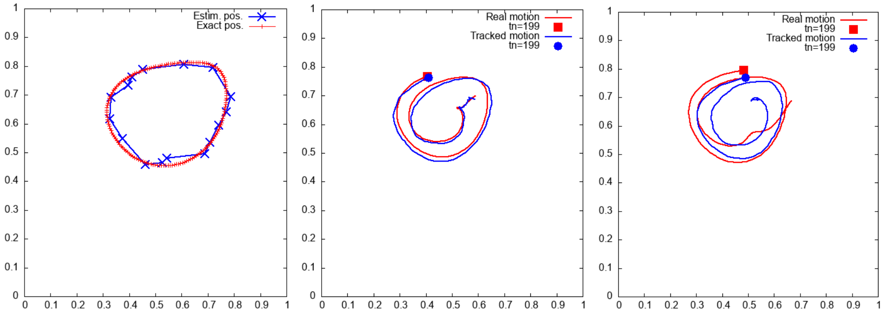
For $\sigma = 2.0$, the averaged Dice coefficient shows a satisfactory surface reconstruction. The Dice coefficient goes from 0.72 (initial value) to 0.92 within 5 iterations, see Figure 13(a). It reaches a narrow range of variation, after only 5-10 iterations. After 60 iterations over 200, the level of similarity is preserved throughout the iterations. The final averaged Dice coefficient is between 0.94



(a) Image blur $\sigma = 2.0$. (b) Image blur $\sigma = 6.0$.

**Fig. 13** Dice evolution, averaged over 100 runs. $\gamma = 100$, $\kappa = 10$, $\kappa_{\text{obs}}\delta = 0.1$.

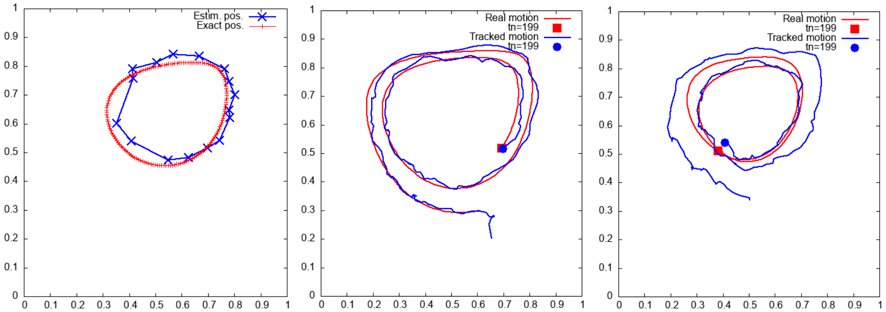
and 0.95. The same trend is observed for the different levels of velocity noise. Note that the lower the level of noise is, the higher the Dice coefficient is in average. For $\sigma = 2$, the gradient magnitude profile is preserved despite the blur and the noise, see Figure 10(a). The contours are actually easily detectable to naked eyes. The observation operator finds the position of the surface, hence the fast convergence towards a large Dice coefficient. Regarding the Dice coefficients, the method shows equivalent level of precision for different levels of noise, going from a low velocity noise $\sigma_v = 0.003$ to more severe noise $\sigma_v = 0.105$, as shown in Figure 13(a). However the point trajectory plots are more sensitive to the velocity noise, as shown by Figures 14 and 15, in terms of regularity.



(a) Final state. (b) "Min" trajectory. (c) "Max" trajectory.

**Fig. 14** Simulation for $\sigma = 2.0$, $\sigma_v = 0.003$, $\sigma_q = 1.5$. Red : real vesicle. Blue : estimation by the method. $\kappa = 10$, $\kappa_{\text{obs}}\delta = 0.1$.

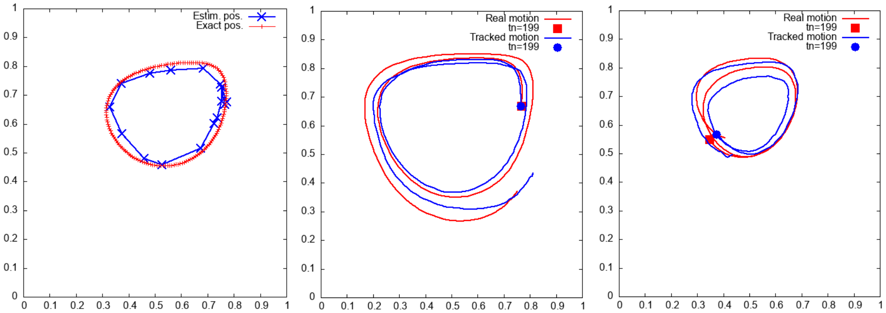
For $\sigma = 6$, an accurate contour detection becomes more challenging, as shown by Figures 10(b) and 11(b) because of the blur in the image. The Dice coefficient requires 10-15 iterations to reach a lower level compared to $\sigma = 6$, namely a value of 0.9. The final Dice coefficient is between 0.85 and 0.90, see



(a) Final state. (b) "Min" trajectory. (c) "Max" trajectory.

**Fig. 15** Simulation for $\sigma = 2.0$, $\sigma_v = 0.105$, $\sigma_q = 1.5$. Red : real vesicle. Blue : estimation by the method. $\kappa = 10$, $\kappa_{\text{obs}}\delta = 0.1$.

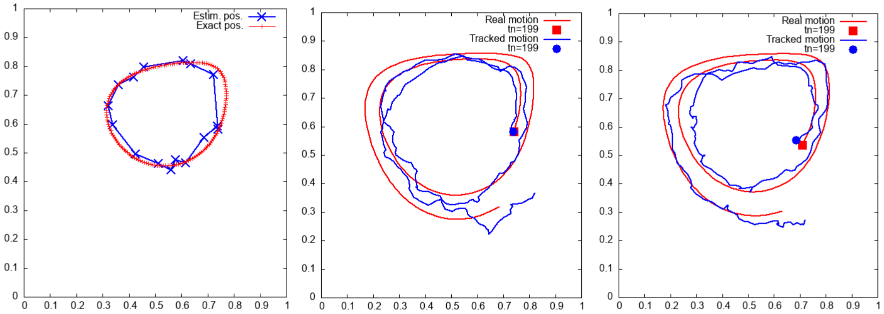
Figure 13(b). The method is globally less accurate in terms of surface reconstruction and point-to-point trajectory tracking, as shown by Figure 16, due to a more diffused image gradient. The behavior of the method differentiates with the noise in the velocity, at least between $\sigma_v = 0.003$ and $\sigma_v > 0.003$. For $\sigma = 6$ large oscillations are observed as the noise in the velocity increases. The point-to-point tracking is more erratic for high velocity noise, see Figure 17.



(a) Final state. (b) "Min" trajectory. (c) "Max" trajectory.

**Fig. 16** Simulation with $\sigma = 6.0$, $\sigma_v = 0.003$, $\sigma_q = 2.5$. Red : real vesicle. Blue : estimation by the method. $\kappa = 10$, $\kappa_{\text{obs}}\delta = 0.1$.

The quality of the segmentation is assessed through the Dice coefficient which characterizes a global similarity. However a large Dice coefficient could hide variable local irregularities in the mesh such as point collapsing, edge twists. The predictive model (spring-mass model (12)) is designed to avoid point collapse, for instance. In fact large values of $\kappa$ tends to get the structure stiffer, hence rigidifying its motion. Despite this model regularization, point collapse can occur in the vesicle test case because the vesicle deforms with large amplitudes with respect to the zero-energy configuration. The model prevents vertices from collapsing by penalizing locally high elastic energy, with the



(a) Final state. (b) "Min" trajectory. (c) "Max" trajectory.

**Fig. 17** Simulation with $\sigma = 6.0$, $\sigma_v = 0.105$, $\sigma_q = 2.5$. Red : real vesicle. Blue : estimation by the method. $\kappa_{\text{obs}}\delta = 0.1$

assumption of small deformations around the circular shape. A more suitable predictive model adapted to the vesicle deformation might require to take into account the local curvature of the surface in the discrete equation (12).

### 3.1.3 Sensitivity to regularization parameters

The isolated and joint effects of the model parameter $\kappa$ and the observation parameter $\kappa_{\text{obs}}\delta$ on the accuracy of the estimation are investigated in this section.

The data quality is fixed by $\sigma = 6.0$, $\sigma_v = 0.075$, which corresponds to considerably lowered data quality. This test would conjointly serves to exhibit a range of stability of the method, if it exists. The parameters are in the CGS unit system.

| Data quality | | Initialization | | Model | | Observation | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $\sigma$ | $\sigma_v$ | $\sigma_{P_0,q}$ | $\sigma_{P_0,p}$ | $\kappa$ | $\sigma_a$ | $\gamma$ | $\Delta N$ | $\kappa_{\text{obs}}\delta$ | $\sigma_q$ | $\sigma_p$ |
| 6.0 | 0.075 | $10h_1$ | 0.1 | {0.0,1.0, 5.0, 10, 30, 50} | 5.0 | 100 | 30 | {0.0, 0.1, 0.5, 0.7, 1.0} | 1.5 | $\sigma_v$ |

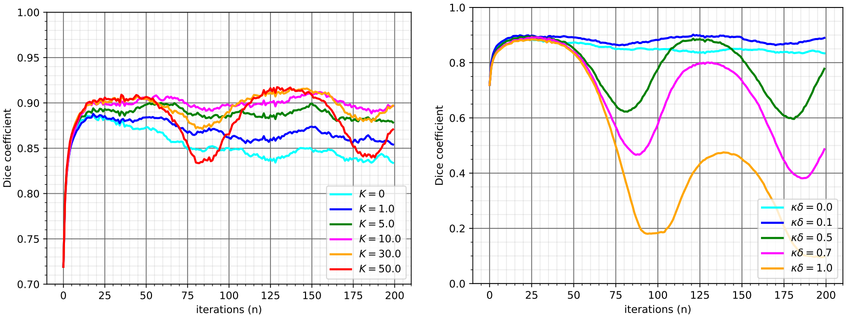
**Table 2** Parameters for the second synthetic test case.

We investigate the following values for the parameters: $\kappa = 0, 1.0, 5.0, 10, 30, 50$ and $\kappa_{\text{obs}}\delta = 0.0, 0.1, 0.5, 0.7, 1.0$. The unchanged parameters are specified in Table 2.

For this second test, we appreciate the robustness of the method, regarding the regularization parameters by considering a fixed point labeled $i^* = 12 \in \{0, \cdots, N_{\text{P}}\}$. The discrete trajectory $\{\hat{\boldsymbol{q}}_{i^*}^n\}_{n=0,\cdots,N_T}$ of $i^*$ is compared to the trajectory $\{\boldsymbol{q}_{j^*}^n\}_{n=0,\cdots,N_T}$ of a material point $j^*$ such that $\boldsymbol{q}_{j^*}^{N_T}$ is the closest point projection of $\hat{\boldsymbol{q}}_{i^*}^{N_T}$ on $\Gamma^{N_T}$. If the method is robust, then $\boldsymbol{q}_{j^*}^{N_T}$ should remain in a same vicinity for the values of parameters improving the regularity of the point trajectories.

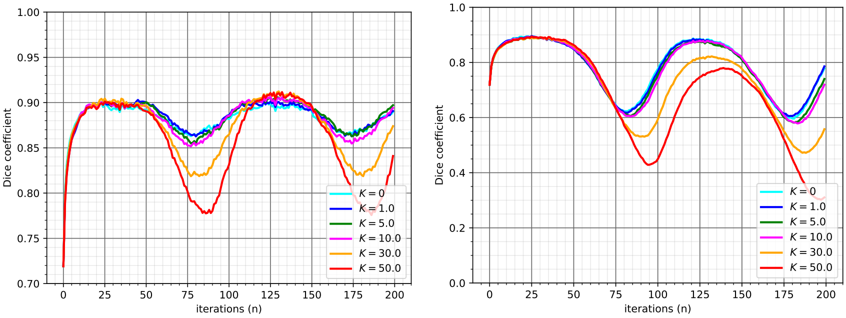
### *Results*

We present the results of the sensitivity analysis with respect to the regularization parameters. The image blur is set to $\sigma = 6$ and the noise in the velocity $\sigma_v = 0.075$.



(a) $\kappa_{\text{obs}}\delta = 0$. (b) $\kappa = 0$.

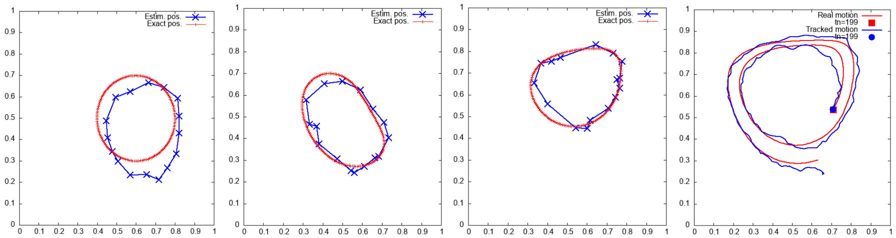
**Fig. 18** Marginal effect. $\gamma = 100$, $\sigma = 6.0$, $\sigma_v = 0.075$.



(a) $\kappa_{\text{obs}}\delta = 0.1$. (b) $\kappa_{\text{obs}}\delta = 0.5$.

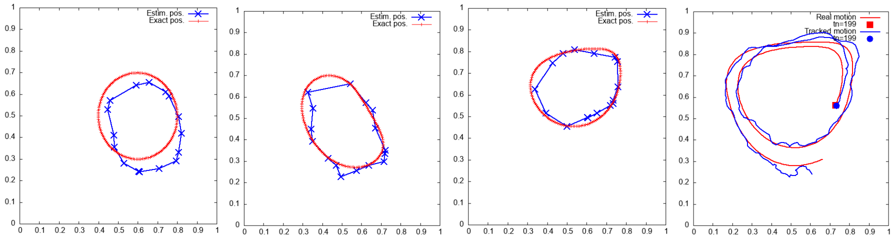
**Fig. 19** Effects of $\kappa$ and $\kappa_{\text{obs}}\delta$. $\gamma = 100$, $\sigma = 6.0$, $\sigma_v = 0.075$.

When $\kappa_{\text{obs}}\delta = 0$ , the effect of the model regularization $\kappa$ on the surface reconstruction is evaluated. Increasing $\kappa$ improves the surface reconstruction within a limited interval, as shown by Figure 18(a). From $\kappa = 0$ to $\kappa = 10$, the final averaged Dice coefficient gains more than 5.0. For larger values of $\kappa$, $\kappa > 30$, considerable oscillations of the Dice coefficient are observed during the reconstruction. It corresponds to alternative shrinking behavior, as the vesicle deforms and rotates. The vesicle actually alternates between circular and ellipsoidal configurations during its motion, as visible on Figure 7. Large values of $\kappa$ tend to shrink the capsule to a circle, since the zero-energy state of the model is the initial shape which is circular.



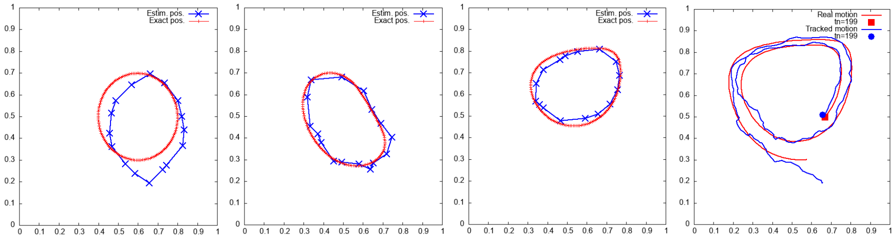
(a) Configuration at $t_1$. (b) Configuration at $t_{15}$. (c) Configuration at $t_{199}$. (d) Point to point trajectories.

**Fig. 20** Vesicle configuration (blue) at $t_n, n = 1, 69, 199$ against the solution (red). Right hand side : trajectory of a point in the estimation (blue) compared to the one of the presumed tracked point in the real vesicle (red) at final iteration. Parameters : $\kappa = 0.0$,$\kappa_{\text{obs}}\delta = 0.1$, $\sigma = 6.0$, $\sigma_v = 0.075$.



(a) Configuration at $t_1$. (b) Configuration at $t_{15}$. (c) Configuration at $t_{199}$. (d) Point to point trajectories.

**Fig. 21** Vesicle configuration (blue) at $t_n, n = 1, 69, 199$ against the solution (red). Right hand side : trajectory of a point in the estimation (blue) compared to the one of the presumed tracked point in the real vesicle (red) at final iteration. Parameters : $\kappa = 10$,$\kappa_{\text{obs}}\delta = 0.0$,$\sigma = 6.0$, $\sigma_v = 0.075$.



(a) Configuration at $t_1$. (b) Configuration at $t_{15}$. (c) Configuration at $t_{199}$. (d) Point to point trajectories.

**Fig. 22** Vesicle configuration (blue) at $t_n, n = 1, 69, 199$ against the solution (red). Right hand side : trajectory of a point in the estimation (blue) compared to the one of the presumed tracked point in the real vesicle (red) at final iteration. Parameters : $\kappa = 10$, $\kappa_{\text{obs}}\delta = 0.1$,$\sigma = 6.0$, $\sigma_v = 0.075$

When $\kappa = 0$, the effect of the observation regularization is analyzed. Note that having $\kappa = 0$ inhibits the model regularization, and more specifically the velocity regularization. Consequently, the noise in the velocity might amplify and propagate with the model integration. For $\kappa = 0.0$, $\kappa_{\text{obs}}\delta = 0.1$ and $\kappa = 10.0$, $\kappa_{\text{obs}}\delta = 0.0$, the reconstruction provides equivalent results, see Figure 20, and Figure 21. For $\kappa_{\text{obs}}\delta > 0.1$, alternative oscillations of very large amplitude are systematically observed, see Figure 18(b). They account for shrinkage behavior and also degraded surface reconstruction. The stable range of variation of $\kappa_{\text{obs}}\delta$ is limited of $0.0 - 0.1$. For larger values of $\kappa_{\text{obs}}\delta$, the numerical resolution with the algorithm provides uncontrolled results.

Concerning the combined effect of the model and the position observation regularization, oscillation of the Dice coefficient are systematically observed for $\kappa_{\text{obs}}\delta = 0.1$. In fact, the observation generates an additional recall force to the circular initial shape when $\kappa_{\text{obs}}\delta$ is large. The algorithm is unable to properly capture the deformation of the surface. The vesicle remains mostly circular and the Dice coefficient is bounded from below by the initial Dice coefficient, which ensures some robustness. As shown through Figure 19, $\kappa_{\text{obs}}\delta = 0.1$ is an upper bound in this case. The parameter $\kappa_{\text{obs}}\delta$ seems to be a very critical parameter with dramatic effect on the reconstruction. Note that large values of $\kappa_{\text{obs}}\delta$ are actually very small, which prevents the user to set $\kappa_{\text{obs}}\delta$ arbitrary.

Finally, a salient feature of Algorithm 2 is that the spectrum of covariance matrix $\mathsf{P}_n$ can be used to provide a bound on the uncertainty of the estimation. This is reported in Figure (23), which provides an $\ell^\infty$ bound of the discrepancy between the estimated and actual positions at final time, for the for the set of parameters $\sigma_v = 0.03$, $\sigma = 2$ and $\sigma_q = 1.5$.

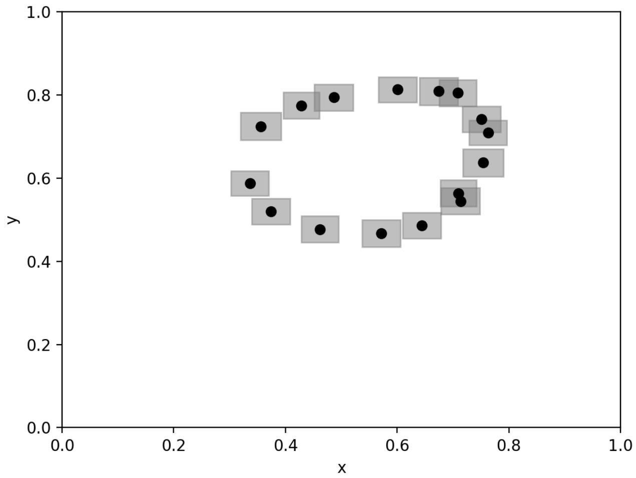


**Fig. 23** Uncertainty of the position estimation at the final time, for $\sigma_v = 0.03$, $\sigma = 2$ and $\sigma_q = 1.5$.

## 3.2 Aorta tracking from 4D-flow MRI data

In the section, we apply Algorithm 2 to the dynamical reconstruction of the aorta from a real 4D-MRI exam, provided by Dijon University Hospital.

### 3.2.1 Data generation

The 4D-flow MRI acquisition was performed with complete volumetric coverage of the thoracic aorta, on a patient with a dilatation of the ascending aorta. The MRI System of acquisition was a 3T Skyra Siemens. The data were provided and anonymized by Dijon University Hospital. The considered patient was part of a research protocol[5] which has been approved by the local ethics committee. The 4D-flow MRI data include:

- Time-resolved magnitude images of the volume of acquisition;
- Three series of time resolved images encoding the three directions of the velocity. The velocity encoding (VENC) is $2500\,\mathrm{mm}\cdot s^{-1}$, which ensures proper representation of velocities in the range $\pm 2500$ $\mathrm{mm}\cdot s^{-1}$.

The image parameters are the following:

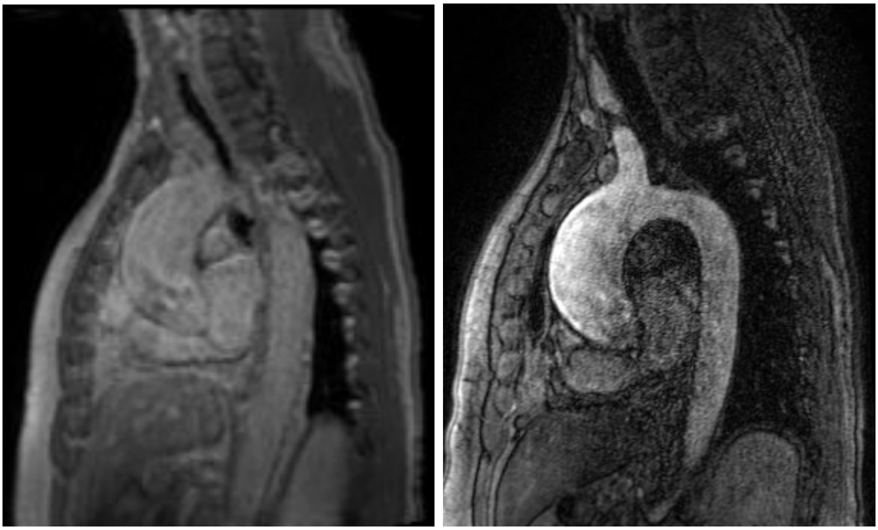
- Temporal resolution: $\Delta T = 43.319\,\mathrm{ms}$, with $N_T = 25$ frames;
- Spatial resolution (spacing): $1.988 \times 1.988 \times 2.0\,\mathrm{mm}^3$.

In addition to those images, a contrast enhanced MR angiography (MRA), representing an averaged image of the volume of acquisition, obtained with breath hold was also provided, for the same patient. The spatial resolution of the MRA is $1.0\times1.0\times1.0$ $\mathrm{mm}^3$. The image quality of MRA is more suitable for both vessel visualization and segmentation than 4D Flow MRI, due to the contrast as visible on Figure 24. The angiographic series has been in fact acquired right after the injection of contrast agent. Note that a direct segmentation of an aortic configuration directly from 4D anatomic series (Figure 24(a)) at any time of the cardiac cycle is challenging, if only based on images, because of the low image quality.

Parsing, reading and extraction of images from the Dicom files were performed using QIR 4D, a 4D MRI post-processing software developed by the CASIS company. The magnitude series and the flow series were exported as VTI files and handled with the VTK Library. No image filter (Gaussian, median, etc.) has been applied to the magnitude series. The MRA exam was segmented and reconstructed with the Mimics/Materialise software, and smoothened with Autodesk Meshmixer's tools. The inlet and outlets of the aortic mesh have then been cut by perpendicular planes to the centerlines at the level of the inlet (ascending aorta) and the four outlets (three carotid vessels on the aortic arch and descending aorta). Semi-automatic rigid registration to the first magnitude image has been performed on the resulting geometry to serve as initial configuration for Algorithm 2, Figure 25 summarizes the corresponding input data in the present example:

---

[5] See clinicalTrials.gov with protocol code NCT03817008.



(a) 4D-MRI magnitude image at $t_3$. (b) MRA images.

**Fig. 24** Slice of magnitude series and MRA series. The view is oblique sagittal. The slices are taken in the middle of the respective volume of acquisitions.

1. The mesh point positions $\left\{\hat{\boldsymbol{q}}_i^0\right\}_{i=1,\cdots,N_{\mathrm{P}}}$ of the initial configuration obtained after MRA segmentation, reconstruction and registration;
2. Time-dependent grayscale image $\left\{F_{\mathrm{M}}^n\right\}_{n=0,\cdots,N_T}$ provided by the raw magnitude series;
3. Measured velocity field $\left\{\boldsymbol{u}_{\mathrm{M}}^n\right\}_{n=0,\cdots,N_T}$ from the three time-dependent flow series.

One can refer to Figure 26 for a three dimensional visualization of the (noisy) velocity field provided by 4D Flow MRI.

### 3.2.2 Description of the test

As indicated above, the initial mesh of the aorta comes from MRA data. As shown on Figure 27(A.), the initial mesh is a reasonable guess for the initial state of the tracking. There is in fact a good match between the slice cut of the initial mesh and the position of the aortic wall on the anatomic series at time $t_0$.

Note that the largest deformations of the aortic wall during the cardiac cycle are observed close to the inlet (see Figures 27(A.–C.)), with a maximum displacement at peak systolic phase. The systolic phase goes approximatively from time iteration $t_1$ to $t_{10}$, with the peak systolic at $t_4$, as assessed by the flow rate calculation at the inlet (see Figure 28).

The inlet motion observed in the grayscale images requires a proper tracking, which is the purpose of Algorithm 2. As shown on Figure 26(a), the velocity

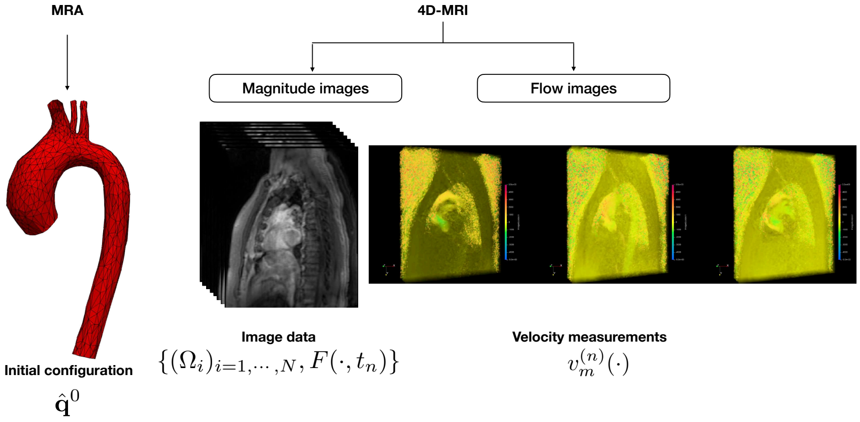
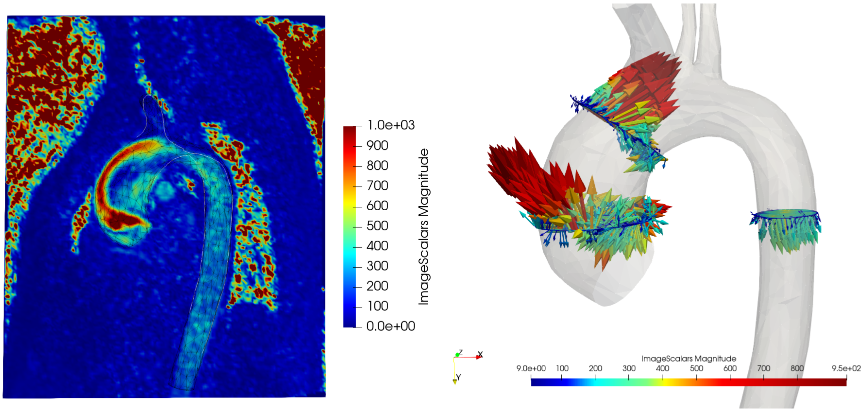


**Fig. 25** Input data for the proposed method.



(a) Slice view of the velocity magnitude. (b) velocity field visualization at 3 slice cuts.

**Fig. 26** Snapshots of the 4D flow MRI velocity field data at peak systolic phase.

field is noisy, especially around the aorta. The mesh consists of $N_{\mathrm{P}} = 793$ points and 1582 triangles. The state estimation $\hat{\mathbf{X}}$ is of size $6N_{\mathrm{P}} = 4758$.

Compared to the reconstruction of the motion of the immersed vesicle example (Section 3.1), the tracking of the aorta has some specificities. The aorta is surrounded by other vessels and cavities, whose voxel intensities are also sensitive to blood flow over time (the principle of MRI). Consequently the local window search, introduced to avoid undesired segmentation of neighboring vessels, has to be limited to 1 or 2 voxels. Additionally, the gradient will be normalized by the maximum value inside the local window search instead

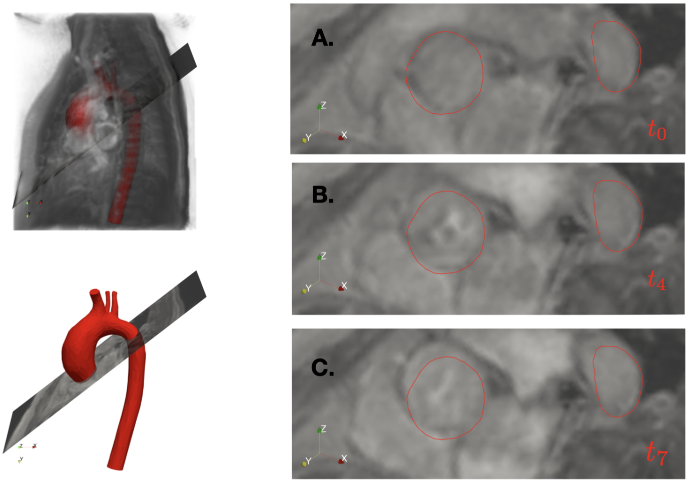


**Fig. 27** Slice view of the initial configuration against the magnitude images of the 4D MRI series, at time iteration $t_0$ (A.), $t_4$ (B.) and $t_7$(C.).

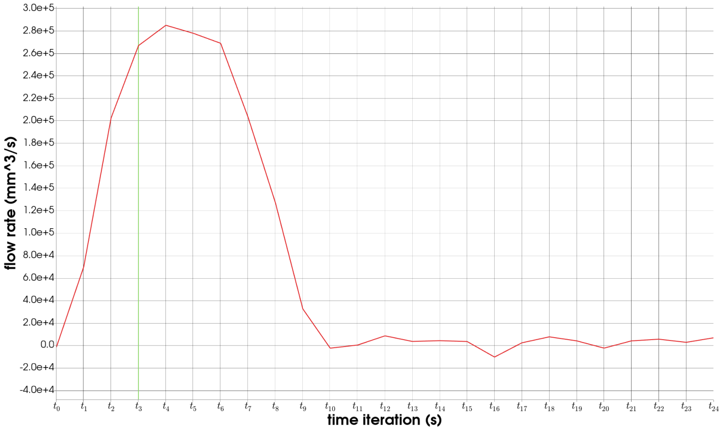


**Fig. 28** Flow rate at the inlet of the initial configuration. Time iteration $t_k = k \cdot \Delta T$.

of the maximum value of the volume of acquisition in the position observation operator (16) of Algorithm 2. The purpose is to reduce the sensitivity of the position observation to distant and high image intensity variations. Except those specificities, the parameters that will be tested are almost the same as in the immersed vesicle example:

- $\kappa$, which represents the stiffness of the aorta, modeled as spring-mass network. After some preliminary tests, we found appropriate range of variation for $\kappa$, namely between $5.0 \cdot 10^3$ and $5.0 \cdot 10^4$;
- $\sigma_a$, which indirectly quantifies the level of the confidence in the observation compared to the predictive model. The formula (26) provides a reasonable

order of magnitude for $\sigma_a$, namely $\sigma_a \sim 10^4$, N=2. It assumes that the uncertainty induced by the model is comparable to N-voxel size. We have performed a few tests around $10^4$ and we propose $1.0 \cdot 10^3 < \sigma_a < 1.0 \cdot 10^4$.

- $\gamma$, $\kappa_{\mathrm{obs}}$, the first one weights the gradient compared to the distance to the prior in the local maximization of the gradient (17), the second parametrizes the regularity of the position observation outcome (18).
- $\sigma_p$, the global level of noise in the velocity data. The order of magnitude of level noise $\sigma_p$ for the velocity is assessed by estimating the standard deviation of each component of the velocity field in a region of interest in the middle in the descending aorta at the end the diastolic phase. We found approximatively 50 mm/s. The values obtained are possibly underestimated since they do not take into account the noise around the aorta, which is much higher, see Figure 26(a).

The parameters of the filter are given in the Table 3 (length in mm and time in s).

| Initialization | | Model | | Observation | | | | |
|---|---|---|---|---|---|---|---|---|
| $\sigma_{P_0,q}$ | $\sigma_{P_0,p}$ | $\kappa$ | $\sigma_a$ | $\gamma$ | $\Delta N$ | $\kappa_{\mathrm{obs}}\delta$ | $\sigma_q$ | $\sigma_p$ |
| 2.0 | 500 | $5.0 \cdot 10^3$ | $1.0 \cdot 10^3$ | 20 | 1 | 5.0 | 1.0 | 75 |
| (voxels) | (mm/s) | $1.0 \cdot 10^4$ | $1.0 \cdot 10^4$ | 40 | (voxels) | 20 | | 100 |
| | | $5.0 \cdot 10^4$ | mm/s$^2$ | 80 | | 40 | | (mm/s) |

**Table 3** Parameters used in Algorithm 2 for the reconstruction of the aorta.

*Remark 3* For the 4D MRI exam provided, note that the observation of the carotid vessels on the aortic arch from the magnitudes images is very challenging, due to the limited contrast and the spacial resolution. This can be inferred from Figure 24, by comparing the magnitude of the 4D Flow exam with the angiographic series.

### 3.2.3 Results

The purpose of this section is to show the behavior of the tracking algorithm and to provide insight on the sensitivity of the reconstruction with respect to the user-defined parameters. Algorithm 2 returns a sequence of time-dependent point-to-point displacements from the initial configuration, with the associated velocity field. For each set of parameters, the quality of the reconstruction is assessed by computing the Dice coefficient at time $t_3$. To this purpose, we have performed a manual segmentation of the aorta (at time $t_3$) as ground-truth. This manual segmentation is partial (see Figure 29(A)), since one carotid on the aortic arch is not visible due to the quality of the 4D MRI magnitude series. The Dice coefficient is evaluated within a region of interest (ROI) which contains the ascending aorta, the aortic arch, and the beginning of the descending aorta (see Figure 29(C)).

Figure 30 reports the values of the Dice obtained over all the reconstructions performed with the parameters specified in Table 3. The average Dice on

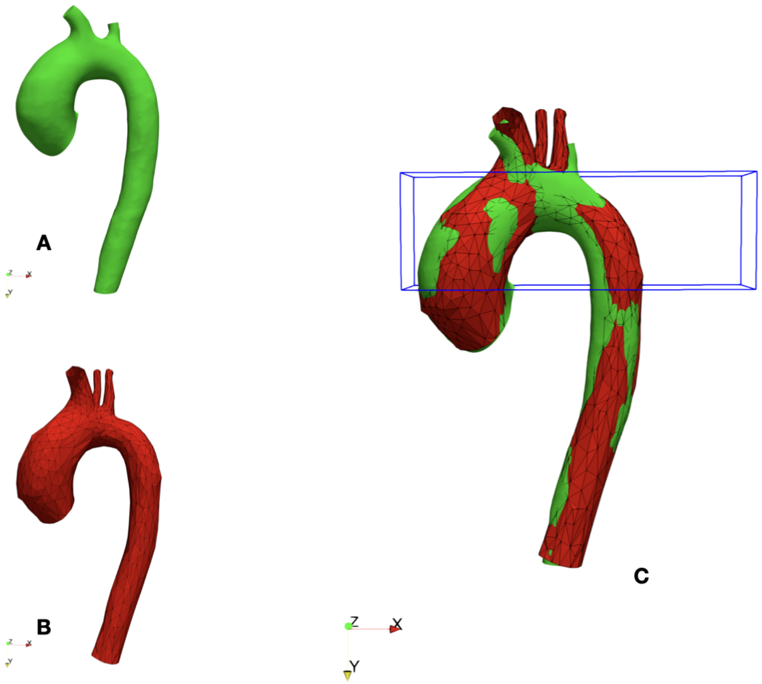


**Fig. 29** Definition of the ROI and the slice plane location for the computation of the Dice coefficient. A: the aorta from manual segmentation at $t_3$ (green). B: the reconstruction at $t_3$ (red). C: The region of interest for the calculation of the Dice in blue.

the ROI is $0.929 \pm 0.00367$, with as maximum Dice 0.935 for $\kappa = 5.0 \cdot 10^3 \mathrm{s}^{-2}$, $\gamma = 80$, $\kappa_{\mathrm{obs}}\delta = 10$, $\sigma_a = 1.0 \cdot 10^4\,\mathrm{mm} \cdot \mathrm{s}^{-2}$, $\sigma_p = 100\,\mathrm{mm} \cdot \mathrm{s}^1$, and a minimum Dice 0.920 for $\kappa = 1.0 \cdot 10^4 \mathrm{s}^{-2}$, $\gamma = 20$, $\kappa_{\mathrm{obs}}\delta = 20$, $\sigma_a = 1.0 \cdot 10^3\,\mathrm{mm} \cdot \mathrm{s}^{-2}$, $\sigma_p = 75\,\mathrm{mm} \cdot \mathrm{s}^{-1}$. As a reference the Dice of the initial configuration (registered MRA segmentation) is 0.915.

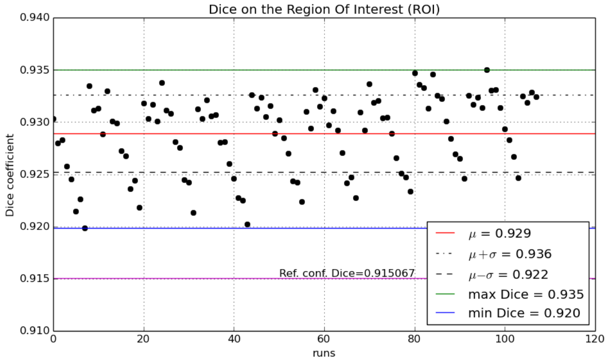


**Fig. 30** Dice on the region of interest

The reconstruction is performed with acceptable accuracy (Dice>0.920) inside the region of interest at time iteration $t_3$. For the range of parameters specified in Table 3, the reconstructions are equivalent with a Dice between 0.920 and 0.935 as shown by Figure 30 and the slice cut views in Figure 31. The best and worst configuration are defined as the ones with the maximum and minimum respectively, and an average Dice coefficient has a Dice equal to the average Dice of all simulations, namely 0.929, with a tolerance of $10^{-4}$. Table 4 summarizes the values of the best, average and worse Dice configurations with the parameters that generated them. For 3 reconstructions are of comparable quality. The significant discrepancy with the manual segmentation especially at the level of the descending aorta (DAO_P3, DAO_P4, Figure 31) can be mainly explained by the low variations of the image gradient intensity around the descending aorta.

| Configuration | Dice coefficient | $\kappa$ | $\alpha$ | $\kappa_{\text{obs}}\delta$ | $\sigma_a$ | $\sigma_p$ |
|---|---|---|---|---|---|---|
| Worse Dice | 0.920 | $1.0 \cdot 10^4$ | 20 | 20 | $1.0 \cdot 10^3$ | 75 |
| Average Dice | 0.929 | $5.0 \cdot 10^4$ | 40 | 10 | $1.0 \cdot 10^4$ | 75 |
| Best Dice | 0.935 | $5.0 \cdot 10^3$ | 80 | 10 | $1.0 \cdot 10^4$ | 100 |

**Table 4** Parameters of best,average and worse dice configurations, with Dice on the ROI at $t_3$.

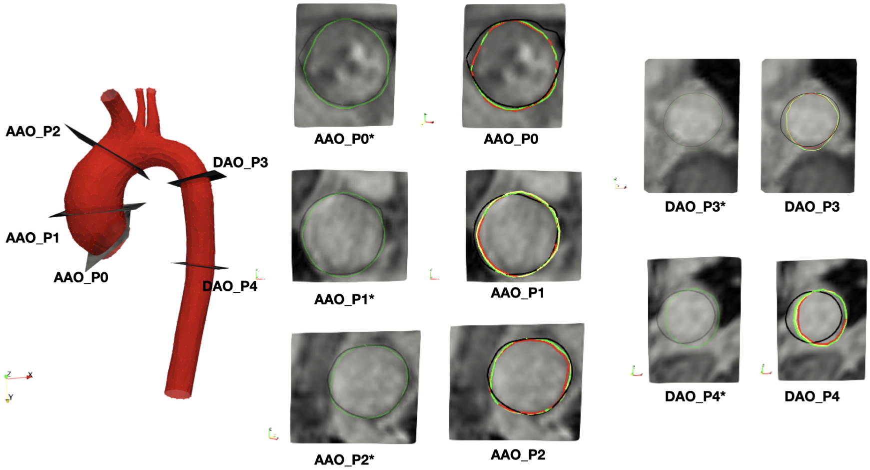


**Fig. 31** Visualization of the contours obtained by cutting the geometries at 5 planes locations : AAO_P0 = inlet of the geometry, AAO_P1, AAO_P2 = ascending aorta, DAO_P3, DAO_P4 = descending aorta. On the left, the position of the 5 planes. At the right, 10 slices cut views. For each plane, on the left side (with asterix (*)), the contour with the best Dice configuration and the manual segmentation with low opacity for better image visualization, and on the right side, the 4 contours corresponding to the worse Dice configuration (red), one of the average Dice configurations (yellow), the best Dice configuration (green), the manual segmentation (black), see Table 4 for more detail on the parameters.

Note that, compared to the vesicle test case, the Dice of the initial configuration is much higher (0.72 compared to 0.915), because a MRA exam was available, which is not always the case. Otherwise, a MRA-type image could be generated using the whole time sequences of the 4D MRI exam (see [9]) to perform segmentation, without need for further registration.

Considering the whole cardiac cycle, the tracking is qualitatively satisfying, at the level of the inlet, the beginning of the descending aorta (Figure 32) and the mid-ascending aorta (Figure 33). The slice views of the reconstruction show that the filter properly tracks the position of the surface (also observed in the vesicle example of Section 3.1). A more notable discrepancy with the image slice contour is however observed at the inlet, especially during the systolic phase (see Figure 32(B)).

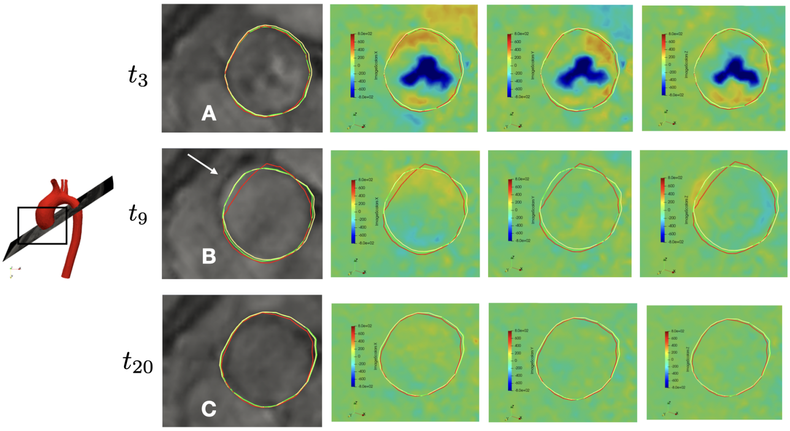


**Fig. 32** Slice cut view, at the inlet of the geometry. On the left, the position of the plane for slice visualization. On the right, the slice cut at 3 time iterations : A) $t_3$, B)$t_9$ is the end of the systolic phase, C) $t_{20}$ during the diastolic phase. The best Dice reconstruction is in green, one of the 3 average Dice reconstruction in yellow, and the worst Dice case is in red.

# 4 Conclusion

We introduced a novel approach to track the aortic wall motion from medical imaging data. Geometrical configuration and Lagrangian deformation of the wall are estimated from one patient specific reconstruction and the complete 4D MRI data set (magnitude and flow series). The magnitude image sequences are interpreted as geometrical information and the flow series as kinematical prior. The reconstruction does not require any pre-processing on the image sequences (smoothing, segmentation, contouring, etc.). The tracking is based on Kalman filter with ad hoc position observation operator which converts grayscale image sequences into a geometrical position given a patient specific prior. The Kalman filter provides a quantification of the uncertainty on the

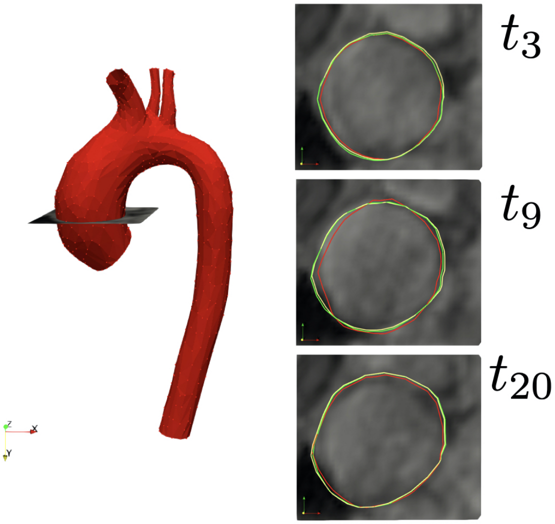


**Fig. 33** Slice cut view, at the level of the ascending aorta. On the left, the position of the plane for slice visualization. On the right, the slice cut at 3 time iterations : A) $t_3$, B)$t_9$ is the end of the systolic phase, C) $t_{20}$ during the diastolic phase. The best Dice reconstruction is in green, one of the 3 average Dice reconstruction in yellow, and the worst Dice case is in red.

reconstruction. The method was extensively investigated on a 2D synthetic example to track the deformation of a vesicle immersed in an incompressible flow. To this purpose, synthetic data has been generated to mimic 4D-flow (image sequences and volumetric velocity). The results showed that the method achieves proper tracking of the surface of the vesicle and pointed out the dependance of the parameters on the data quality (image and velocity). A promising tracking of the aorta has been obtained from real 4D-flow MRI and a patient-specific initial configuration. The quality of the tracking has been assessed by comparing with one manual reconstruction. It is important to point out that the Lagrangian reconstruction of the aortic wall returned by the tracking algorithm using Kalman filter represents an average outcome. The Kalman filter provides also a covariance matrix of the error associated to the state estimation. Assessing the quality of the local (point-to-point trajectory) tracking would be more challenging, since the literature lacks of validated point-to-point 3D reconstruction of the aorta. The proposed algorithm is currently being tested on a cohort of patients exams acquired with the same MR machine. The method is not limited to the tracking of the aorta, and might be used to reconstruct the motion of other cardiovascular parts (cardiac wall, pulmonary artery, etc.), as long as 4D MRI acquisition is available with a patient specific initial configuration. The initial configuration could be generated from different approaches (MRA, reconstructed MRA, one time reconstruction, etc.). Another interesting point that deserves investigation is the assessment of the impact of the discretization on the quality of the estimation. Preliminary results in the 2D example indicate that spatial refinement might improve the estimation. In particular, we observe that there is a range of grid refinements for which the estimations are satisfactory. A forthcoming extension of this work will address state estimation in fluid-structure simulations of the aorta from 4D MRI data, by exploiting the complete dynamical information obtained with the present tracking approach.

# Conflict of interest disclosures

M. Agbalessi, A. Lalande, O. Bouchot, T. Hayase, J.-J. Christophe, M.A. Fernández and D. Lombardi declare that they have no conflict of interest.

# References

[1] Davies, R.R., Goldstein, L.J., Coady, M.A., Tittle, S.L., Rizzo, J.A., Kopf, G.S., Elefteriades, J.A.: Yearly rupture or dissection rates for thoracic aortic aneurysms: simple prediction based on size. The Annals of Thoracic Surgery **73**(1), 17–28 (2002)

[2] Kühnl, A., Erk, A., Trenner, M., Salvermoser, M., Schmid, V., Eckstein, H.-H.: Incidence, treatment and mortality in patients with abdominal aortic aneurysms. Deutsches Arzteblatt international **114**(22-23), 391–398 (2017)

[3] Chaikof, E.L., Dalman, R.L., Eskandari, M.K., Jackson, B.M., Lee, W.A., Mansour, M.A., Mastracci, T.M., Mell, M., Murad, M.H., Nguyen, L.L., Oderich, G.S., Patel, M.S., Schermerhorn, M.L., Starnes, B.W.: The society for vascular surgery practice guidelines on the care of patients with an abdominal aortic aneurysm. Journal of Vascular Surgery **67**(1), 2–772 (2018). https://doi.org/10.1016/j.jvs.2017.10.044

[4] Authors/Task Force members, Erbel, R., Aboyans, V., Boileau, C., Bossone, E., Bartolomeo, R.D., Eggebrecht, H., Evangelista, A., Falk, V., Frank, H., Gaemperli, O., Grabenwöger, M., Haverich, A., Iung, B., Manolis, A.J., Meijboom, F., Nienaber, C.A., Roffi, M., Rousseau, H., Sechtem, U., Sirnes, P.A., Allmen, R.S.v., Vrints, C.J.M., for Practice Guidelines (CPG), E.C., Zamorano, J.L., Achenbach, S., Baumgartner, H., Bax, J.J., Bueno, H., Dean, V., Deaton, C., Erol, Ç., Fagard, R., Ferrari, R., Hasdai, D., Hoes, A., Kirchhof, P., Knuuti, J., Kolh, P., Lancellotti, P., Linhart, A., Nihoyannopoulos, P., Piepoli, M.F., Ponikowski, P., Sirnes, P.A., Tamargo, J.L., Tendera, M., Torbicki, A., Wijns, W., Windecker, S., reviewers, D., Nihoyannopoulos, P., Tendera, M., Czerny, M., Deanfield, J., Mario, C.D., Pepi, M., Taboada, M.J.S., Sambeek, M.R.v., Vlachopoulos, C., Zamorano, J.L., Grimm, M., Musayev, O., Pasquet, A., Kušljugić, Z., Cikes, M., Georghiou, G.P., Stasek, J., Molgaard, H., Kõvask;, S., Kytö, V., Jondeau, G., Bakhutashvili, Z., von

Kodolitsch, Y., Tsioufis, C., Temesvári, A., Rubinshtein, R., Antonini-Canterin, F., Lunegova, O., Stradins, P., Chammas, E., Jonkaitiene, R., Cassar, A., Bjørnstad, K., Widenka, K., Sousa Uva, M., Lighezan, D., Perunicic, J., Madaric, J., Vilacosta, I., Bäck, M., Mahdhaoui, A., Demirbag, R., Kravchenko, I.: 2014 ESC Guidelines on the diagnosis and treatment of aortic diseases: Document covering acute and chronic aortic diseases of the thoracic and abdominal aorta of the adult-The Task Force for the Diagnosis and Treatment of Aortic Diseases of the European Society of Cardiology (ESC). European Heart Journal **35**(41), 2873–2926 (2014) https://arxiv.org/abs/https://academic.oup.com/eurheartj/article-pdf/35/41/2873/17898679/ehu281.pdf. https://doi.org/10.1093/eurheartj/ehu281

[5] Rozado, J., Martin, M., Pascual, I., Hernandez-Vaquero, D., Moris, C.: Comparing american, european and asian practice guidelines for aortic diseases. Journal of thoracic disease **9**(Suppl 6), 551–560 (2017)

[6] Pape, L.A., Tsai, T.T., Isselbacher, E.M., Oh, J.K., O'Gara, P.T., Evangelista, A., Fattori, R., Meinhardt, G., Trimarchi, S., Bossone, E., Suzuki, T., Cooper, J.V., Froehlich, J.B., Nienaber, C.A., Eagle, K.A.: Aortic diameter &#x2265;5.5 cm is not a good predictor of type a aortic dissection. Circulation **116**(10), 1120–1127 (2007) https://arxiv.org/abs/https://www.ahajournals.org/doi/pdf/10.1161/CIRCULATIONAHA.107.702720. https://doi.org/10.1161/CIRCULATIONAHA.107.702720

[7] Hope, M.D., Sigovan, M., Wrenn, S.J., Saloner, D., Dyverfeldt, P.: MRI hemodynamic markers of progressive bicuspid aortic valve-related aortic disease. Journal of Magnetic Resonance Imaging **40**(1), 140–145 (2014) https://arxiv.org/abs/https://onlinelibrary.wiley.com/doi/pdf/10.1002/jmri.24362. https://doi.org/10.1002/jmri.24362

[8] Stankovic, Z., Allen, B.D., Garcia, J., Jarvis, K.B., Markl, M.: 4D flow imaging with MRI. Cardiovascular diagnosis and therapy **4**(2), 173–192 (2014)

[9] Bustamante, M., Gupta, V., Carlhäll, C.-J., Ebbers, T.: Improving visualization of 4D flow cardiovascular magnetic resonance with four-dimensional angiographic data: generation of a 4D phase-contrast magnetic resonance CardioAngiography (4D PC-MRCA). Journal of Cardiovascular Magnetic Resonance **19**(1), 47 (2017). https://doi.org/10.1186/s12968-017-0360-8

[10] Lee, S.-H., Lee, S.: Adaptive kalman snake for semi-autonomous 3d vessel tracking. Computer Methods and Programs in Biomedicine **122**(1), 56–75 (2015). https://doi.org/10.1016/j.cmpb.2015.06.008

[11] Moccia, S., De Momi, E., El Hadji, S., Mattos, L.S.: Blood vessel segmentation algorithms — review of methods, datasets and evaluation metrics. Computer Methods and Programs in Biomedicine **158**, 71–91 (2018). https://doi.org/10.1016/j.cmpb.2018.02.001

[12] Zhao, F., Zhang, H., Wahle, A., Scholz, T., Sonka, M.: Automated 4d segmentation of aortic magnetic resonance images, pp. 247–256 (2006). https://doi.org/10.5244/C.20.26. British Machine Vision Conference

[13] Lim, C.W., Su, Y., Yeo, S.Y., Ng, G.M., Nguyen, V.T., Zhong, L., Tan, R.S., Poh, K.K., Chai, P.: Automatic 4d reconstruction of patient-specific cardiac mesh with 1-to-1 vertex correspondence from segmented contours lines. PLOS ONE **9**(4), 1–14 (2014). https://doi.org/10.1371/journal.pone.0093747

[14] Wang, X., Mihalef, V., Qian, Z., Voros, S., Metaxas, D.: 3D cardiac motion reconstruction from CT data and tagged MRI. In: 2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pp. 4083–4086 (2012). https://doi.org/10.1109/EMBC.2012.6346864

[15] Seungbin Ko: Improvement of hemodynamic analysis by 4d flow MRI: Reynolds resolution, partial volume effect, carotid flow. Publisher: Hanyang University

[16] Puiseux, T.: Numerical simulations for phase-contrast magnetic resonance imaging. Theses, Université Montpellier (November 2019). https://tel.archives-ouvertes.fr/tel-02934286

[17] Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active contour models. International Journal of Computer Vision **1**(4), 321–331 (1988). https://doi.org/10.1007/BF00133570

[18] Acton, S.T.: Chapter 20 - diffusion partial differential equations for edge detection. In: Bovik, A. (ed.) The Essential Guide to Image Processing, pp. 525–552. Academic Press, Boston (2009). https://doi.org/10.1016/B978-0-12-374457-9.00020-2. https://www.sciencedirect.com/science/article/pii/B9780123744579000202

[19] Xu, C., Prince, J.L.: Generalized gradient vector flow external forces for active contours1A preliminary version of this paper appeared in the Proceedings of the Johns Hopkins University 1997 Conference of Information Sciences and Systems.1. Signal Processing **71**(2), 131–139 (1998). https://doi.org/10.1016/S0165-1684(98)00140-6

[20] Patwardhan, K.A., Yu, Y., Gupta, S., Dentinger, A., Mills, D.: 4d vessel

segmentation and tracking in ultrasound. In: 2012 19th IEEE International Conference on Image Processing, pp. 2317–2320 (2012). https://doi.org/10.1109/ICIP.2012.6467360

[21] Leo, H.-L., Canè, F., Verhegghe, B., De Beule, M., Bertrand, P.B., Van der Geest, R.J., Segers, P., De Santis, G.: From 4d medical images (ct, mri, and ultrasound) to 4d structured mesh models of the left ventricular endocardium for patient-specific simulations. BioMed Research International **2018**, 7030718 (2018). https://doi.org/10.1155/2018/7030718

[22] Wigström, L., Sjöqvist, L., Wranne, B.: Temporally resolved 3d phase-contrast imaging. Magnetic Resonance in Medicine **36**(5), 800–803 (1996) https://arxiv.org/abs/https://onlinelibrary.wiley.com/doi/pdf/10.1002/mrm.1910360521. https://doi.org/10.1002/mrm.1910360521

[23] Markl, M., Frydrychowicz, A., Kozerke, S., Hope, M., Wieben, O.: 4D flow MRI. Journal of Magnetic Resonance Imaging **36**(5), 1015–1036 (2012) https://arxiv.org/abs/https://onlinelibrary.wiley.com/doi/pdf/10.1002/jmri.23632. https://doi.org/10.1002/jmri.23632

[24] Sobel, I.: An isotropic 3x3 image gradient operator. Presentation at Stanford A.I. Project 1968 (2014)

[25] Prewitt, J.M.S.: Object enhancement and extraction. Picture Processing and Psychopictorics (1), 75–149 (1970)

[26] Canny, J.: A computational approach to edge detection. Pattern Analysis and Machine Intelligence, IEEE Transactions on **PAMI-8**, 679–698 (1986). https://doi.org/10.1109/TPAMI.1986.4767851

[27] Montagnat, J.: Modèles déformables pour la segmentation et la mod'elisation d'images m'edicales 3d et 4d. PhD thesis, Université de Nice Sophia-Antipolis (1999)

[28] Moireau, P.: A Discrete-time Optimal Filtering Approach for Non-linear Systems as a Stable Discretization of the Mortensen Observer. ESAIM: Control, Optimisation and Calculus of Variations **24**(4), 1815–1847 (2018). https://doi.org/10.1051/cocv/2017077

[29] Moireau, P.: Filtering based data assimilation for second order hyperbolic PDEs - Applications in cardiac mechanics. PhD thesis, Ecole Polytechnique X (Dec 2008). https://pastel.archives-ouvertes.fr/tel-00358582

[30] Kálmán, R.: A new approach to linear filtering and prediction problems. Transactions of the ASME–Journal of Basic Engineering **82**, 35–45 (1960)

[31] Kálmán, R., Bucy, R.: New results in linear filtering and prediction theory.

Journal of Basic Engineering **83**, 95–108 (1961)

[32] Guennebaud, G., Jacob, B., et al.: Eigen v3. http://eigen.tuxfamily.org (2010)

[33] Schroeder, W., Martin, K., Lorensen, B.: The Visualization Toolkit–An Object-Oriented Approach To 3D Graphics, 4th edn. Kitware, Inc., ??? (2006)

[34] Boilevin-Kayl, L., Fernández, M.A., Gerbeau, J.-F.: Numerical methods for immersed FSI with thin-walled structures. Computers and Fluids **179**, 744–763 (2019). https://doi.org/10.1016/j.compfluid.2018.05.024