---

**Article:**
Wang, Y (2022) A monolithic one-velocity-field optimal control formulation for fluid–structure interaction problems with large solid deformation. Journal of Fluids and Structures, 111. 103577. p. 103577. ISSN: 0889-9746

https://doi.org/10.1016/j.jfluidstructs.2022.103577

---

# A monolithic one-velocity-field optimal control formulation for fluid-structure interaction problems with large solid deformation

Yongxing Wang[a,*]

[a]*School of Computing, University of Leeds, Leeds, UK.*

**Abstract**

In this article, we formulate a monolithic optimal control method for general time-dependent Fluid-Structure Interaction (FSI) systems with large solid deformation: we consider a displacement-tracking type of objective with a constraint of the solid velocity, and tackle the time-dependent control problems by a piecewise-in-time control method; we cope with the large solid displacement using a one-velocity fictitious domain method, and solve the fully-coupled FSI and the corresponding adjoint equations in a monolithic manner. The proposed method is implemented in open-source software package FreeFEM++ and assessed by three numerical experiments, in the aspects of stability of the numerical scheme for different regularisation parameters, and efficiency of reducing the objective function with control of the solid velocity.

*Keywords:* Fluid-Structure Interaction, Optimal Control, Piecewise Control, Monolithic Method, One-Velocity Method

## 1. Introduction

Fluid-Structure Interaction (FSI) problems arise from aerodynamics [1–3], ocean mechanics [4–6], hemodynamics [7–9], and so on. For most FSI problems, analytical solutions of the controlling equations are impossible to obtain, whereas laboratory experiments are complex, expensive and limited in scope. Therefore, numerical simulations play an important role in order to understand the fundamental physics involved in the complex interaction between fluids and structures. Computational methods for FSI problems have developed rapidly in past decade and reached a significant level of maturity. A brief review and broad categorisation of the existing FSI methods can be based upon three questions: first, what kind of mesh do we use (one interface-fitted mesh, one interface-unfitted mesh or two meshes)? second, which variables do we solve (fluid velocity, pressure and solid displacement; or one velocity for both fluid and solid)? third, what type of coupling strategies do we use (monolithic/fully-coupled, or partitioned/segregated)? Therefore, a combination of the answers to these three

*Corresponding author

*Email address:* scsywan@leeds.ac.uk/yongxingwang6@gmail.com (Yongxing Wang)

questions would produce different types of numerical methods. In particular, we have classical partitioned/segregated methods [2, 10–12] using one interface-fitted mesh and solving for both the fluid velocity and solid displacement; monolithic methods [13–17] using one interface-fitted mesh and solving for fluid velocity, pressure, solid displacement and a Lagrange multiplier to enforce the continuity at the fluid-solid interface; immerse methods [18–20] or fictitious domain methods [15, 21, 22] use two meshes, the former solve for one velocity field, and the latter solve for the fluid velocity, pressure, solid displacement and a Lagrange multiplier; recent developed one-velocity methods [16, 17, 23] solve for one-velocity field in a monolithic manner using either one interfaced-fitted mesh or two meshes; there are also fully Eulerian methods [24–27] using one interface-unfitted mesh and solve for both velocity and displacement in a monolithic manner.

Optimal control is a branch of mathematical optimization which seeks to optimise an objective of a stationary or dynamical system by a control variable of the system [28]. We focus on reviewing a fluid dynamical system, in which case the objective could be reduction of the drag force by shape optimisation [3, 29–35] or by active turbulence control at the boundary layer [3, 36–40]; it could also be velocity tracking (or steering velocity) by controlling a distributed body force [41–50] or boundary force [45, 47, 51–55]; there are also objective of reducing vorticity [46, 47, 56] or matching a turbulence kinetic energy [47, 48, 50] by controlling a distributed body or boundary force. Velocity-tracking type of optimal distributed control has a rigorous mathematical theory for its solution existence [51, 52, 56, 57], and convergence and stability of the its numerical algorithm [42, 43, 53, 57].

In the context of optimal control for fluid-structure systems, the research remains limited and publications can be found in the past two decades. The earliest study of FSI control could be found in [58] where the sensitivity of a rigid body's movement inside a fluid with respect to a boundary velocity is analysed. This method was extended to another fluid-structure interaction case (fluid inside an elastic solid) in [59, 60], which is a pioneering work for FSI control through shape analysis. A general quadratic objective functional is minimised by a boundary control for a fixed solid inside a fluid, and the well-posedness of this FSI control problem is established in [61–63]. A velocity-tracking objective is optimised by controlling a boundary pressure, and the optimal control problem is first formulated and solved using Newton method in [64] for a static FSI problem, and latter extended to time-dependent case in [65]. Recently, a linearised distributed FSI control problem is analysed, and the solution existence is proved in [66]. The existence of an optimal FSI control for the problem of minimizing flow turbulence, by controlling a distributed force, is established in [67]. In recent years, many studies of the optimal FSI control focus on numerical methods and implementations: monolithic formulation and Newton multigrid method are presented in [68, 69]; a velocity-tracking objective is considered by controlling either a distributed body force [70] or a boundary pressure [71–73].

In this paper we shall apply the distributed control method to fluid-structure interaction problems, and study a displacement-tracking type of objective which has not been fully studied in literature, especially in the case of large solid deformation. To the best of the author's knowledge, up to now there is no publication concerning the control of large-deformed solid interacting with fluids. However, this displacement-tracking FSI control has potentials to

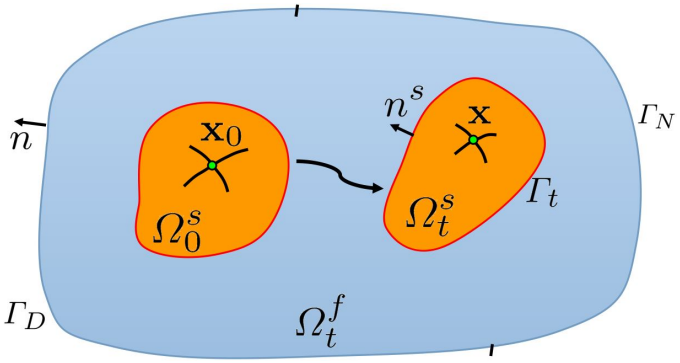



Figure 1: A sketch of FSI problems in which $\Gamma_t = \overline{\Omega}_t^f \cap \overline{\Omega}_t^s$ and $\Gamma_D \cap \Gamma_N = \Gamma$.

be applied to accurate designing and controlling a range of biologically inspired robots, such as swimming robots (observing secretive sea life or carrying out a search-and-rescue mission [74]) or micro medical robots (crawling through the human body to perform an operation or deliver a medicine [75]).

The paper is organized as follows. The control Partial Differential Equations (PDE) for the FSI problem are introduced in Section 2, followed by time discretisation of these PDEs and the optimisation problem in Section 3. The main deduction of the optimality system using the Lagrange multiplier method is presented in Section 4, and a monolithic scheme of the primal and adjoint equations is formulated in Section 5. Numerical experiments are carried out in Section 6, with conclusions drawn in Section 7.

## 2. Control partial differential equations for the fluid-structure interaction problems

We consider general fluid-structure interaction problems sketched in Figure 1, in which $\Omega_t^f \subset \mathbb{R}^d$ $(d = 2, 3)$ and $\Omega_t^s \subset \mathbb{R}^d$ are the fluid and solid domain respectively (which are time dependent regions), and $\Gamma_t = \overline{\Omega}_t^f \cap \overline{\Omega}_t^s$ is the moving interface between the fluid and solid. The superscripts $f$ and $s$ denote fluid and solid respectively, and the subscript $t$ explicitly highlights when regions are time dependent. $\Omega = \overline{\Omega}_t^f \cup \overline{\Omega}_t^s$ is a fixed domain with an outer boundary $\Gamma = \Gamma_D + \Gamma_N$, where $\Gamma_D$ is the Dirichlet boundary and $\Gamma_N$ is the Neumann boundary on which the zero-normal stress is enforced in this article.

We consider both an incompressible fluid and an incompressible hyperelastic solid in this paper, and we shall only solve for one velocity field in the whole domain. The conservation of momentum and conservation of mass take the same form in the fluid and solid, which just differs in the specific expressions of the stress tensor. Therefore, it is convenient to introduce an indicator function $1_\omega(\mathbf{x}) = 1$ if $\mathbf{x} \in \omega$ and $1_\omega(\mathbf{x}) = 0$ otherwise, and let $\rho = \rho^f 1_{\Omega_t^f} + \rho^s 1_{\Omega_t^s}$, $\mathbf{u} = \mathbf{u}^f 1_{\Omega_t^f} + \mathbf{u}^s 1_{\Omega_t^s}$, $\boldsymbol{\sigma} = \boldsymbol{\sigma}^f 1_{\Omega_t^f} + \boldsymbol{\sigma}^s 1_{\Omega_t^s}$ denote the density, velocity vector and stress tensor respectively. The control partial differential equations, with initial and boundary conditions,

for the FSI problem can then be expressed as follows.

$$\text{Momentum equation:} \quad \rho\frac{\partial \mathbf{u}}{\partial t} + \rho\left((\mathbf{u}-\mathbf{w})\cdot\nabla\right)\mathbf{u} - \nabla\cdot\boldsymbol{\sigma} = \mathbf{f}\mathit{1}_{\Omega_t^s}, \tag{1}$$

$$\text{Continuity equation:} \quad \nabla\cdot\mathbf{u} = 0, \tag{2}$$

$$\text{Initial condition:} \quad \mathbf{u}|_{t=0} = \mathbf{u}_0, \tag{3}$$

$$\text{Dirichlet BC:} \quad \mathbf{u}|_{\Gamma_D} = \bar{\mathbf{u}}\mathit{1}_{\Gamma_D}, \tag{4}$$

$$\text{Neumann BC:} \quad \boldsymbol{\sigma}\mathbf{n}|_{\Gamma_N} = \mathbf{0}, \tag{5}$$

$$\text{Continuity of velocity:} \quad \left(\mathbf{u}^s - \mathbf{u}^f\right)\big|_{\Gamma_t} = \mathbf{0}, \tag{6}$$

$$\text{Continuity of normal stress:} \quad \left(\boldsymbol{\sigma}^s - \boldsymbol{\sigma}^f\right)\mathbf{n}^s\big|_{\Gamma_t} = \mathbf{0}. \tag{7}$$

We shall use the body force $\mathbf{f}$ in (1) as a control variable in the following. The stress tensor of an incompressible Newtonian flow is expressed as:

$$\boldsymbol{\sigma}^f = \mu^f \mathrm{D}\mathbf{u}^f - p^f\mathbf{I}, \tag{8}$$

with $\mathrm{D}(\cdot) = \nabla(\cdot) + \nabla^T(\cdot)$, and $\mu^f$ being the viscosity parameter. The stress tensor of an incompressible neo-Hookean solid is expressed as [17, 23]:

$$\boldsymbol{\sigma}^s = c_1\left(\mathrm{D}\mathbf{d} - \nabla^T\mathbf{d}\nabla\mathbf{d}\right) - p^s\mathbf{I}, \tag{9}$$

where $c_1$ is the elasticity parameter and $\mathbf{d}$ is the solid displacement. Notice that although the solid stress tensor is expressed as a function of displacement $\mathbf{d}$, we shall not solve for $\mathbf{d}$ as an independent variable. Instead we view it as a function of velocity, and solve the whole FSI problem based upon a one-field-velocity method [16]. In the above equation (1), $\mathbf{w}$ is an arbitrary velocity field of the moving frame in order to describe the FSI system; $\mathbf{w} = \mathbf{0}$ in the case of Eulerian description and $\mathbf{w} = \mathbf{u}$ in the case of Lagrangian description. In the following sections, we shall use the Eulerian description for the background fluid (including the fictitious fluid covered by the solid domain) and Lagrangian description for the moving solid.

## 3. Time discretisation and the piecewise control problem

In order to introduce the piecewise-in-time control problem, we first disretise the control PDEs at a sequence of time points: $t_0 = 0, t_1, t_2\ldots$, with $t_{n+1} - t_n = \Delta t$ ($n \in \mathbb{N}_0$ is a non-negative integer). We then solve for $\mathbf{u}_{n+1}$, $\mathbf{d}_{n+1}$ and $\boldsymbol{\sigma}_{n+1}$ given $\mathbf{u}_n$, $\mathbf{d}_n$ and $\boldsymbol{\sigma}_n$ using the backward-Euler scheme. The disretised momentum equation (1) is

$$\rho\frac{\mathbf{u}_{n+1}-\mathbf{u}_n}{\Delta t} + \rho\left((\mathbf{u}_{n+1}-\mathbf{w}_{n+1})\cdot\nabla\right)\mathbf{u}_{n+1} - \nabla\cdot\boldsymbol{\sigma}_{n+1}, = \mathbf{f}_{n+1}, \tag{10}$$

and the disretised version of all the equations from (2) to (7) would take the same form except introducing a subscript $n+1$ to corresponding variables. Therefore, it is convenient to omit the subscript $n+1$ in the rest of this paper. We shall focus on computing $\mathbf{u} = \mathbf{u}_{n+1}$

given $\mathbf{u}_n$ in the time interval $[t_n, t_{n+1}]$, on which we shall also formulate an optimal control problem. Notice that $\mathbf{d}_{n+1}$ and $\boldsymbol{\sigma}_{n+1}$ are not explicit unknowns based upon the one-velocity-field formulation we shall introduce.

Let $L^2(\omega)$ be the square integrable functions in domain $\omega$ with inner product $(u,v)_\omega = \left(\int_\omega uvdx\right)$, $\forall u, v \in L^2(\omega)$, and the induced norm $\|v\|_{L^2(\omega)} = (v,v)_\omega^{1/2}$, $\forall v \in L^2(\omega)$. For a vector function $\mathbf{v} \in L^2(\omega)^d$, the norm is defined component-wise as $\|\mathbf{v}\|^2_{L^2(\omega)^d} = \sum_{i=1}^d \|v_i\|^2_{L^2(\omega)}$. Then let $H^1(\omega) = \left\{v : v, \nabla v \in L^2(\omega)^d\right\}$, and denote by $H^1_{u(\gamma)}(\omega)$ the subspace of $H^1(\omega)$, which has the boundary data $u$ on $\gamma$. We also denote by $L^2_0(\Omega)$ the subspace of $L^2(\Omega)$ whose functions have zero mean values.

We consider the following optimisation problem: reducing the discrepancy between the solid displacement $\mathbf{d}$ and an objective displacement $\mathbf{d}_g$ profile, with constraint of the solid velocity, by controlling a distributed force $\mathbf{f}$ on the solid body.

**Problem 1** (piecewise-in-time control)**.** *Given an objective displacement profile* $\mathbf{d}_g(t)$ *and objective velocity norm* $u_c(t)$ *of the solid,*

$$\underset{\mathbf{f}\in L^2(\Omega_n^s)^d}{\textit{minimise}} \quad J(\mathbf{u},\mathbf{f}) = \frac{1}{2}\|\mathbf{d} - \mathbf{d}_g(t_{n+1})\|^2_{\Omega_n^s} + \frac{\alpha}{2}\|\mathbf{f}\|^2_{\Omega_n^s}, \tag{11}$$

*subject to*

$$\|\mathbf{u}\|_{\Omega_n^s} \leq u_c(t_{n+1}), \tag{12}$$

*and equations (10); (2) to (7) after time discretisation (omitting the subscript* $n+1$ *of* $\mathbf{u}_{n+1}$ *and* $\mathbf{d}_{n+1} = \mathbf{u}_{n+1}\Delta t + \mathbf{d}_n$ *for notation simplicity).*

In the above Problem 1, we consider an optimisation problem integrated in the old time domain $\Omega_n^s$, and we shall also solve our FSI problem using this explicit formulation. It is not significant to iteratively construct $\Omega_{n+1}^s$ and perform integration on it using a small time step as pointed out in [17, 23]. The first term in (11) is the real objective to be minimised, and the second term is a regularisation term with a regularisation parameter $\alpha$. A too large $\alpha$ would make it difficult to achieve the real objective, while a too small $\alpha$ may cause convergence issues for the numerical scheme. The inequality constraint (12) provides an upper bound for the magnitude of the velocity.

## 4. The Lagrange multiplier method

In this section, we introduce the Lagrange multipliers (or adjoint variables) $\hat{\mathbf{u}} \in H^1_{0(\Gamma_D)}(\Omega)$ and $\hat{p} \in L^2(\Omega)$ to eliminate the equality constraints of Problem 1. For the inequality constraint (12) we simply introduce a penalty (or barrier) parameter $\lambda$ to be included in the Lagrangian functional. Other methods, such as active-set or trust-region algorithm [76, 77], may be used to deal with inequality constraints as well, which however would not be the

main focus of this paper.

$$
\begin{aligned}
&L\left(\mathbf{u}, p, \hat{\mathbf{u}}, \hat{p}, \mathbf{f}\right) = J(\mathbf{u}, \mathbf{f}) + \frac{\lambda}{u_c^2 - \|\mathbf{u}\|_{\Omega_n^s}^2} \\
&+\rho^f \int_{\Omega_n^f} \frac{\mathbf{u} - \mathbf{u}_n}{\Delta t} \cdot \hat{\mathbf{u}} + \rho^s \int_{\Omega_n^s} \frac{\mathbf{u} - \mathbf{u}_n}{\Delta t} \cdot \hat{\mathbf{u}} \\
&+\rho^f \int_{\Omega_n^f} \left((\mathbf{u} - \mathbf{w}) \cdot \nabla\right) \mathbf{u} \cdot \hat{\mathbf{u}} + \rho^s \int_{\Omega_n^s} \left((\mathbf{u} - \mathbf{w}) \cdot \nabla\right) \mathbf{u} \cdot \hat{\mathbf{u}} \\
&- \int_{\Omega_n^f} \left(\nabla \cdot \boldsymbol{\sigma}^f\right) \cdot \hat{\mathbf{u}} - \int_{\Omega_n^s} \left(\nabla \cdot \boldsymbol{\sigma}^s\right) \cdot \hat{\mathbf{u}} \\
&- \int_{\Omega} \hat{p} \nabla \cdot \mathbf{u} - \int_{\Omega_n^s} \mathbf{f} \cdot \hat{\mathbf{u}} + \int_{\Gamma_n} \left(\boldsymbol{\sigma}^s - \boldsymbol{\sigma}^f\right) \mathbf{n}^s \cdot \hat{\mathbf{u}}.
\end{aligned}
\tag{13}
$$

We integrate stress term by part and the last term in (13) would be cancelled out thanks to the interface condition (7). We also rearrange all the integrals such that the integrations only exist in the whole domain $\Omega$ and the solid domain $\Omega_n^s$. In this case, we shall use an Eulerian framework to describe the background fluid (including the fictitious fluid covered by the solid domain) on $\Omega$ and an updated Lagrangian framework to describe the solid on $\Omega_n^s$, i.e.: $\mathbf{w} = \mathbf{0}$ on $\Omega$ and $\mathbf{w} = \mathbf{u}$ on $\Omega_n^s$. Substituting the constitutive equations (8) and (9) into (13) and discretising the solid displacement $\mathbf{d} = \mathbf{d}_n + \Delta t \mathbf{u}$, equation (13) can then be expressed as:

$$
\begin{aligned}
&L\left(\mathbf{u}, p, \hat{\mathbf{u}}, \hat{p}, \mathbf{f}\right) = J(\mathbf{u}, \mathbf{f}) + \frac{\lambda}{u_c^2 - \|\mathbf{u}\|_{\Omega_n^s}^2} \\
&+\rho^f \int_{\Omega} \frac{\mathbf{u} - \mathbf{u}_n}{\Delta t} \cdot \hat{\mathbf{u}} + \left(\rho^s - \rho^f\right) \int_{\Omega_n^s} \frac{\mathbf{u} - \mathbf{u}_n}{\Delta t} \cdot \hat{\mathbf{u}} \\
&+\frac{\mu^f}{2} \int_{\Omega} \mathrm{D}\mathbf{u} : \mathrm{D}\hat{\mathbf{u}} + \frac{\Delta t c_1 - \mu^f}{2} \int_{\Omega_n^s} \mathrm{D}\mathbf{u} : \mathrm{D}\hat{\mathbf{u}} + \frac{c_1}{2} \int_{\Omega_n^s} \mathrm{D}\mathbf{d}_n : \mathrm{D}\hat{\mathbf{u}} \\
&+\rho^f \int_{\Omega} \left(\mathbf{u} \cdot \nabla\right) \mathbf{u} \cdot \hat{\mathbf{u}} - \int_{\Omega} p \nabla \cdot \hat{\mathbf{u}} - \int_{\Omega} \hat{p} \nabla \cdot \mathbf{u} - \int_{\Omega_n^s} \mathbf{f} \cdot \hat{\mathbf{u}} \\
&-c_1 \Delta t \int_{\Omega_n^s} \left(\nabla^T \mathbf{u} \nabla \mathbf{d}_n + \nabla^T \mathbf{d}_n \nabla \mathbf{u}\right) : \nabla \hat{\mathbf{u}} - c_1 \int_{\Omega_n^s} \nabla^T \mathbf{d}_n \nabla \mathbf{d}_n : \nabla \hat{\mathbf{u}}.
\end{aligned}
\tag{14}
$$

120 The following Karush-Kuhn-Tucker (KKT) conditions are the first-order necessary conditions in order to minimise (14):

$$
\frac{\partial L\left(\mathbf{u}, p, \mathbf{f}, \hat{\mathbf{u}}, \hat{p}\right)}{\partial\left(\hat{\mathbf{u}}, \hat{p}\right)} \left[\delta\hat{\mathbf{u}}, \delta\hat{p}\right] = 0, \tag{15}
$$

$$
\frac{\partial L\left(\mathbf{u}, p, \mathbf{f}, \hat{\mathbf{u}}, \hat{p}\right)}{\partial\left(\mathbf{u}, p\right)} \left[\delta\mathbf{u}, \delta p\right] = 0, \tag{16}
$$

$$
\frac{\partial L\left(\mathbf{u}, p, \mathbf{f}, \hat{\mathbf{u}}, \hat{p}\right)}{\partial \mathbf{f}} \left[\delta\mathbf{f}\right] = 0, \tag{17}
$$

where

$$\frac{\partial L(\cdot)}{\partial \mathbf{q}}[\delta\mathbf{q}] = \frac{d}{d\epsilon} L\left(\mathbf{q} + \epsilon\delta\mathbf{q}\right)\Big|_{\epsilon=0} \tag{18}$$

is the Gâteaux derivative with respective to variable $\mathbf{q}$ along the direction $\delta\mathbf{q}$ [2, 78].

*4.1. Primal equation*

The optimality condition (15) gives us the primal equation in a weak form as follows. Given $\mathbf{u}_n$ and $\mathbf{d}_n$, find $\mathbf{u} \in H^1_{\hat{\mathbf{u}}(\Gamma_D)}(\Omega)^d$ and $p \in L^2_0(\Omega)$, such that $\forall\delta\hat{\mathbf{u}} \in H^1_{0(\Gamma_D)}(\Omega)^d$ and $\forall\delta\hat{p} \in L^2(\Omega)$:

$$\begin{aligned}
&\rho^f \int_\Omega \frac{\mathbf{u}-\mathbf{u}_n}{\Delta t}\cdot\delta\hat{\mathbf{u}} + (\rho^s-\rho^f)\int_{\Omega_n^s}\frac{\mathbf{u}-\mathbf{u}_n}{\Delta t}\cdot\delta\hat{\mathbf{u}} \\
&+\frac{\mu^f}{2}\int_\Omega \mathrm{D}\mathbf{u}:\mathrm{D}\delta\hat{\mathbf{u}} + \frac{\Delta t c_1-\mu^f}{2}\int_{\Omega_n^s}\mathrm{D}\mathbf{u}:\mathrm{D}\delta\hat{\mathbf{u}} \\
&+\rho^f\int_\Omega (\mathbf{u}\cdot\nabla)\,\mathbf{u}\cdot\delta\hat{\mathbf{u}} - \int_\Omega p\nabla\cdot\delta\hat{\mathbf{u}} - \int_\Omega \delta\hat{p}\nabla\cdot\mathbf{u} \\
&-c_1\Delta t\int_{\Omega_n^s}\left(\nabla^T\mathbf{u}\nabla\mathbf{d}_n + \nabla^T\mathbf{d}_n\nabla\mathbf{u}\right):\nabla\delta\hat{\mathbf{u}} \\
&= \int_{\Omega_n^s}\mathbf{f}\cdot\delta\hat{\mathbf{u}} + c_1\int_{\Omega_n^s}\nabla^T\mathbf{d}_n\nabla\mathbf{d}_n:\nabla\delta\hat{\mathbf{u}} - \frac{c_1}{2}\int_{\Omega_n^s}\mathrm{D}\mathbf{d}_n:\mathrm{D}\delta\hat{\mathbf{u}}.
\end{aligned} \tag{19}$$

The solid domain is updated by $\Omega^s_{n+1} = \{\mathbf{x} : \mathbf{x} = \mathbf{x}_n + \Delta t\mathbf{u}, \forall\mathbf{x}_n \in \Omega^s_n\}$ after solving the above primal equation.

*4.2. Adjoint equation*

The optimality condition (16) gives us the adjoint equation in a weak form as follows. Given $\mathbf{u}$ and $\mathbf{d}_n$, find $\hat{\mathbf{u}} \in H^1_{0(\Gamma_D)}(\Omega)^d$ and $\hat{p} \in L^2_0(\Omega)$, such that $\forall\delta\mathbf{u} \in H^1_{0(\Gamma_D)}(\Omega)^d$ and $\forall\delta p \in L^2(\Omega)$:

$$\begin{aligned}
&\Delta t\int_{\Omega_n^s}(\mathbf{d}-\mathbf{d}_g)\cdot\delta\mathbf{u} + 2\lambda\int_{\Omega_n^s}\mathbf{u}\cdot\delta\mathbf{u}/\left(\|\mathbf{u}\|^2_{\Omega_n^s} - u_g^2(t_{n+1})\right)^2 \\
&+\frac{\rho^f}{\Delta t}\int_\Omega \delta\mathbf{u}\cdot\hat{\mathbf{u}} + \frac{\rho^s-\rho^f}{\Delta t}\int_{\Omega_n^s}\delta\mathbf{u}\cdot\hat{\mathbf{u}} \\
&+\frac{\mu^f}{2}\int_\Omega \mathrm{D}\delta\mathbf{u}:\mathrm{D}\hat{\mathbf{u}} + \frac{\Delta t c_1-\mu^f}{2}\int_{\Omega_n^s}\mathrm{D}\delta\mathbf{u}:\mathrm{D}\hat{\mathbf{u}} \\
&+\rho^f\int_\Omega (\delta\mathbf{u}\cdot\nabla)\,\mathbf{u}\cdot\hat{\mathbf{u}} + \rho^f\int_\Omega (\mathbf{u}\cdot\nabla)\,\delta\mathbf{u}\cdot\hat{\mathbf{u}} \\
&-\int_\Omega \delta p\nabla\cdot\hat{\mathbf{u}} - \int_\Omega\hat{p}\nabla\cdot\delta\mathbf{u} \\
&-c_1\Delta t\int_{\Omega_n^s}\left(\nabla^T\delta\mathbf{u}\nabla\mathbf{d}_n + \nabla^T\mathbf{d}_n\nabla\delta\mathbf{u}\right):\nabla\hat{\mathbf{u}} = 0.
\end{aligned} \tag{20}$$

In the above, the first order variation of the displacement $\mathbf{d}$ is approximated as $\delta\mathbf{d} = \Delta t\delta\mathbf{u}$.

*4.3. Optimality equation*

The optimality condition (17) gives the relation between the control force and adjoint variable:

$$\alpha \int_{\Omega_n^s} \delta\mathbf{f} \cdot \mathbf{f} = \int_{\Omega_n^s} \delta\mathbf{f} \cdot \hat{\mathbf{u}}. \tag{21}$$

## 5. A monolithic scheme

Substituting the optimality condition (21), specifically its strong form $\mathbf{f} = \frac{\hat{\mathbf{u}}}{\alpha}$, into equation (19), we have a monolithic scheme:

**Problem 2** (monolithic formulation for FSI control)**.** *Given* $\mathbf{u}_n$ *and* $\mathbf{d}_n$*, find* $\mathbf{u} \in H^1_{\hat{\mathbf{u}}(\Gamma_D)}(\Omega)^d$*,* $\hat{\mathbf{u}} \in H^1_{0(\Gamma_D)}(\Omega)^d$ *and* $p$*,* $\hat{p} \in L^2_0(\Omega)$*, such that* $\forall\delta\mathbf{u}$*,* $\delta\hat{\mathbf{u}} \in H^1_0(\Omega)^d$ *and* $\forall\delta p$*,* $\delta\hat{p} \in L^2(\Omega)$*:*

$$\begin{aligned}
&\frac{\rho^f}{\Delta t}\int_\Omega (\mathbf{u}\cdot\delta\hat{\mathbf{u}} + \delta\mathbf{u}\cdot\hat{\mathbf{u}}) + \frac{\rho^s-\rho^f}{\Delta t}\int_{\Omega_n^s}(\mathbf{u}\cdot\delta\hat{\mathbf{u}} + \delta\mathbf{u}\cdot\hat{\mathbf{u}})\\
+&\frac{\mu^f}{2}\int_\Omega (\mathrm{D}\mathbf{u}:\mathrm{D}\delta\hat{\mathbf{u}} + \mathrm{D}\delta\mathbf{u}:\mathrm{D}\hat{\mathbf{u}}) + \frac{\Delta t c_1-\mu^f}{2}\int_{\Omega_n^s}(\mathrm{D}\mathbf{u}:\mathrm{D}\delta\hat{\mathbf{u}} + \mathrm{D}\delta\mathbf{u}:\mathrm{D}\hat{\mathbf{u}})\\
+&\rho^f\int_\Omega [(\mathbf{u}\cdot\nabla)\,\mathbf{u}\cdot\delta\hat{\mathbf{u}} + (\delta\mathbf{u}\cdot\nabla)\,\mathbf{u}\cdot\hat{\mathbf{u}} + (\mathbf{u}\cdot\nabla)\,\delta\mathbf{u}\cdot\hat{\mathbf{u}}]\\
-&\int_\Omega p\nabla\cdot\delta\hat{\mathbf{u}} - \int_\Omega \delta\hat{p}\nabla\cdot\mathbf{u} - \int_\Omega \delta p\nabla\cdot\hat{\mathbf{u}} - \int_\Omega \hat{p}\nabla\cdot\delta\mathbf{u}\\
-&c_1\Delta t\int_{\Omega_n^s}\left[\left(\nabla^T\mathbf{u}\nabla\mathbf{d}_n + \nabla^T\mathbf{d}_n\nabla\mathbf{u}\right):\nabla\delta\hat{\mathbf{u}} + \left(\nabla^T\delta\mathbf{u}\nabla\mathbf{d}_n + \nabla^T\mathbf{d}_n\nabla\delta\mathbf{u}\right):\nabla\hat{\mathbf{u}}\right]\\
-&\frac{1}{\alpha}\int_{\Omega_n^s}\hat{\mathbf{u}}\cdot\delta\hat{\mathbf{u}} + 2\lambda\int_{\Omega_n^s}\mathbf{u}\cdot\delta\mathbf{u}/\left(\|\mathbf{u}\|^2_{\Omega_n^s} - u_g^2(t_{n+1})\right)^2\\
=&\frac{\rho^f}{\Delta t}\int_\Omega \mathbf{u}_n\cdot\delta\hat{\mathbf{u}} + \frac{\rho^s-\rho^f}{\Delta t}\int_{\Omega_n^s}\mathbf{u}_n\cdot\delta\hat{\mathbf{u}}\\
+&c_1\int_{\Omega_n^s}\nabla^T\mathbf{d}_n\nabla\mathbf{d}_n:\nabla\delta\hat{\mathbf{u}} - \frac{c_1}{2}\int_{\Omega_n^s}\mathrm{D}\mathbf{d}_n:\mathrm{D}\delta\hat{\mathbf{u}} - \Delta t\int_{\Omega_n^s}(\mathbf{d}_n-\mathbf{d}_g)\cdot\delta\mathbf{u}.
\end{aligned} \tag{22}$$

We use the mixed finite elements $(P_2, P_2, P_1, P_1)$ to disretise space $(H^1, H^1, L^2, L^2)$ of the solution pair $\mathbf{z} = (\mathbf{u}, \hat{\mathbf{u}}, p, \hat{p})$. Based upon the fictitious domain method, an Eulerian mesh is used to discretise the integrations in the augmented fluid domain $\Omega$, and an updated Lagrangian mesh to discretise the integrations in the moving solid domain $\Omega_n^s$. We then have the following linear equation system after space discretisation:

$$\left(\mathbf{K} + \mathbf{P}^T\mathbf{K}^s\mathbf{P}\right)\mathbf{z} = \mathbf{g} + \mathbf{P}^T\mathbf{g}^s, \tag{23}$$

where $\mathbf{K}$ and $\mathbf{K}^s$ are the system matrices from discretisation of the integrations, on the left-hand side of equation (22), in domain $\Omega$ and $\Omega_n^s$ respectively, and $\mathbf{g}$ and $\mathbf{g}^s$ are vectors from discretisation of the integrations, on the right-hand side of equation (22), in domain

$\Omega$ and $\Omega_n^s$ respectively. $\mathbf{P}$ is the finite element interpolation matrix from the background mesh and to solid mesh. Notice that the proposed monolithic scheme has similar features with our previous one-field monolithic fictitious domain method for FSI problems [16]. In this paper, we develop the previous monolithic scheme to include both the state variables $(\mathbf{u}, p)$ and the adjoint variables $(\hat{\mathbf{u}}, \hat{p})$ in order to solve FSI control problems with large solid deformation.

In the rest of this section, we present a reduced version the above monolithic formulation in order to solve a pure flow control problem: a monolithic method for velocity-tracking type of flow control by a body force in $\Omega$. This can be achieved by first, replacing the last term in (22) by a velocity objective: $\int_\Omega (\mathbf{u} - \mathbf{u}_g) \cdot \delta\mathbf{u}$; second, changing the domain of integration of term $\frac{1}{\alpha}\int_{\Omega_n^s} \hat{\mathbf{u}} \cdot \delta\hat{\mathbf{u}}$ in (22) to $\Omega$, which is related to the control force; third, removing all the other integrations in the solid domain in equation (22) (correspondingly solid matrix $\mathbf{K}^s$ and vector $\mathbf{g}^s$ in (23)). Finally, we have the following monolithic formulation for a flow control problem.

**Problem 3** (monolithic formulation for flow control)**.** *Given* $\mathbf{u}_n$*, find* $\mathbf{u} \in H^1_{\bar{\mathbf{u}}(\Gamma_D)}(\Omega)^d$*,* $\hat{\mathbf{u}} \in H^1_{0(\Gamma_D)}(\Omega)^d$ *and* $p$*,* $\hat{p} \in L^2_0(\Omega)$*, such that* $\forall\delta\mathbf{u}$*,* $\delta\hat{\mathbf{u}} \in H^1_0(\Omega)^d$ *and* $\forall\delta p$*,* $\delta\hat{p} \in L^2(\Omega)$*:*

$$
\begin{aligned}
&\frac{\rho^f}{\Delta t}\int_\Omega (\mathbf{u}\cdot\delta\hat{\mathbf{u}} + \delta\mathbf{u}\cdot\hat{\mathbf{u}}) + \frac{\mu^f}{2}\int_\Omega (\mathrm{D}\mathbf{u} : \mathrm{D}\delta\hat{\mathbf{u}} + \mathrm{D}\delta\mathbf{u} : \mathrm{D}\hat{\mathbf{u}}) \\
&+\rho^f\int_\Omega [(\mathbf{u}\cdot\nabla)\,\mathbf{u}\cdot\delta\hat{\mathbf{u}} + (\delta\mathbf{u}\cdot\nabla)\,\mathbf{u}\cdot\hat{\mathbf{u}} + (\mathbf{u}\cdot\nabla)\,\delta\mathbf{u}\cdot\hat{\mathbf{u}}] \\
&-\int_\Omega p\nabla\cdot\delta\hat{\mathbf{u}} - \int_\Omega \delta\hat{p}\nabla\cdot\mathbf{u} - \int_\Omega \delta p\nabla\cdot\hat{\mathbf{u}} - \int_\Omega \hat{p}\nabla\cdot\delta\mathbf{u} \\
&-\frac{1}{\alpha}\int_\Omega \hat{\mathbf{u}}\cdot\delta\hat{\mathbf{u}} + \int_\Omega \mathbf{u}\cdot\delta\mathbf{u} = \frac{\rho^f}{\Delta t}\int_\Omega \mathbf{u}_n\cdot\delta\hat{\mathbf{u}} + \int_\Omega \mathbf{u}_g\cdot\delta\mathbf{u}.
\end{aligned}
\tag{24}
$$

## 6. Numerical experiments

In this section, we assess and validate the proposed method using three numerical tests implemented using FreeFEM++ [79]. We first validate the scheme using a flow control problem which is widely studied in literature. The second numerical test is a benchmark FSI problem whose controllability is studied by an ALE formulation in [80], and we will show that the proposed two-mesh method can reduce the objective as effectively as the ALE method. Our third numerical experiment involves controlling a large-deformed solid; this problem is widely studied as a forward FSI problem in literature, which however has not been considered as an inverse control problem up to now. We hope our result will provide a potential benchmark for other researches in the area of optimal FSI control in the future.

### *6.1. Cavity flow*

In this example, we solve the reduced version of the proposed monolithic scheme formulated in Problem 3, and we consider control of a dynamic cavity pure fluid flow: steering the velocity to be a complicated predefined velocity profile with some vortices by controlling a

distributed body force in the whole fluid domain, which was studied in [42–44, 53]. We shall demonstrate that the proposed monolithic scheme can efficiently and accurately tracking the fluid field for a long time, with many vortices being developed (previous publications studied the case of less vortices). The computational domain is a $[0,1] \times [0,1]$ unit square. A wall boundary condition is prescribed for all the four sides of the cavity, and the fluid with $\rho^f = 1$ and $\mu^f = 0.1$ is initially stationary. The goal velocity

$$\mathbf{u}_g(x,y,t) = \left( \frac{\partial}{\partial y}\Psi(x,y,t), -\frac{\partial}{\partial x}\Psi(x,y,t) \right), \tag{25}$$

is derived from the following stream function:

$$\Psi(x,y,t) = \psi(x,t)\psi(y,t) \tag{26}$$

with

$$\psi(s,t) = (1-s)^2 \left(1 - cos(4\pi st)\right), \quad s \in [0,1]. \tag{27}$$

To get an intuition of the flow field, we visualise the objective flow at different times in Figure 2, from which it can be seen that more and more vortices are developed as time evolves, and the magnitude of the velocity increase at the same time. These figures are plotted on a mesh of 2138 triangles with 1130 vertices and 4394 degrees of freedom. Using the same mesh to carry out the simulation, we find that the controlled flow field can almost duplicate the objective flow to a very high accuracy. A typical comparison is shown in Figure 3, from which we cannot distinguish the objective and controlled flow field by a naked eye – the $L^2$ error is less than 0.001.

Convergence of the objective function with respective to the regularisation parameter $\alpha$, using $\Delta t = 0.01$, is presented in Figure 4, from which it can be seen that the error between the state velocity and the objective velocity gradually increase as time involves. This is not surprising because the flow field becomes more complicated and the control is more difficult as time increases. However, the accuracy can be further improved by refining the mesh in order to capture more details of the vortices. The convergence of the control force is presented in Figure 5, from which it can be seen that the same force, which cannot improve the accuracy on a coarse mesh, does improve the accuracy on a finer mesh.

### *6.2. Oscillating leaflet in a fluid channel*

In this test, we consider a benchmark FSI problem of an oscillating leaflet attached to a cylinder [23, 80, 81], and our objective is to minimise the solid deflection through an activation force on the solid leaflet. The computational domain is a rectangle $(L \times H)$ with a cut hole of radius $r$ and centre $(c, c)$ as shown in Figure 6. The geometry parameters are: $L = 2.5$, $H = 0.41$, $l = 0.35$, $h = 0.02$, $c = 0.2$ and $r = 0.05$. The fluid and solid parameters are: $\rho^f = \rho^s = 10^3$, $\mu^f = 1$ and $c_1 = 2.0 \times 10^6$. The inlet flow is prescribed as:

$$\bar{u}_x = \frac{12y}{H^2}(H - y), \quad \bar{u}_y = 0. \tag{28}$$

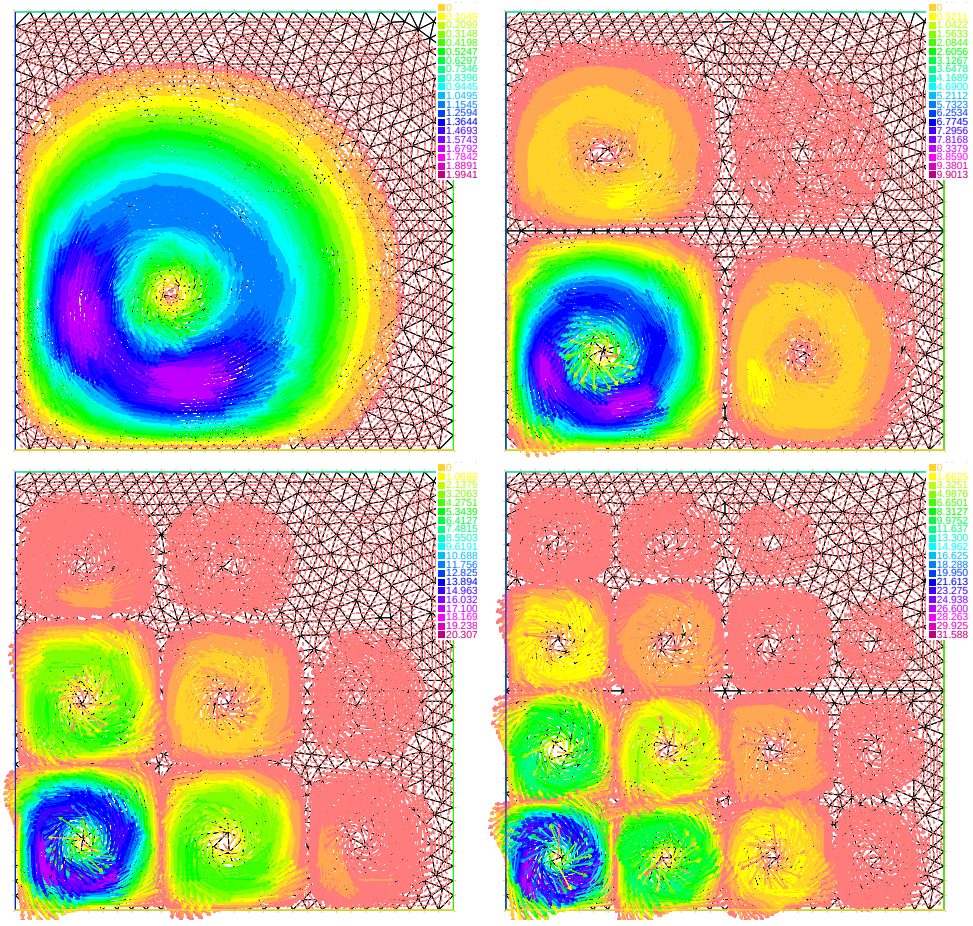


Figure 2: Velocity field at different times: $t = 0.5$, $t = 1$, $t = 1.5$ and $t = 2$ (from top to bottom and left to right).

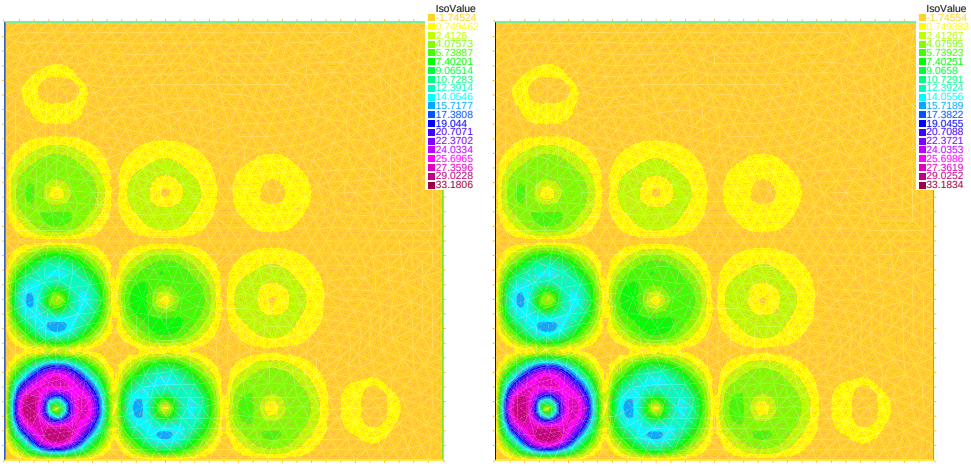



Figure 3: Velocity norm of the objective (left) and controlled (right) flow at $t = 2$ with $\alpha = 10^{-10}$; $\|\mathbf{u} - \mathbf{u}_g\|/\|\mathbf{u}_g\| < 10^{-3}$.

We use a mesh of 9008 elements with 4668 vertices for the background fluid (see Figure

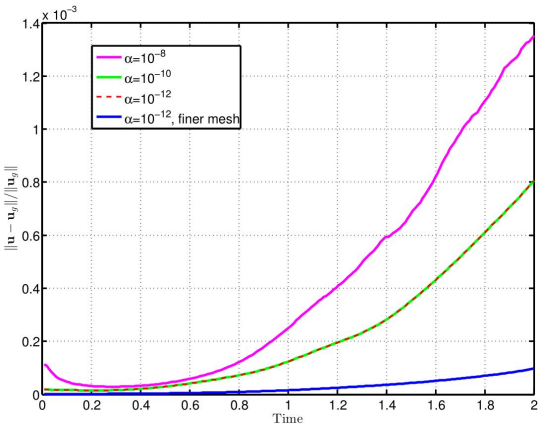



Figure 4: Convergence of the objective.

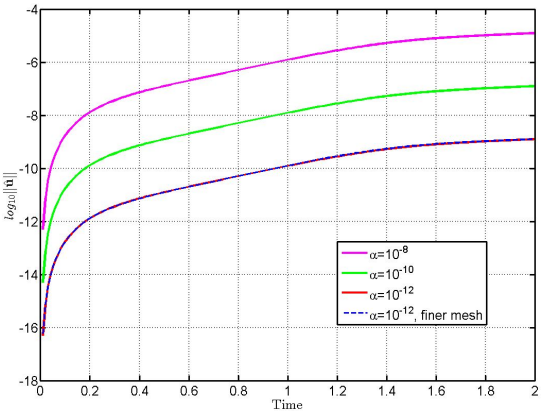



Figure 5: $L^2$ norm of the adjoint velocity $\hat{\mathbf{u}} = \alpha \mathbf{f}(t)$.

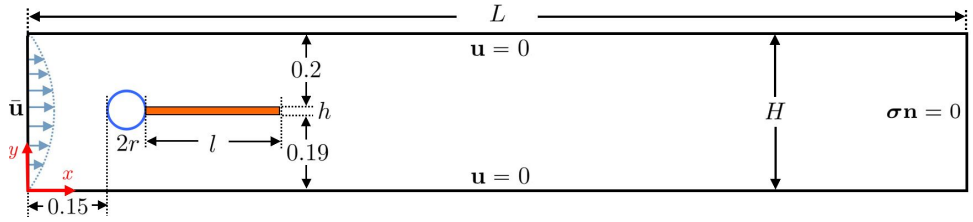



Figure 6: Computational domain and boundary conditions for the oscillating leaflet.

7), and a mesh of 314 elements with 213 vertices for the solid leaflet (see Figure 8). A converged time step of $\Delta t = 10^{-3}$ is used for this test, and our two-mesh method presents the same accuracy as the fitted-mesh method [23] with an oscillation period and amplitude being 0.530 and 0.03 respectively as shown in Figure 9 (red curve). We then focus on the control of this FSI system, and start to add an activation force $\mathbf{f}$ on the solid leaflet from $t = 3$ by solving the Problem 2 ($\lambda = 0$, constraint (12) is turned off). The overall control is tractable, and the vertical displacement at the tip of the leaflet is plotted in Figure 9 (dashed blue curve) from which it can be seen that the deflection of the leaflet is reduced around 50%. It is interesting to notice that the frequency of the leaflet's oscillation increases as its amplitude decreases after the control. The magnitude of the corresponding activation force is plotted in Figure 10, from which it can be seen that a large control force is computed at the very beginning of the control at $t = 3$, and then it decreases rapidly and responses periodically to the oscillation of the leaflet and keeps its deflection down. We also study the effect the regularisation parameter $\alpha$ on the reduction of the objective as shown in Figure 11 and 12. It is clear that smaller $\alpha$ can reduce the objective more, but it could also introduce more instability to the algorithm as shown for the case of $\alpha = 10^{-18}$ in Figure 11 and 12 (green curve).

### *6.3. Solid disc within a lid-driven cavity flow*

This FSI problem is considered in many publications [16, 82–84] as a forward FSI benchmark problem, whose controllability however has not be studied due to the complex move-

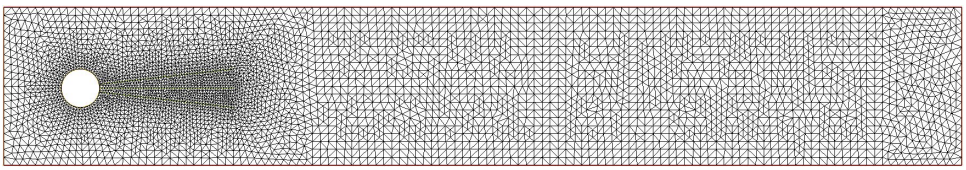


Figure 7: Background fluid mesh.

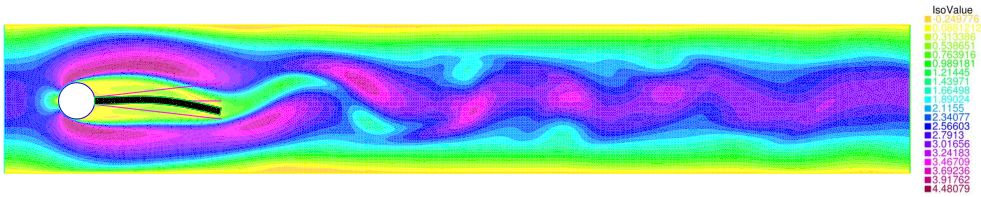



Figure 8: A snap shot of the velocity norms at $t = 4$ when the leaflet is maximally deformed.

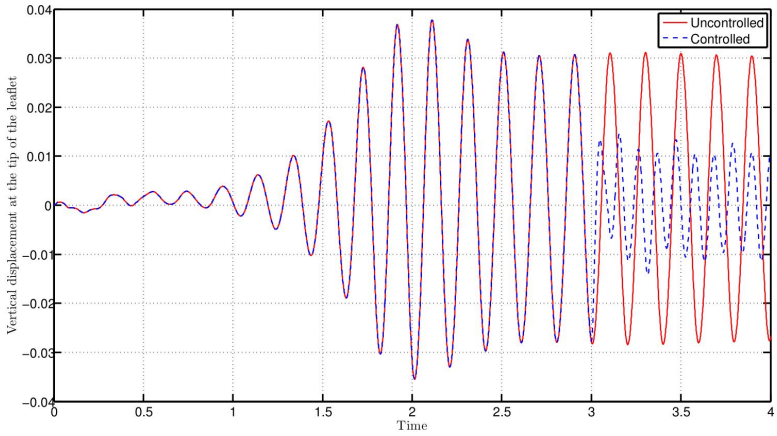



Figure 9: Vertical displacement at the tip of the leaflet. $\alpha = 10^{-17}$ for the controlled case.

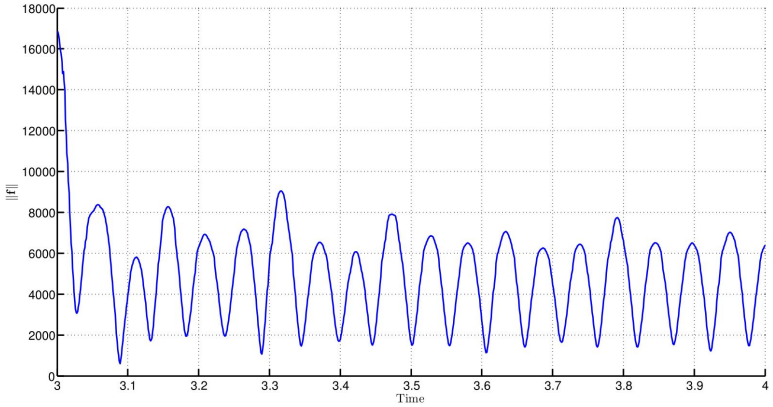



Figure 10: $L^2$ norm of the control force $\mathbf{f}(t)$, $\alpha = 10^{-17}$.

ment and large deformation of the solid disc. The computational domain is a unit square $[0, 1] \times [0, 1]$ and a solid disc of radius $r = 0.2$ is initially located at $(x_0, y_0) = (0.6, 0.5)$ as

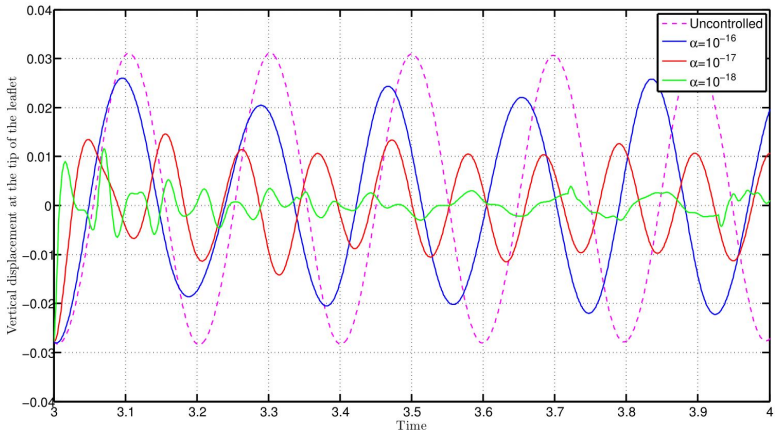



Figure 11: Vertical displacement at the tip of the leaflet for different regularisation parameters.

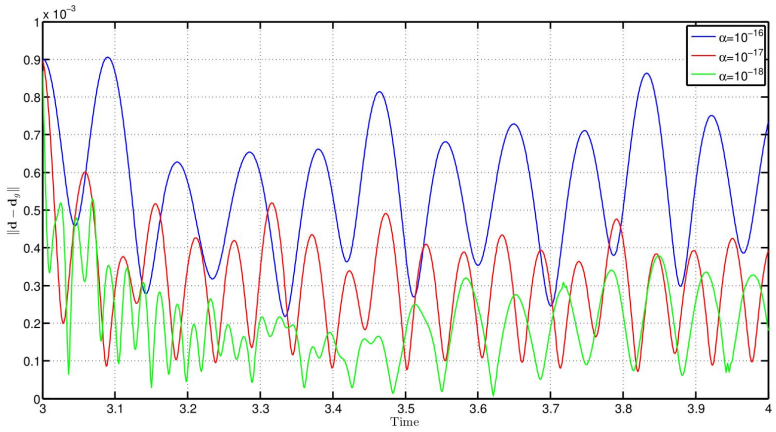



Figure 12: Objective redcution for different regularisation parameters.

shown in Figure 13. The fluid and solid material parameters are: $\rho^f = \rho^s = 1$, $\mu^f = 0.01$ and $c_1 = 1$. Due to drag prescribed at the top of the cavity, the solid gradually moves and rotates inside the cavity. We use a stable time step of $\Delta t = 0.005$, background fluid mesh of 2404 triangles with 1273 vertices, and solid mesh of 642 triangles with 352 vertices.

We first consider a case of pulling/pushing the solid back (by a distributed body force on the solid as shown in Problem 1) to the original position at different times, i.e., we solve Problem 2 with $\mathbf{d}_g = \mathbf{0}$ and penalty parameter $\lambda = 0$ (constraint (12) is turned off). Figure 14 shows the solid disc at different stages without control and Figures 15 shows that the proposed control method can successfully pull the solid back to the original position at different control times. We test effect of the regularisation parameter $\alpha$ on the control results as shown in Figure 16, from which it can be seen that larger $\alpha$ would not reduce the objective sufficiently, and smaller $\alpha$ can reduce the objective more while it also introduces slight oscillations for both the objective function and the control force.

In the above control, we have no control of the velocity of the solid disc by setting $\lambda = 0$. We now start to control the movement of the solid at $t = 4$ using $\alpha = 5 \times 10^{-7}$ and $u_c = 0.08$

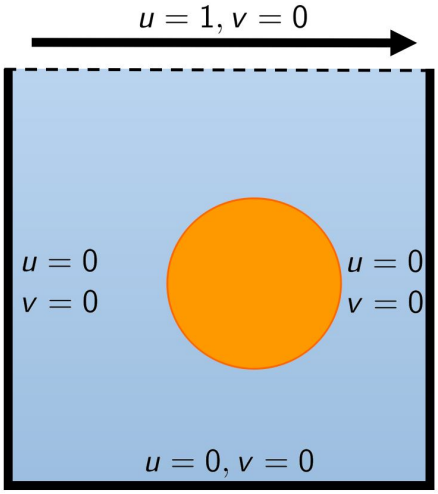



Figure 13: Sketch of a solid disc within a lid-driven cavity flow.

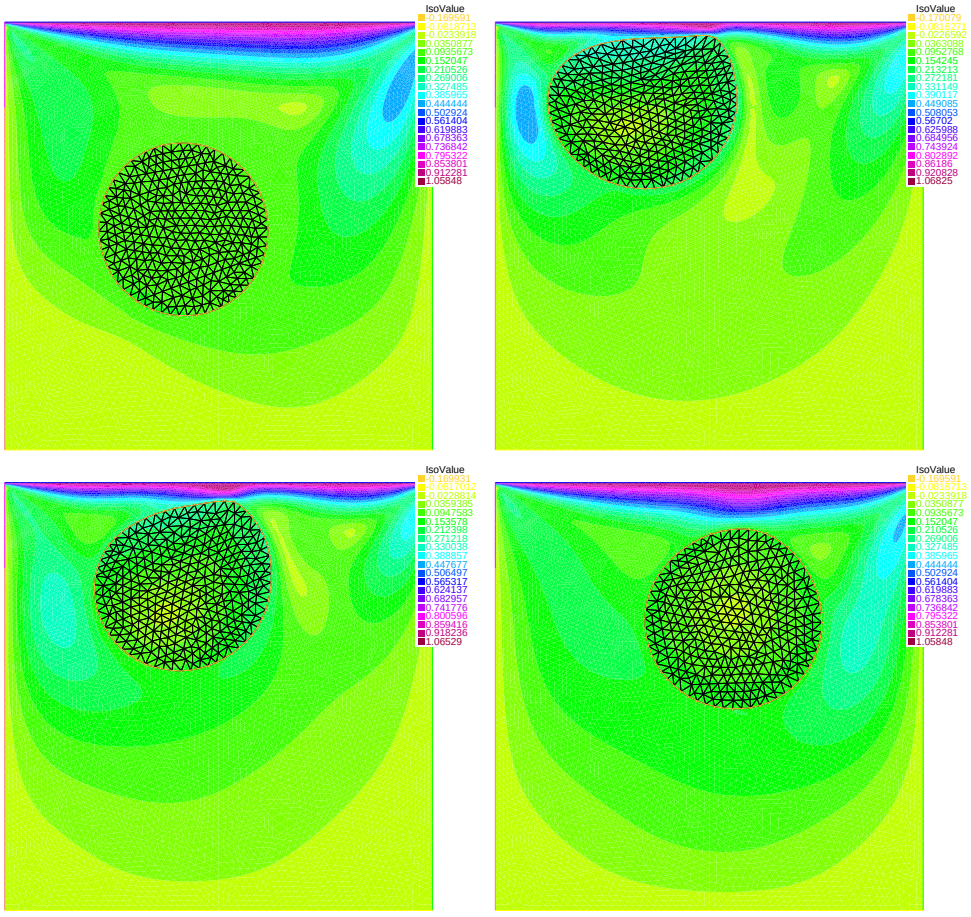



Figure 14: Velocity norm at different times: $t = 2$, $t = 5$, $t = 6$ and $t = 20$ (from top to bottom and left to right).

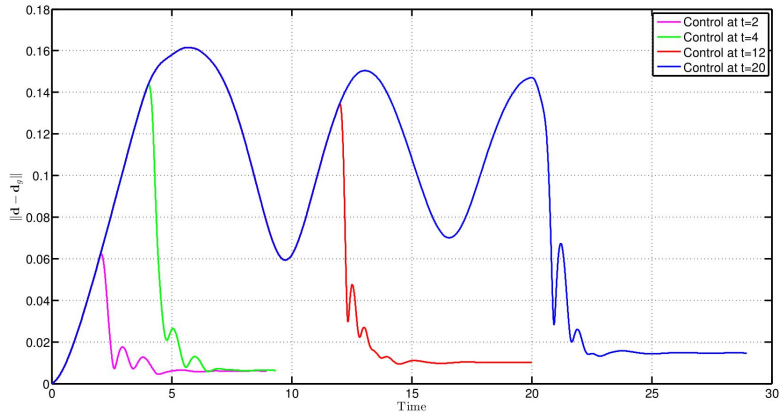
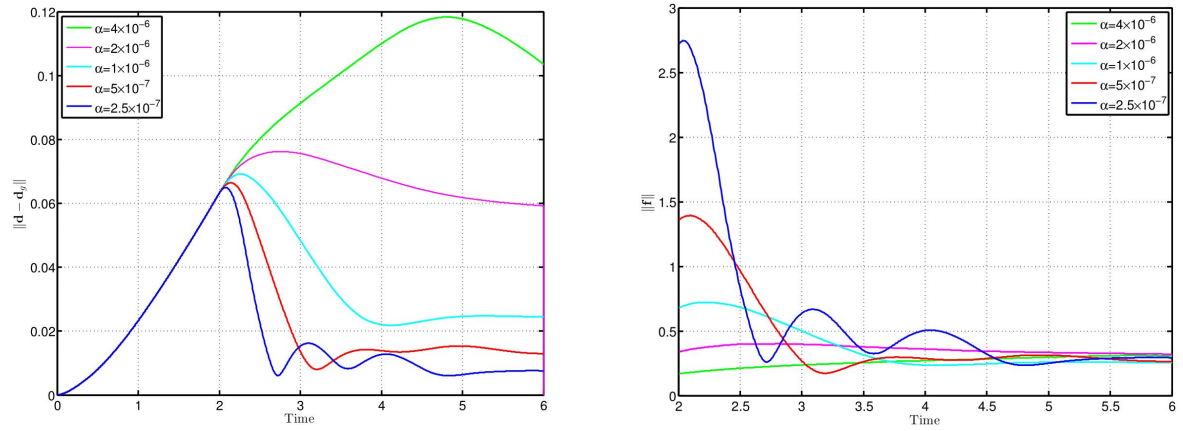



Figure 15: Reduction of the objective at different times using $\alpha = 2.5 \times 10^{-7}$.



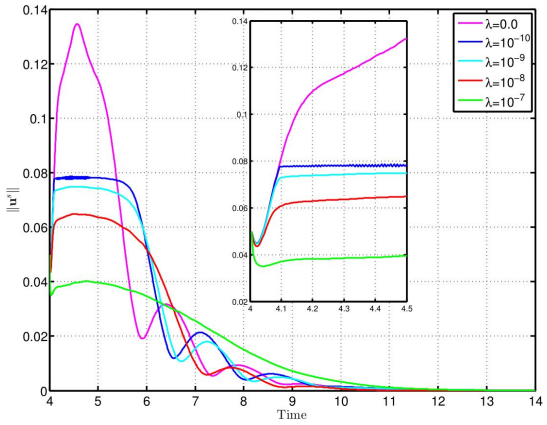

(a) Reduction of the objective using different regularisation parameter $\alpha$.

(b) Control force for different cases of regularisation parameter $\alpha$.

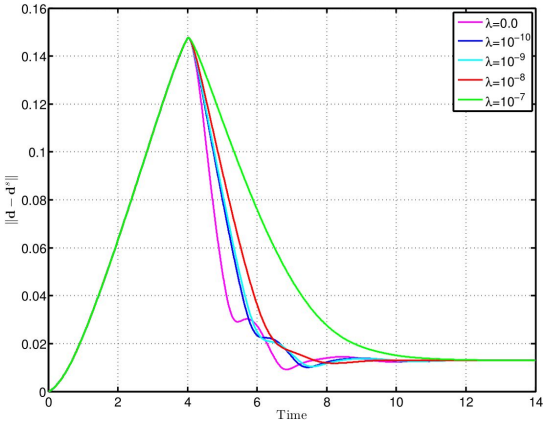
Figure 16: Apply the control from $t = 2$ using different regularisation parameter $\alpha$.

in problem 1, and investigate the speed of the solid body. It can be seen from Figure 17 (a) that the solid speed can be reduced bellow the predefined upper bound by choosing a reasonable penalty parameter $\lambda$. Notice that the solid speed would not approach to the case of $\lambda = 0$ (i.e.: constraint (12) is inactive) as $\lambda \to 0$, instead, it approaches to the equality scenario of constraint of (12). We also notice that too small $\lambda$ would cause instability issue as can be observed from the blue curve in Figure 17 (a). Therefore, a reasonable penalty parameter $\lambda$ should be used in order to control the speed of the solid disc. All these control of the solid speed does not have a significant influence of the reduction of the real objective as shown in Figure 17 (b).

Without any control, the movement of the solid is dominated by the surrounding fluid, which ends up rotating near the top of the cavity as shown in Figure 14. For this numerical test, we consider another challenging control of the solid: computing an appropriate force to hold the solid disc at its initial position $(x_0, y_0)$ and push it to rotate there without moving




(a) Velocity norm of the solid disc for different penalty $\lambda$ and objectives.




(b) Reduction of the objective for different penalty $\lambda$ and objectives.

Figure 17: Reducing the solid diplacement with a control of the solid's speed, starting from $t = 4$.

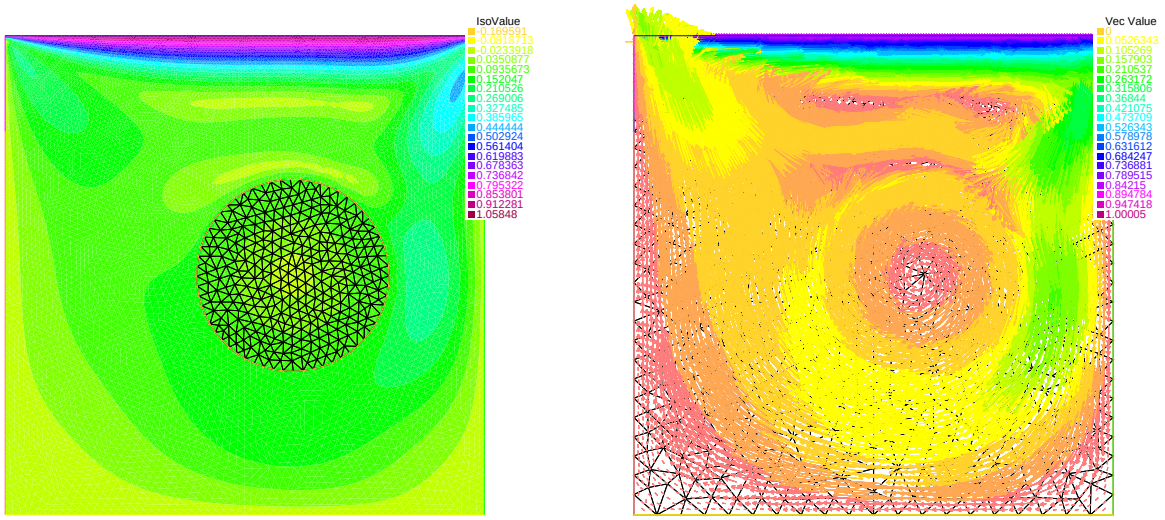
away. The objective displacement can be expressed as:

$$\begin{pmatrix} d_x \\ d_y \end{pmatrix} = \begin{bmatrix} cos\,(\omega t) & -sin\,(\omega t) \\ sin\,(\omega t) & cos\,(\omega t) \end{bmatrix} \begin{pmatrix} x - x_0 \\ y - y_0 \end{pmatrix},$$

with $\omega = -\pi/4$ being the angular velocity of the rotating disc we want to control. For this case, because the objective function is time dependent we find that a converged time step size is smaller: $\Delta t = 0.001$. Using this time step, we presents the controlled velocity field at $t = 2$ in Figure 18, from which it can be seen that the movement of the solid now dominates the cavity flow and a large vortex is created by the rotating disc. We also find that the solid disc gradually and slowly shifts away from its initial position using the previous coarse mesh as shown in Figure 19 (a), which presents the convergence of the objective function. However this shift becomes insignificant by using a finer mesh: 9618 triangles with 4950 vertices for the fluid, and 2570 triangles with 1346 vertices for the solid. We plot the $L^2$-norm of the control force in Figure 19 (b), from which it can be seen that the force dynamically responses to the error of the control and gradually approaches to a stable magnitude when the solid disc becomes tractable.

## 7. Conclusion

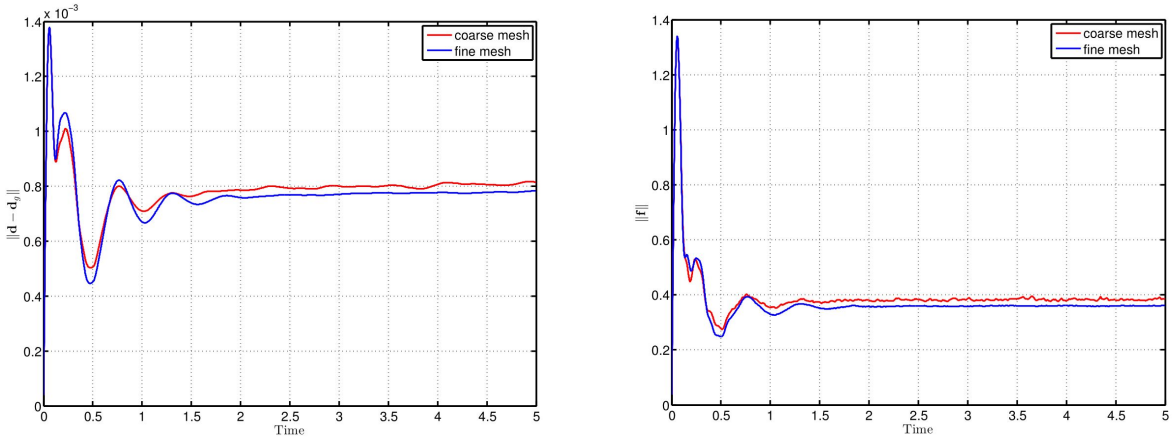
It is challenging to solve time-dependent FSI control problems with large solid deformations and very few examples have appeared in the literature. In this paper, we formulate a monolithic optimal control approach in the framework of piecewise-in-time control, which is stable for a range of regularisation parameters and efficient in reducing the displacement-tracking type of objective function; we consider an inequality constraint of the magnitude of the solid velocity, so that the solid speed can also be controlled when tracking its displacement; the proposed FSI control formulation, together with a reduced formulation for pure flow control problems, is first assessed by a typical flow control and a benchmark FSI




(a) Velocity norm and the solid mesh.

(b) Veocity field shown by arrows.

Figure 18: A control force holds and pushes the solid disc to rotate at its inital position; control at $t = 5$ using a coarse mesh.




(a) Objective function as a function of time.

(b) Control force as a function of time.

Figure 19: A control force holds and pushes the solid disc to rotate at its inital position.

problem, and then applied to a very challenging FSI control problem involving complicated movement and large deformation of the solid; all the numerical tests are implemented in open-source software package FreeFEM++ and shared via public Github site.

There are relevant topics which are interesting for future studies: it is widely accepted that piecewise-in-time control is effective in dealing with velocity-tracking type of objectives [43, 56] (now also displacement-tracking demonstrated in this paper), it is interesting to investigate other types of objective functions, such as reduction of drag force, based upon the proposed monolithic scheme; it is also worth investigating other types of control parameters, such as force distribution at the interface between the solid and surrounding fluids which

would be very useful for accurate design and control of biologically inspired robots, such as swimming robots [74] or micro medical robots [75].

This article focuses on methodological studies of optimal control for fluid-structure interaction problems. One application of the proposed method is to study the locomotion of biological worms. For example, the path of a C. elegans can be constructed based on experimental data [85, 86]; it is however difficult to understand how a worm contracts its muscles in order to follow the path. This is exactly a FSI control problem and can be modelled based upon the proposed method in the paper: the objective function is the worm's path/displacement and the control variable is its muscle force. We can then quantify the worm's muscle force and further understand its locomotion.

## Appendix A. Replication of results

The following FreeFEM code is an implementation of numerical test in Section 6.2. A complete FreeFEM code for all the numerical tests can be found in the Github repository: https://github.com/yongxingwang/.

```
// FSI control using two meshes
real muf=1,c1=2.e6,g=-0,rhof=1.e3,rhos=1.e3,rhod=rhos-rhof;
real dt=0.001, t, T0=3, Tc=4, mus=dt*c1-muf;
real x0=0.2, y0=0.2, r=0.05, L=2.5, H=0.41, h=0.02, l=0.35, theta=0.2013579208;
real x1=x0+r*cos(theta), y1=y0-h/2, x2=x1, y2=y0+h/2;
real alpha=1.e-17, xtip=0.6,ytip=y0,xotip=xtip,yotip=ytip;
int mh=3, m=20;
//fluid region
border a1(t=0,L) {x=t; y=0 ;label=1;};
border a2(t=0,H) {x=L; y=t ;label=2;};
border a3(t=L,0) {x=t; y=H ;label=1;};
border a4(t=H,0) {x=0; y=t ;label=3;};
//hole
border disc(t=0, 2*pi)  {x=x0+r*cos(t); y=y0+r*sin(t); label=4;};
//lines to refine mesh
border l0(t=0.25,0.6) {x=t; y=y0;label=5;};
border l1(t=x2,0.6) {x=t; y=0.04*(x-x2)/l+y2;  label=5;};
border l2(t=x1,0.6) {x=t; y=-0.04*(x-x1)/l+y1; label=5;};
mesh Th = buildmesh(a1(L*m/H)+a2(m)+a3(L*m/H)+a4(m)
+disc(-pi*mh/theta)+l0(l*mh/h)+l1(l*mh/h)+l2(l*mh/h));
//solid region
border b1(t=x1,0.6) {x=t; y=y1 ;label=5;};
border b2(t=y1,y2) {x=0.6; y=t ;label=5;};
border b3(t=0.6,x2) {x=t; y=y2 ;label=5;};
border b4(t=theta,-theta) {x=x0+r*cos(t); y=y0+r*sin(t);label=5;};
mesh Ths = buildmesh(b1(l*mh/h)+b2(mh)+b3(l*mh/h)+b4(mh));
```

```
plot(Th, Ths, wait=1);

mesh Thso=Ths, Ths0=Ths;
fespace Vh(Th,P2);
fespace Ph(Th,P1);
fespace Vhs(Ths,P2);
fespace Vhso(Thso,P2);
fespace Rh(Th,[P2,P2,P1]);
fespace RhAdj(Th,[P2,P2,P2,P2,P1,P1]);
fespace Rhs(Ths,[P2,P2]);
fespace RhsAdj(Ths,[P2,P2,P2,P2]);

Ph p,ph,phat,phath;
Vh u,v,uhat,vhat,uh,vh,uhath,vhath,uold=0,vold=0,uu;
Vhs us,vs,ushat=0,vshat=0,ush,vsh,ushath,vshath,usold=0,vsold=0,d1=0,d2=0,dg1=0,dg2=0;
Vhso uso,vso,usohat,vsohat,do1,do2;;

macro div(u,v) ( dx(u)+dy(v) ) // EOM
macro DD(u,v)  [[2*dx(u),div(v,u)],[div(v,u),2*dy(v)]] // EOM
macro Grad(u,v)[[dx(u),dy(u)],[dx(v),dy(v)]] // EOM

varf fluid([u,v,p],[uh,vh,ph]) =
int2d(Th)(rhof*[u,v]'*[uh,vh]/dt-div(uh,vh)*p-div(u,v)*ph
+ muf/2*trace(DD(u,v)'*DD(uh,vh)))
+ on(1,u=0, v=0) + on(3,u=12*y*(H-y)/H/H,v=0) + on(4,u=0,v=0);

varf resf([u,v,p],[uh,vh,ph]) =
int2d(Th)(g*rhof*vh+rhof*[convect([uold,vold],-dt,uold),
convect([uold,vold],-dt,vold)]'*[uh,vh]/dt )
+ on(1,u=0, v=0) + on(3,u=12*y*(H-y)/H/H,v=0) + on(4,u=0,v=0);

varf solid([us,vs],[ush,vsh]) =
int2d(Ths)( rhod*[us,vs]'*[ush,vsh]/dt
+mus/2*trace(DD(us,vs)'*DD(ush,vsh))
-dt*c1*trace((Grad(us,vs)'*Grad(d1,d2)+Grad(d1,d2)'*Grad(us,vs))*Grad(ush,vsh)')) ;

varf ress([us,vs],[ush,vsh]) =
int2d(Ths)( g*rhod*vsh-c1*trace((DD(d1,d2)-Grad(d1,d2)'*Grad(d1,d2))*Grad(ush,vsh)')
+ rhod*[usold,vsold]'*[ush,vsh]/dt );

matrix A = fluid(Rh,Rh);

ofstream file0("tip_disp.txt");
```

```
file0.precision(16);
for(t=dt;t<T0;t+=dt){
real[int] rhs1 = resf(0,Rh);
real[int] rhs2 = ress(0,Rhs);
matrix A = fluid(Rh,Rh);
matrix B = solid(Rhs,Rhs);
matrix P = interpolate(Rhs,Rh);
real[int] rhs = P'*rhs2;
rhs += rhs1;
matrix T = P'*B;
matrix AB = T*P;
AB+=A;

set(AB,solver=UMFPACK);
Rh [w1, w2, wp];
real[int] sol(Rh.ndof);
sol= w1[]; sol = AB^-1 * rhs;
w1[]=sol; u=w1; v= w2; p=wp;

Thso=Ths;
uso=u; vso=v; do1=d1; do2=d2;

xtip += uso(xotip,yotip)*dt; ytip += vso(xotip,yotip)*dt;
xotip=xtip;yotip=ytip;

Ths = movemesh(Ths, [x+us*dt, y+vs*dt]);

d1=0;  d1[]=do1[]+uso[]*dt;
d2=0;  d2[]=do2[]+vso[]*dt;
us=0;  us[]=uso[];
vs=0;  vs[]=vso[];

uold=u;vold=v; usold=us;vsold=vs;

uu=sqrt(u^2+v^2);
plot(uu,Ths,coef=0.1,fill=1,value=1,wait=0);

file0 <<d2(xtip,ytip)<< endl;
cout<<"NS Time: "<<t<<endl;
}

varf NSAdj([u,v,uhat,vhat,p,phat],[uh,vh,uhath,vhath,ph,phath]) =
int2d(Th)(rhof*[u,v]'*[uh,vh]/dt-div(uh,vh)*p-div(u,v)*ph
```

```
+rhof*[uhat,vhat]'*[uhath,vhath]/dt-div(uhath,vhath)*phat-div(uhat,vhat)*phath
+rhof*[uhath,vhath]'*(Grad(uold,vold)'*[uhat,vhat])
-rhof*[uhath,vhath]'*(Grad(uhat,vhat) *[uold,vold])
+muf/2*trace(DD(u,v)'*DD(uh,vh))
+muf/2*trace(DD(uhat,vhat)'*DD(uhath,vhath)))
+on(1,u=0,v=0,uhat=0,vhat=0) + on(3,u=12*y*(H-y)/H/H,v=0,uhat=0,vhat=0)
+on(4,u=0,v=0,uhat=0,vhat=0);

varf resNSAdj([u,v,uhat,vhat,p,phat],[uh,vh,uhath,vhath,ph,phath]) =
int2d(Th)(g*rhof*vh+rhof*[convect([uold,vold],-dt,uold),
convect([uold,vold],-dt,vold)]'*[uh,vh]/dt)
+on(1,u=0,v=0,uhat=0,vhat=0) + on(3,u=12*y*(H-y)/H/H,v=0,uhat=0,vhat=0)
+on(4,u=0,v=0,uhat=0,vhat=0);

varf solidAdj([us,vs,ushat,vshat],[ush,vsh,ushath,vshath]) =
int2d(Ths)(rhod*[us,vs]'*[ush,vsh]/dt
+rhod*[ushath,vshath]'*[ushat,vshat]/dt
+mus/2*trace(DD(us,vs)'*DD(ush,vsh))
+mus/2*trace(DD(ushat,vshat)'*DD(ushath,vshath))
-dt*c1*trace((Grad(us,vs)'*Grad(d1,d2)+Grad(d1,d2)'*Grad(us,vs))*Grad(ush,vsh)')
-dt*c1*trace((Grad(ushath,vshath)'*Grad(d1,d2)+Grad(d1,d2)'
Grad(ushath,vshath))*Grad(ushat,vshat)')
-[ushat,vshat]'*[ush,vsh]/alpha);

varf ressAdj([us,vs,ushat,vshat],[ush,vsh,ushath,vshath]) =
int2d(Ths)(g*rhod*vsh + c1*trace((Grad(d1,d2)'*Grad(d1,d2))*Grad(ush,vsh)')
-0.5*c1*trace(DD(d1,d2)*DD(ush,vsh)')
+rhod*[usold,vsold]'*[ush,vsh]/dt
-dt*[d1-dg1,d2-dg2]'*[ushath,vshath]);

matrix Aadj = NSAdj(RhAdj,RhAdj);

ofstream file1("objective_force.txt");
file1.precision(16);
for(t=T0;t<Tc;t+=dt){
real[int] rhs1 = resNSAdj(0,RhAdj);
matrix Badj = solidAdj(RhsAdj,RhsAdj);
real[int] rhs2 = ressAdj(0,RhsAdj);
matrix P = interpolate(RhsAdj,RhAdj);
real[int] rhs = P'*rhs2;
rhs += rhs1;
matrix T = P'*Badj;
matrix AB = T*P;
```

```
AB += Aadj;

set(AB,solver=UMFPACK);
RhAdj [w1, w2, s1, s2, wp, sp];
real[int] sol(RhAdj.ndof);
sol= w1[]; sol = AB^-1 * rhs; w1[]=sol;
u=w1; v=w2; uhat=s1; vhat= s2; p=wp; phat=sp;

Thso=Ths;
uso=u; vso=v; usohat=uhat; vsohat=vhat;
do1=d1; do2=d2;

xtip += uso(xotip,yotip)*dt; ytip += vso(xotip,yotip)*dt;
xotip=xtip;yotip=ytip;

Ths = movemesh(Ths, [x+us*dt, y+vs*dt]);

d1=0;  d1[]=do1[]+uso[]*dt;
d2=0;  d2[]=do2[]+vso[]*dt;
us=0;  us[]=uso[];
vs=0;  vs[]=vso[];
ushat=0; ushat[]=usohat[];
vshat=0; vshat[]=vsohat[];

uold=u;vold=v;usold=us;vsold=vs;

uu=sqrt(u^2+v^2);
plot(uu,Ths, coef=10,fill=1,value=1,wait=0);

real error=sqrt(int2d(Ths)((d1-dg1)^2+(d2-dg2)^2));
real force=sqrt(int2d(Ths)(ushat^2+vshat^2))/alpha;
file1 << error <<" "<< force << endl;
file0 <<d2(xtip,ytip)<< endl;
cout<<"Control Time: "<<t<<"  "<<error<<endl;
}
file0.flush;
file1.flush;
```

## References


[1] G. Morgenthal, Fluid–Structure Interaction in Bluff-body Aerodynamics and Long-span Bridge Design: Phenomena and Methods, University of Cambridge, Department of Engineering Cambridge, 2000.

[2] Y. Bazilevs, K. Takizawa, T. E. Tezduyar, Computational fluid-structure interaction: methods and applications, John Wiley & Sons, 2013.

[3] B. Mohammadi, O. Pironneau, Applied shape optimization for fluids, Oxford University Press, 2010.
[4] M. E. McCormick, Ocean engineering mechanics: with applications, Cambridge University Press, 2009.
[5] W. Bai, R. E. Taylor, Fully nonlinear simulation of wave interaction with fixed and floating flared structures, Ocean Engineering 36 (3) (2009) 223–236.
[6] W. Finnegan, J. Goggins, Numerical simulation of linear water waves and wave–structure interaction, Ocean Engineering 43 (2012) 23–31.
[7] S. Čanić, B. Muha, M. Bukač, Fluid–structure interaction in hemodynamics: modeling, analysis, and numerical simulation, in: Fluid-Structure Interaction and Biomedical Applications, Springer, 2014, pp. 79–195.
[8] S. Deparis, D. Forti, G. Grandperrin, A. Quarteroni, Facsi: A block parallel preconditioner for fluid–structure interaction in hemodynamics, Journal of Computational Physics 327 (2016) 700–718.
[9] F. Piatti, F. Sturla, G. Marom, J. Sheriff, T. E. Claiborne, M. J. Slepian, A. Redaelli, D. Bluestein, Hemodynamic and thrombogenic analysis of a trileaflet polymeric valve using a fluid–structure interaction approach, Journal of Biomechanics 48 (13) (2015) 3641–3649.
[10] U. Küttler, W. A. Wall, Fixed-point fluid–structure interaction solvers with dynamic relaxation, Computational Mechanics 43 (1) (2008) 61–72. doi:10.1007/s00466-008-0255-5.
[11] J. Degroote, K.-J. Bathe, J. Vierendeels, Performance of a new partitioned procedure versus a monolithic procedure in fluid–structure interaction, Computers & Structures 87 (11-12) (2009) 793–801. doi:10.1016/j.compstruc.2008.11.013.
[12] J. Degroote, M. Hojjat, E. Stavropoulou, R. Wüchner, K.-U. Bletzinger, Partitioned solution of an unsteady adjoint for strongly coupled fluid-structure interactions and application to parameter identification of a one-dimensional problem, Structural and Multidisciplinary Optimization 47 (1) (2013) 77–94.
[13] M. Heil, An efficient solver for the fully coupled solution of large-displacement fluid–structure interaction problems, Computer Methods in Applied Mechanics and Engineering 193 (1-2) (2004) 1–23. doi:10.1016/j.cma.2003.09.006.
[14] M. Heil, A. L. Hazel, J. Boyle, Solvers for large-displacement fluid–structure interaction problems: segregated versus monolithic approaches, Computational Mechanics 43 (1) (2008) 91–101. doi:10.1007/s00466-008-0270-6.
[15] R. L. Muddle, M. Mihajlović, M. Heil, An efficient preconditioner for monolithically-coupled large-displacement fluid–structure interaction problems with pseudo-solid mesh updates, Journal of Computational Physics 231 (21) (2012) 7315–7334. doi:10.1016/j.jcp.2012.07.001.
[16] Y. Wang, P. K. Jimack, M. A. Walkley, A one-field monolithic fictitious domain method for fluid–structure interactions, Computer Methods in Applied Mechanics and Engineering 317 (2017) 1146–1168. doi:10.1016/j.cma.2017.01.023.
[17] Y. Wang, P. K. Jimack, M. A. Walkley, O. Pironneau, An energy stable one-field monolithic arbitrary Lagrangian-Eulerian formulation for fluid-structure interaction, Journal of Fluids and Structures 98 (2020) 103117. doi:https://doi.org/10.1016/j.jfluidstructs.2020.103117.
[18] C. S. Peskin, The immersed boundary method, Acta Numerica 11 (2002) 479–517. doi:10.1016/j.cma.2015.12.023.
[19] L. Zhang, A. Gerstenberger, X. Wang, W. K. Liu, Immersed finite element method, Computer Methods in Applied Mechanics and Engineering 193 (21) (2004) 2051–2067. doi:doi:10.1016/j.cma.2003.12.044.
[20] F. P. Baaijens, A fictitious domain/mortar element method for fluid-structure interaction, International Journal for Numerical Methods in Fluids 35 (7) (2001) 743–761. doi:10.1002/1097-0363(20010415)35:7<743::AID-FLD109>3.0.CO;2-A.
[21] D. Boffi, L. Gastaldi, A fictitious domain approach with Lagrange multiplier for fluid-structure interactions, Numerische Mathematik 135 (3) (2016) 711–732. doi:10.1007/s00211-016-0814-1.
[22] D. Boffi, N. Cavallini, L. Gastaldi, The finite element immersed boundary method with distributed Lagrange multiplier, SIAM Journal on Numerical Analysis 53 (6) (2015) 2584–2604. doi:10.1137/140978399.

[23] F. Hecht, O. Pironneau, An energy stable monolithic Eulerian fluid-structure finite element method, International Journal for Numerical Methods in Fluids 85 (7) (2017) 430–446. doi:10.1002/fld.4388.

[24] T. Wick, Fully Eulerian fluid–structure interaction for time-dependent problems, Computer Methods in Applied Mechanics and Engineering 255 (2013) 14–26.

[25] T. Richter, T. Wick, Finite elements for fluid–structure interaction in ALE and fully Eulerian coordinates, Computer Methods in Applied Mechanics and Engineering 199 (41-44) (2010) 2633–2642. doi:10.1016/j.cma.2010.04.016.

[26] R. Rannacher, T. Richter, An adaptive finite element method for fluid-structure interaction problems based on a fully Eulerian formulation, in: Fluid–Structure Interaction II, Springer Berlin Heidelberg, 2010, pp. 159–191. doi:10.1007/978-3-642-14206-2_7.

[27] B. Schott, C. Ager, W. A. Wall, A monolithic approach to fluid-structure interaction based on a hybrid Eulerian-ALE fluid domain decomposition involving cut elements, International Journal for Numerical Methods in Engineering 119 (3) (2019) 208–237.

[28] F. Tröltzsch, Optimal control of partial differential equations: theory, methods, and applications, Vol. 112, American Mathematical Soc., 2010.

[29] O. Pironneau, On optimum design in fluid mechanics, Journal of Fluid Mechanics 64 (1) (1974) 97–110.

[30] R. Glowinski, O. Pironneau, On the numerical computation of the minimum-drag profile in laminar flow, Journal of Fluid Mechanics 72 (2) (1975) 385–389.

[31] T. D. Montenegro-Johnson, E. Lauga, The other optimal stokes drag profile, Journal of Fluid Mechanics 762 (2015) 1–11.

[32] A. Henrot, Y. Privat, What is the optimal shape of a pipe?, Archive for Rational Mechanics and Analysis 196 (1) (2010) 281–302.

[33] C. Dapogny, P. Frey, F. Omnès, Y. Privat, Geometrical shape optimization in fluid mechanics using freefem++, Structural and Multidisciplinary Optimization 58 (6) (2018) 2761–2788.

[34] N. Jenkins, K. Maute, An immersed boundary approach for shape and topology optimization of stationary fluid-structure interaction problems, Structural and Multidisciplinary Optimization 54 (5) (2016) 1191–1208.

[35] M. D. Gunzburger, H. Kim, S. Manservisi, On a shape control problem for the stationary navier-stokes equations, ESAIM: Mathematical Modelling and Numerical Analysis 34 (6) (2000) 1233–1258.

[36] H. Choi, P. Moin, J. Kim, et al., Active turbulence control for drag reduction in wall-bounded flows, Journal of Fluid Mechanics 262 (1994) 75–75.

[37] J. Kim, Physics and control of wall turbulence for drag reduction, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 369 (1940) (2011) 1396–1411.

[38] S. Jeon, J. Choi, W.-P. Jeon, H. Choi, J. Park, Active control of flow over a sphere for drag reduction at a subcritical reynolds number, Journal of Fluid Mechanics 517 (2004) 113.

[39] M. Dong, J. Liao, Z. Du, W. Huang, Influences of lateral jet location and its number on the drag reduction of a blunted body in supersonic flows, The Aeronautical Journal 124 (1277) (2020) 1055–1069.

[40] J. McNally, E. Fernandez, G. Robertson, R. Kumar, K. Taira, F. Alvi, Y. Yamaguchi, K. Murayama, Drag reduction on a flat-back ground vehicle with active flow control, Journal of Wind Engineering and Industrial Aerodynamics 145 (2015) 292–303.

[41] J.-L. Lions, Exact controllability, stabilization and perturbations for distributed systems, SIAM Review 30 (1) (1988) 1–68.

[42] L. Hou, Y. Yan, Dynamics and approximations of a velocity tracking problem for the navier–stokes flows with piecewise distributed controls, SIAM Journal on Control and Optimization 35 (6) (1997) 1847–1885.

[43] L. Hou, S. Ravindran, Y. Yan, Numerical solution of optimal distributed control problems for incompressible flows, International Journal of Computational Fluid Dynamics 8 (2) (1997) 99–114.

[44] M. Gunzburger, L. Hou, S. Manservisim, Y. Yan, Computations of optimal controls for incompressible flows, International Journal of Computational Fluid Dynamics 11 (1-2) (1998) 181–191.

[45] M. D. Gunzburger, Perspectives in flow control and optimization, SIAM, 2002.

[46] M. Pošta, T. Roubíček, Optimal control of navier-stokes equations by oseen approximation, Computers & Mathematics With Applications 53 (3-4) (2007) 569–581.
[47] A. Attavino, D. Cerroni, R. Da Vià, S. Manservisi, F. Menghini, Adjoint optimal control problems for the rans system, in: Journal of Physics: Conference Series, Vol. 796, IOP Publishing, 2017, p. 012008.
[48] S. Manservisi, F. Menghini, Numerical simulations of optimal control problems for the reynolds averaged navier–stokes system closed with a two-equation turbulence model, Computers & Fluids 125 (2016) 130–143.
[49] M. D. Gunzburger, S. Manservisi, Analysis and approximation of the velocity tracking problem for navier–stokes flows with distributed control, SIAM Journal on Numerical Analysis 37 (5) (2000) 1481–1512.
[50] S. Manservisi, F. Menghini, Optimal control problems for the navier–stokes system coupled with the k-$\omega$ turbulence model, Computers & Mathematics with Applications 71 (11) (2016) 2389–2406.
[51] M. Gunzburger, L. Hou, T. P. Svobodny, Analysis and finite element approximation of optimal control problems for the stationary navier-stokes equations with dirichlet controls, ESAIM: Mathematical Modelling and Numerical Analysis 25 (6) (1991) 711–748.
[52] H. Fattorini, S. Sritharan, Existence of optimal controls for viscous flow problems, Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences 439 (1905) (1992) 81–102.
[53] M. D. Gunzburger, S. Manservisi, The velocity tracking problem for navier–stokes flows with boundary control, SIAM Journal on Control and Optimization 39 (2) (2000) 594–634.
[54] A. Fursikov, M. D. Gunzburger, L. Hou, Optimal boundary control for the evolutionary navier–stokes system: the three-dimensional case, SIAM Journal on Control and Optimization 43 (6) (2005) 2191–2232.
[55] E. Aulisa, S. Manservisi, A multigrid approach to optimal control computations for navier-stokes flows, in: Robust Optimization-Directed Design, Springer, 2006, pp. 3–23.
[56] F. Abergel, R. Temam, On some control problems in fluid mechanics, Theoretical and Computational Fluid Dynamics 1 (6) (1990) 303–325.
[57] M. D. Gunzburger, Flow control, Vol. 68, Springer Science & Business Media, 2012.
[58] M. Moubachir, J.-P. Zolésio, Optimal control of fluid-structure interaction systems: the case of a rigid solid, Ph.D. thesis, INRIA (2002).
[59] M. Moubachir, J.-P. Zolesio, Moving shape analysis and control: applications to fluid–structure interactions, CRC Press, 2006.
[60] L. Bociu, J.-P. Zolésio, Sensitivity analysis for a free boundary fluid-elasticity interaction, Evolution Equations & Control Theory 2 (1) (2013) 55.
[61] I. Lasiecka, A. Tuffaha, Boundary feedback control in fluid-structure interactions, in: 2008 47th IEEE Conference on Decision and Control, IEEE, 2008, pp. 203–208.
[62] F. Bucci, I. Lasiecka, Optimal boundary control with critical penalization for a pde model of fluid–solid interactions, Calculus of Variations and Partial Differential Equations 37 (1) (2010) 217–235.
[63] I. Lasiecka, A. Tuffaha, Riccati theory and singular estimates for a bolza control problem arising in linearized fluid–structure interaction, Systems & Control Letters 58 (7) (2009) 499–509.
[64] T. Richter, T. Wick, Optimal control and parameter estimation for stationary fluid-structure interaction problems, SIAM Journal on Scientific Computing 35 (5) (2013) B1085–B1104.
[65] T. Wick, W. Wollner, Optimization with nonstationary, nonlinear monolithic fluid-structure interaction, International Journal for Numerical Methods in Engineering.
[66] G. Peralta, K. Kunisch, Analysis and finite element discretization for optimal control of a linear fluid–structure interaction problem with delay, IMA Journal of Numerical Analysis 40 (1) (2020) 140–206.
[67] L. Bociu, L. Castle, K. Martin, D. Toundykov, Optimal control in a free boundary fluid-elasticity interaction, in: Conference Publications, Vol. 2015, American Institute of Mathematical Sciences, 2015, p. 122.
[68] L. Failer, D. Meidner, B. Vexler, Optimal control of a linear unsteady fluid–structure interaction problem, Journal of Optimization Theory and Applications 170 (1) (2016) 1–27.
[69] L. Failer, T. Richter, A newton multigrid framework for optimal control of fluid–structure interactions,

Optimization and Engineering (2020) 1–29.
[70] A. Chierici, L. Chirco, R. Da Vià, M. Manservisi, S. Magnaniand, Distributed optimal control applied to fluid–structure interaction problems, in: Journal of Physics: Conference Series, Vol. 1224, IOP Publishing, 2019, p. 012003.
[71] L. Chirco, R. Da Vià, S. Manservisi, An optimal control method for fluid–structure interaction systems via adjoint boundary pressure, in: Journal of Physics: Conference Series, Vol. 923, IOP Publishing, 2017, p. 012026.
[72] L. Chirco, S. Manservisi, An adjoint based pressure boundary optimal control approach for fluid-structure interaction problems, Computers & Fluids 182 (2019) 118–127.
[73] L. Chirco, S. Manservisi, On the optimal control of stationary fluid–structure interaction systems, Fluids 5 (3) (2020) 144.
[74] A. Crespi, D. Lachat, A. Pasquier, A. J. Ijspeert, Controlling swimming and crawling in a fish robot using a central pattern generator, Autonomous Robots 25 (1) (2008) 3–13.
[75] J. Xiao, Q. Wu, D. Sun, C. He, Y. Chen, Classifications and functions of vitreoretinal surgery assisted robots-a review of the state of the art, in: 2019 International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS), IEEE, 2019, pp. 474–484.
[76] D. P. Bertsekas, Constrained optimization and Lagrange multiplier methods, Academic press, 2014.
[77] B. EL-Sobky, A. Aboutahoun, An active-set algorithm and a trust-region approach in constrained minimax problem, Computational and Applied Mathematics 37 (2018) 2605–2631.
[78] L. B. Rall, Nonlinear functional analysis and applications: proceedings of an advanced seminar conducted by the Mathematics Research Center, the University of Wisconsin, Madison, October 12-14, 1970, no. 26, Elsevier, 2014.
[79] F. Hecht, New development in FreeFem++, Journal of Numerical Mathematics 20 (3-4) (2012) 251–265.
[80] Y. Wang, P. K. Jimack, M. A. Walkley, D. Yang, H. M. Thompson, An optimal control method for time-dependent fluid-structure interaction problems, Structural and Multidisciplinary Optimization (2021) 1–24.
[81] S. Turek, J. Hron, Proposal for numerical benchmarking of fluid–structure interaction between an elastic object and laminar incompressible flow, in: Fluid-Structure Interaction, Springer, 2006, pp. 371–385.
[82] H. Zhao, J. B. Freund, R. D. Moser, A fixed-mesh method for incompressible flow–structure systems with finite solid deformations, Journal of Computational Physics 227 (6) (2008) 3114–3140. doi:10.1016/j.jcp.2007.11.019.
[83] Y. Wang, P. K. Jimack, M. A. Walkley, A theoretical and numerical investigation of a family of immersed finite element methods, Journal of Fluids and Structures 91 (2019) 102754.
[84] S. Roy, L. Heltai, F. Costanzo, Benchmarking the immersed finite element method for fluid–structure interaction problems, Computers & Mathematics with Applications 69 (2015) 1167–1188.
[85] J. Chung, C. A. Brittin, S. D. Evans, N. Cohen, J.-u. Shim, Rotatable microfluidic device for simultaneous study of bilateral chemosensory neurons in caenorhabditis elegans, Microfluidics and Nanofluidics 24 (8) (2020) 1–11.
[86] M. P. Dekkers, F. Salfelder, T. Sanders, O. Umuerri, N. Cohen, G. Jansen, Plasticity in gustatory and nociceptive neurons controls decision making in C. elegans salt navigation, Communications Biology 4 (2021) 1053.