HAL open science

Swimming at Low Reynolds Number

Luca Berti, Laetitia Giraldi, Christophe Prud'Homme

To cite this version:

Luca Berti, Laetitia Giraldi, Christophe Prud'Homme. Swimming at Low Reynolds Number. ESAIM: Proceedings and Surveys, 2020, 67, pp.46-60. 10.1051/proc/202067004 . hal-02369537

HAL Id: hal-02369537

https://hal.science/hal-02369537v1

Submitted on 27 Nov 2019

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.

Swimming at Low Reynolds number

Luca Berti, Laetitia Giraldi, Christophe Prud'homme

November 27, 2019

Abstract. We address the swimming problem at low Reynolds number. This regime, which is typically used for micro-swimmers, is described by Stokes equations. We couple a PDE solver of Stokes equations, derived from the Feel++ finite elements library, to a quaternion-based rigid-body solver. We validate our numerical results both on a 2D exact solution and on an exact solution for a rotating rigid body respectively. Finally, we apply them to simulate the motion of a one-hinged swimmer, which obeys to the scallop theorem.

Resumé. Nous considérons le problème de la nage à bas nombre de Reynolds. Ce régime, typique des micro-organismes nageurs, est décrit par les équations de Stokes. Nous couplons un solveur d'EDP des équations de Stokes, construit à l'aide de la librairie aux éléments finis Feel++, avec un solveur de corps rigide basé sur les quaternions. Nous validons les deux solveurs à l'aide d'une solution exacte en 2D pour le fluide et d'une solution exacte pour un corps rigide qui tourne. Nous les appliquons pour simuler la nage d'un micro-organisme qui obéit au théorème de la coquille de Saint-Jacques.

Introduction

One of the first works in the field of microswimming was proposed by E. Purcell, who presented in 1977 his observations regarding micro-swimmers' swimming strategies. In [33] he noticed that, at the microscale, Reynolds number is small, which means that viscous dissipative forces overcome inertial ones. This observation has two consequences: firstly, Navier-Stokes fluid equations, at the swimmer's scale, can be approximated very well by Stokes equations. These have the characteristics of instantaneously diffusing any change in momentum, and they also guarantee the adherence of the fluid to the boundary of the swimmer at every instant; secondly, certain types of deformations, that usually at human scale produce a net displacement, do not give any motion at the micro-scale. The well-known Scallop theorem [13, 33] specifies some of them.

Since Stokes equations are time-invariant, the important factors, in terms of swimming, are just the body configurations and not the speed of deformation.

A typical value of Reynolds number $Re = \frac{UL}{\nu}$ at the microscale is 10^{-5}. In this case the length-scale of microscopic swimmers is taken to be $L \approx 10^{-6}\, m$, their mean velocity $U \approx 30\ \mu m/s$ and water kinematic viscosity $\nu \approx 10^{-6}\ m^2/s$.

Mathematical analysis of microswimming has mainly developed in the field of control theory, due to its applications to micro-robotics. In fact, it is a matter of great importance to understand how the propelling mechanisms of a robot influence its motion and its controllability. As microrobots might be used as vessels for targeted drug administration or as microsurgery devices, the necessity of control theory becomes clear. In [35, chapter 16] some applications are mentioned: they can range from medical imaging of regions that cannot be reached through endoscopy, to targeted drug delivery or micro-surgery. Different propulsion techniques can be devised: in [2] a combined acoustic and magnetic control was proposed, while in [30] a magnetic control alone was studied.

Microswimming (under the approximation given by the Stokes regime) has already been analyzed from the ODE point of view using Resistive Force Theory [21]. This theory allows the approximation of fluid effects via appropriate coefficients. Such coefficients, inserted in the ordinary differential equations that arise from the Newton equations of motion, approximate horizontal and vertical drag forces for a variety of

swimmers made of simple shapes, like segments or circles. The approximation deriving from Resistive Force Theory makes it possible to use ODE control techniques in this framework [3, 4, 19, 17, 18].

Swimming simulations have been addressed also in the PDE setting by several authors studying fluid-structure interaction. In this case, fluid forces could be computed for swimmers with different shapes. Several techniques can be applied, like level set methods [8, 9], fictitious domain methods [1] or immersed boundary methods [15]. Other schemes, like boundary element methods, can be applied when the governing equations have an associated Green function, like Stokes problem does (see [31]).

Up to now, literature has mainly considered microswimmers that can be represented by finite dimensional control systems, and very often the effects of fluid forces on the body and of its elasticity are modeled in a simplified manner, in order to still use tools from control theory of ODEs. Swimmers of different shapes have been investigated: among them there are Purcell 3-link microswimmer, the N-link microswimmer [3], sphere-based microswimmers [29] and some other models that add elasticity effects through torsional springs or control the swimmer through an externally imposed magnetic field. An important challenge, coming from microrobot control and linked to the applications we have mentioned earlier, is controlling the object close to walls or obstacles. Some results concerning controllability in the half space have been obtained by L. Giraldi [5, 22].

Other studies about control of microswimmers have been presented in [25], where a deformable amoeba was studied, and in subsequent papers by Lohéac and Munnier [27]. Other studies focused on the geometry arising from microswimming, for example [13, 34]. In [13] the scallop theorem is extended using Riemannian geometry and in [11] optimal control of a finite-dimensional swimmer moving on a straight line is studied.

The aim of this paper is to compute the displacement of a microscopic scallop in order to illustrate the Scallop theorem of Purcell, using the finite element library Feel++.

The paper is organised as follows: in Section 1 we will present the mathematical model and the governing equations of swimmer's displacement. Section 2 introduces the associated discrete problem. Section 3 gives some details about the implementation of it. Section 4 presents our numerical results on the scallop swimmer. A paragraph with remarks and perspectives concludes the paper.

1 The model

In this section, the equations that are considered along the paper are introduced. Section 1.1 is devoted to the description of the fluid model. Section 1.2 describes how the rigid body motion can be recovered given an imposed swimmer body deformation.

1.1 The fluid equations

In our problem we consider a micro-swimmer immersed in an incompressible fluid which is modeled, in the Eulerian reference frame, by Navier-Stokes equations

$$\begin{aligned}
&\frac{\partial u}{\partial t} + (u \cdot \nabla)u = \nu \Delta u - \frac{\nabla p}{\rho} + f, \\
&\nabla \cdot u = 0.
\end{aligned}$$

where $u(t,x) : \mathbb{R} \times \mathbb{R}^d \to \mathbb{R}^d$, $d = 2, 3$, is the fluid velocity field, $p(t,x) : \mathbb{R} \times \mathbb{R}^d \to \mathbb{R}$ is the fluid pressure field, ν is the fluid kinematic viscosity, ρ is the fluid density and $f(t,x) : \mathbb{R} \times \mathbb{R}^d \to \mathbb{R}^d$ is the distribution of external forces per unit mass. The Navier-Stokes system, in the case of micro-swimmers, can be simplified to the Stokes equations by neglecting the inertial contributions of the flow.

This simplification is justified by the small value of the Reynolds number associated to the flow regime in which we are interested. This can be seen in the adimensionalized version of Navier-Stokes equations (here without external forces, for simplicity)

$$\begin{aligned}
&\mathrm{Re}\Big(\frac{\partial u'}{\partial t'} + (u' \cdot \nabla)u'\Big) = \Delta u' - \nabla p', \\
&\nabla \cdot u' = 0,
\end{aligned}$$

where $u' = \frac{u}{U}$, $p' = \frac{pL}{\mu U}$, $t' = \frac{L}{U}$ and U is the swimmer's characteristic swimming speed, L is its characteristic length and ν is the fluid kinematic viscosity. In the momentum conservation equations, Reynolds number multiplies the inertial terms, which can therefore be neglected, since in the case of micro-swimmers Re $\approx 10^{-5}$.

As a result, the set of equations we will be using to model the fluid behaviour is the Stokes system

$$\begin{cases} \mu\Delta u - \nabla p = 0, \\ \nabla \cdot u = 0. \end{cases}$$

Denoting by $\sigma(t,x) = -p(t,x)\mathbb{I} + \mu(\nabla u(t,x) + \nabla u(t,x)^T)$ the Newtonian fluid stress tensor and considering a time-dependent family of open sets $\Omega^t \subseteq \mathbb{R}^d\, \forall t$, we can therefore write the fluid problem as follows: find $u(t,x), p(t,x)$ such that

$$\begin{aligned} &\nabla \cdot \sigma = 0 && \text{in } \mathbb{R}_+ \times \Omega^t, \\ &\nabla \cdot u = 0 && \text{in } \mathbb{R}_+ \times \Omega^t, \\ &u(t,x) = \bar{u}(t,x) && \text{on } \mathbb{R}_+ \times \partial\Omega^t_D, \\ &\Big(-p(t,x)\mathbb{I} + \mu\big(\nabla u(t,x) + \nabla u(t,x)^T\big)\Big)\vec{n}(t,x) = \bar{g}(t,x) && \text{on } \mathbb{R}_+ \times \partial\Omega^t_N, \end{aligned} \tag{1}$$

where $\partial\Omega^t_D$ and $\partial\Omega^t_N$ are the portions of $\partial\Omega^t$ where Dirichlet and, respectively, Neumann boundary conditions are imposed, and $\vec{n}(t,x)$ is the outward unit normal to $\partial\Omega^t$. In particular, one requires $\partial\Omega^t = \partial\Omega^t_D \cup \partial\Omega^t_N$ and $\partial\Omega^t_D \cap \partial\Omega^t_N = \emptyset$. Conditions on $\bar{u}$ and $\bar{g}$ are imposed so that the problem has a unique solution in $[\mathrm{H}^1(\Omega^t)]^d \times \mathrm{L}^2(\Omega^t)$, notably $\bar{u} \in \mathrm{H}^{1/2}(\partial\Omega^t_D)$ and $\bar{g} \in \mathrm{H}^{-1/2}(\partial\Omega^t_N)$ provide such result [10].

In the case of our interest, Ω^t will be $\mathbb{R}^d$ deprived of the region occupied by the swimmer, and the Dirichlet boundary conditions correspond to the velocity field on the swimmer boundary. Such vector field can be decomposed in two parts: one represents the rigid motion and the other comes from the deformation of the swimmer's body $\bar{u}(t,x) = V + \omega \wedge (x - x_0) + v(t,x)$, where x_0 is the center of mass of the swimmer. When this latter is known in advance, it is possible to recover the linear velocity V and the angular velocity ω by solving a linear system as presented in [26]. This method exploits the linearity of Stokes equations with respect to pressure and velocity, producing a system of equations which is linear in (V, ω) and that depends on the stresses produced by a number of "fundamentally-driven" flows.

1.2 The rigid body dynamics

Following Newton's second law of dynamics and Euler's equations for angular momentum, we can write the equations describing the dynamics of the swimmer. However, in our case, the microscopic nature of the immersed object and the Stokes approximation of fluid equations lead us to neglect the inertial terms. As a result, the aforementioned equations read respectively (see [26])

$$\begin{aligned} &\int_{\partial\Omega^t} \sigma(t,x)\vec{n}(t,x)\,\mathrm{d}S(x) = 0, \\ &\int_{\partial\Omega^t} \sigma(t,x)\vec{n}(t,x) \wedge x\,\mathrm{d}S(x) = 0. \end{aligned}$$

The equations above are written in the swimmer's reference frame. As proposed and developed in [26], the linearity of the Stokes equation in pressure and velocity allows a decomposition of the two fields as a sum of solutions to fundamentally driven Stokes equations. Each of these problems is characterized by the boundary conditions on the swimmer's surface. If we denote the canonical basis of the swimmer's reference frame by $\{e_i\}_{i=1}^3$ and by $v(t,x)$ the deformation velocity we already mentioned, the fundamental boundary conditions are of the form $\bar{u}(t,x) = e_i$ for $i \in \{1,2,3\}$, $\bar{u}(t,x) = e_i \wedge x$ for $i \in \{1,2,3\}$ and $\bar{u}(t,x) = v(t,x)$. They provide seven different Stokes problems and allow an expansion of the velocity and pressure fields as $u(t,x) = \sum_{i=1}^3 V_i u_i(t,x) + \sum_{i=4}^6 \omega_i u_i(t,x) + u_7(t,x)$ and $p(t,x) = \sum_{i=1}^3 V_i p_i(t,x) + \sum_{i=4}^6 \omega_i p_i(t,x) + p_7(t,x)$,

where problems 1 to 3 are characterized by $\bar{u}(t,x) = e_i$, problems 4 to 6 are characterized by $\bar{u}(t,x) = e_i \wedge x$ and problem 7 by $\bar{u}(t,x) = v(t,x)$. This decomposition carries onto the fluid stress tensors, since they are linear with respect to pressure and velocity: $\sigma(u,p) = \sum_{i=1}^{3} V_i \sigma(u_i,p_i) + \sum_{i=4}^{6} \omega_i \sigma(u_i,p_i) + \sigma(u_7,p_7)$. Substituting this decomposition into (1.2) and appropriately rearranging the terms brings us to the solution of a linear system in (V,ω) of 6×6 matrix M and right-hand side $N \in \mathbb{R}^6$ as follows

$$M_{ij} = \begin{cases} \displaystyle\int_{\partial\Omega^t} e_i \cdot \sigma(u_j,p_j)\vec{n}\,\mathrm{d}S & \text{for } i \in \{1,2,3\}, j \in \{1,\dots,6\}, \\ \displaystyle\int_{\partial\Omega^t} e_{i-3} \cdot \big(\sigma(u_j,p_j)\vec{n} \wedge x\big)\,\mathrm{d}S & \text{for } i \in \{4,5,6\}, j \in \{1,\dots,6\}, \end{cases} \tag{2}$$

$$N_i = \begin{cases} \displaystyle -\int_{\partial\Omega^t} e_i \cdot \sigma(u_7,p_7)\vec{n}\,\mathrm{d}S & \text{for } i \in \{1,2,3\}, \\ \displaystyle -\int_{\partial\Omega^t} e_{i-3} \cdot \big(\sigma(u_7,p_7)\vec{n} \wedge x\big)\,\mathrm{d}S & \text{for } i \in \{4,5,6\}. \end{cases} \tag{3}$$

Once the velocities (in the swimmer's reference frame) are recovered by solving the previous system, as done in [26], we can determine the position and orientation of the swimmer in the laboratory reference frame by solving a set of ODEs of the form [16]

$$\begin{cases} \dfrac{\mathrm{d}q}{\mathrm{d}t} = \dfrac{1}{2}[0, R(q)\omega] * q, \\ \dfrac{\mathrm{d}X}{\mathrm{d}t} = R(q)V, \end{cases} \tag{4}$$

where q is the unit quaternion representing the orientation of the body in the laboratory fixed frame, $R(q)$ is the rotation matrix associated to the unit quaternion q, $*$ denotes the quaternion product and X is the position of the swimmer's centroid in the fixed reference frame. Appropriate initial conditions for the ODEs have to be chosen. The derivation of the quaternion equation and definition of the quaternion product are described in section 2.

Choosing unit quaternions to represent rotations in $\mathbb{R}^d$ is dictated by the fact that they cover SO(3), sending the pair $\{q, -q\}$ to a unique $R(q) \in \mathrm{SO}(3)$, and avoid the classical complications that arise when Euler angles are used [16, 24].

1.3 The self-propulsion constraints

In the previous section we addressed the computation of the linear and angular velocity of the swimmer in its reference frame. In order to model the displacement of a swimmer, the deformation velocity $v(t,x)$ has to satisfy two constraints ensuring that the motion is entirely due to internal forces and is not relying on external actions. In particular, these constraints express the fact that in the swimmer's reference frame, linear and angular momenta remain unchanged because no external forces are applied [12]. In the linear momentum case, such invariance writes

$$\frac{\mathrm{d}}{\mathrm{d}t} \int_{\Omega^0} x\, dm(x) = 0. \tag{5}$$

Following [12], (5) and its angular momentum analogue can be rewritten as

$$\int_{\partial\Omega^t} v(t,x)\,\mathrm{d}S(x) = 0, \qquad \int_{\partial\Omega^t} v(t,x) \wedge x\,\mathrm{d}S(x) = 0.$$

2 Numerical method

In subsection 2.1 we address the discretization of the Stokes system and in subsection 2.2 we will address the derivation and discretization of the ODE system for rigid body motion.

2.1 The fluid model

This part presents the PDE discretization that is used in the rest of the paper to numerically recover the swimmer's motion using finite element method. The indices denoting the time dependence of the fluid domain are abandoned and the focus is placed on the stationary problem to be solved at each time instant. The solution of the Stokes equations was performed via mixed finite elements, where the approximation spaces were chosen in order to guarantee the inf-sup stability of the discrete problem. The variational formulation is the following: find $(u, p) \in [\mathrm{H}^1(\Omega)]^d \cap \{v = \bar{u}|_{\partial\Omega_D}\} \times \mathrm{L}^2(\Omega)$ such that

$$\mu \int_\Omega (\nabla u + \nabla u^T) : \nabla\phi \,\mathrm{d}x - \int_\Omega p(\nabla \cdot \phi)\,\mathrm{d}x = \int_{\partial\Omega_N} g \cdot \phi \,\mathrm{d}S(x),$$
$$\int_\Omega q(\nabla \cdot u)\,\mathrm{d}x = 0,$$

for all $(\phi, q) \in [\mathrm{H}^1_0(\Omega)]^d \cap \{v = 0|_{\partial\Omega_D}\} \times \mathrm{L}^2(\Omega)$ such that the boundary conditions in (1) are satisfied.

For the finite element translation, we chose the pair of Taylor-Hood spaces (X_h^k, M_h^{k-1}) for velocity and pressure, where

$$X_h^k = \{u_h \in [\mathrm{C}^0(\Omega)]^d |\ u_h \text{ is piecewise polynomial of order } k \text{ on each element of the triangulation}\}$$
$$M_h^{k-1} = \{p_h \in \mathrm{C}^0(\Omega)|\ p_h \text{ is piecewise polynomial of order } k-1 \text{ on each element of the triangulation}\}$$

provided that $k \in \mathbb{N}$, $k > 1$. They are characterized by different approximation orders on u and p in order to guarantee the inf-sup stability of the discrete problem. Namely, a continuous, piecewise polynomial approximation of order k is performed on each velocity component and a continuous, piecewise polynomial approximation of order $k-1$ is performed on the pressure.

In the following, we assume that the computational triangulated domain coincides with the continuous one. X_h^k and M_h^{k-1} are subsets of the functional spaces in which (u, p), solutions of the continuous problem, are chosen. We can write the discrete version of the finite element problem as follows: find $(u_h, p_h) \in X_h^k \times M_h^{k-1}$ such that

$$\mu \int_\Omega (\nabla u_h + \nabla u_h^T) : \nabla\phi_h \,\mathrm{d}x - \int_\Omega p_h(\nabla \cdot \phi_h)\,\mathrm{d}x = \int_{\partial\Omega_N} g \cdot \phi_h \,\mathrm{d}S(x),$$
$$\int_\Omega q_h(\nabla \cdot u_h)\,\mathrm{d}x = 0,$$

for all $(\phi_h, q_h) \in X_h^k \times M_h^{k-1}$. Dirichlet boundary conditions are applied strongly at the algebraic level.

2.2 The rigid body dynamics

As shown in subsection 1.2, from the finite element solution of the fluid problem it is possible to recover the rigid body dynamics of the swimmer. Unit quaternions are a good option to represent the orientation of a rigid object, since they avoid parametrization issues that can affect other representations, like the gimbal lock for Euler angles [16, 24].

A quaternion q is defined by a scalar q_0 and a 3-dimensional vector $q_v = (q_{v_1}, q_{v_2}, q_{v_3})$, $q = (q_0, q_v)$. The conjugate of q will be denoted by $\bar{q} = (q_0, -q_v)$ and the quaternion product is defined as $q * r = (q_0 r_0 - q_v \cdot r_v\,,\ q_0 r_v + q_v r_0 + q_v \wedge r_v)$. The norm of a quaternion is defined as $||q|| = \sqrt{q\bar{q}} = \sqrt{q_0^2 + q_{v_1}^2 + q_{v_2}^2 + q_{v_3}^2}$. Each quaternion of unit norm represents a reflection in 3D space, and together with its conjugate concurs to describe a rotation in space. The rotation matrix linked to a unit quaternion $q = (q_0, q_v)$ is given by

$$R(q) = \begin{bmatrix} q_0^2 + q_{v_1}^2 - q_{v_2}^2 - q_{v_3}^2 & 2(q_{v_1}q_{v_2} - q_0 q_{v_3}) & q_{v_1}q_{v_3} + q_0 q_{v_2} \\ 2(q_{v_1}q_{v_2} + q_0 q_{v_3}) & q_0^2 - q_{v_1}^2 + q_{v_2}^2 - q_{v_3}^2 & 2(q_{v_2}q_{v_3} - q_0 q_{v_1}) \\ 2(q_{v_1}q_{v_3} - q_0 q_{v_2}) & 2(q_{v_2}q_{v_3} + q_0 q_{v_1}) & q_0^2 - q_{v_1}^2 - q_{v_2}^2 + q_{v_3}^2 \end{bmatrix}.$$

In what follows we present how to get the swimmer frame dynamics in the laboratory frame. Coordinates X in the laboratory frame are expressed in the swimmer frame by $[0, x] = q * [0, X] * \bar{q}$. Performing time differentiation we get

$$\frac{\mathrm{d}[0,x]}{\mathrm{d}t} = \frac{\mathrm{d}q}{\mathrm{d}t} * [0, X] * \bar{q} + q * [0, X] * \frac{\mathrm{d}\bar{q}}{\mathrm{d}t} = \frac{\mathrm{d}q}{\mathrm{d}t} * \bar{q} * [0, x] + [0, x] * q * \frac{\mathrm{d}\bar{q}}{\mathrm{d}t}.$$

Since

$$\frac{\mathrm{d}[0,x]}{\mathrm{d}t} = [0, 2v \wedge x],$$

in our case $2[0, v] = 2\frac{\mathrm{d}q}{\mathrm{d}t} * \bar{q} = [0, R(q)\omega]$. Then, multiplying both sides of the equality by q gives the following dynamics for the quaternion

$$\frac{\mathrm{d}q}{\mathrm{d}t} = \frac{1}{2}[0, R(q)\omega] * q. \tag{6}$$

3 Implementation

The implementation part combined the FEEL++ finite element library [32] to solve the Stokes problems with an ODE solver for the solution of the rigid body motion. Matlab and Paraview were used for the post-processing.

Since the methodology giving the rigid body problem was posed in the swimmer's reference frame and the swimmer's motion in the computational domain produced strong deformations on the underlying triangular mesh, at every time-step the domain was displaced and remeshed. This procedure was case specific, since both the shape and the deformation velocity on the swimmer's boundary were externally imposed.

Thanks to FEEL++, a Stokes solver with a syntax closely resembling the finite element variational formulation is achieved. Moreover, implementation of the finite elements was strongly simplified: Taylor-Hood spaces of order P^2/P^1 are chosen, but also spaces of higher order (ex. P^3/P^2) could be supported easily.

The solution of the ODEs (4) describing rigid-body motion was embedded in the code through an external ODE solver in C++. A Runge-Kutta scheme was performed. At the end of each time-step, before the new rotation matrix is computed, the quaternion needs to be normalized to have unit norm: the propagation of numerical errors does not preserve the norm of the quaternion. Remark that, if this renormalization is not performed, the rotation matrix won't be orthogonal and the body would suffer a deformation.

4 Numerical tests

4.1 Validation of fluid solver

To validate the fluid solver presented in this paper, we compared the analytical solution of Stokes problem presented in [7] to the numerical results given by our simulations. The analytical solution to Stokes problem on the unit square

$$\begin{cases} -\Delta u + \nabla p = f & \text{on } [0,1] \times [0,1], \\ \nabla \cdot u = 0 & \text{on } [0,1] \times [0,1], \end{cases} \tag{7}$$

with forcing function

$$f(t,(x,y)) = \begin{bmatrix} 128[x^2(x-1)^2 12(2y-1) + 2(y-1)(2y-1)y(12x^2 - 12x + 2)] + y - 0.5 \\ -128[y^2(y-1)^2 12(2x-1) + 2(x-1)(2x-1)x(12y^2 - 12y + 2)] + x - 0.5 \end{bmatrix},$$

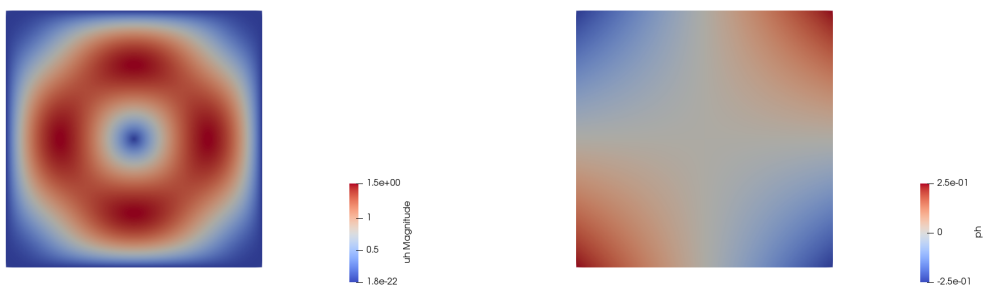
Dirichlet boundary conditions derived from the exact solution presented below and zero-mean pressure is given by

$$u(t,(x,y)) = \begin{bmatrix} -256y(y-1)(2y-1)x^2(x-1)^2 \\ 256x(x-1)(2x-1)y^2(y-1)^2 \end{bmatrix}, \tag{8}$$

$$p(t,(x,y)) = (x-0.5)(y-0.5).$$

Figure 1 presents this latter Stokes solution. We solve the Stokes equations using the pair (X_h^2, M_h^1) for the mixed finite elements. Figure 2 shows a third order convergence of the L^2 norm of the velocity and a second order convergence for both the L^2 norm of the pressure and the H^1 norm of the velocity are obtained. These results are those we expected from the finite element method theory, see for instance [20, Thm 4.3, pag 181].

Remark that, even though the solver is made for the 2D case, Feel++ allows to easily obtain a three dimensional Stokes solution when providing an appropriate 3D mesh.



Velocity magnitude Pressure

Figure 1: Analytical solution of the Stokes problem [7].

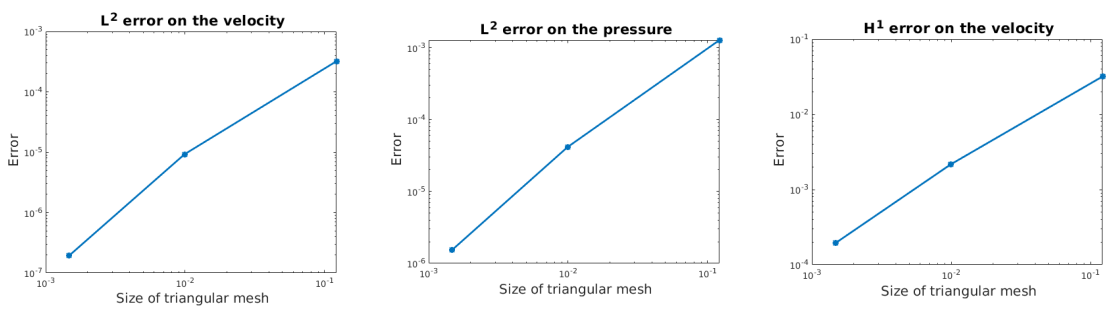


Figure 2: Error convergence of the numerical solution given by our solver. The errors are in good agreement with the classical errors given by the finite element method: the experimental order of convergence for the velocity in L^2 norm is approximately 3, and in H^1 norm it's approximately 2. The experimental order of convergence for the L^2 norm of the pressure is approximately 2.

4.2 Validation of rigid body solver

In order to validate the rigid body solver, we considered the results presented in [28]. The solution to

$$\frac{\mathrm{d}X}{\mathrm{d}t} = R(q)V, \tag{9a}$$

$$\frac{\mathrm{d}q}{\mathrm{d}t} = \frac{1}{2}[0, R(q)\omega] * q, \tag{9b}$$

$$\frac{\mathrm{d}V}{\mathrm{d}t} = F/m, \tag{9c}$$

$$I\frac{\mathrm{d}\omega}{\mathrm{d}t} + \omega \wedge I\omega = T, \tag{9d}$$

complemented with adequate initial conditions, is searched.

Let us remark that the exactness of the translational part of the rigid body motion is easy to check. As a result, we will consider only equations (9b) and (9d).

We validate the rotational part of the rigid body motion by extracting a set of sample points from the graphics of [28] and compared qualitatively such results with ours. The case we will consider is the following: the time-varying torque in the body frame will be given by

$$T = \begin{bmatrix} 1.0 + 2.7 \times 10^{-2} t - 2.4 \times 10^{-4} t^2 + 5.7 \times 10^{-7} t^3 \\ -1.5 - 9.0 \times 10^{-3} t + 1.2 \times 10^{-4} t^2 - 3.0 \times 10^{-7} t^3 \\ 13.5 \end{bmatrix} N \cdot m.$$

The body reference frame at $t = 0$ coincides with the laboratory reference frame, which in terms of quaternions is expressed as $q(0) = (1, 0, 0, 0)$. The components of the angular velocity ω at $t = 0$ along the three principal axes are $(0, 0, 0.329)\, rad/s$, while the inertia matrix is given by

$$I = \begin{bmatrix} 2985 & 0 & 0 \\ 0 & 2729 & 0 \\ 0 & 0 & 4183 \end{bmatrix} kg \cdot m^2$$

Since in [28] the results about the frame orientation are given in terms of Euler angles (ϕ_x, ϕ_y, ϕ_z), we use the following relationship to obtain them from a quaternion $q = (q_0, q_1, q_2, q_3)$:

$$\begin{bmatrix} \phi_x \\ \phi_y \\ \phi_z \end{bmatrix} = \begin{bmatrix} \arctan \frac{2(q_0 q_1 + q_2 q_3)}{1 - 2(q_1^2 + q_2^2)} \\ \arcsin(2(q_0 q_2 - q_3 q_1)) \\ \arctan \frac{2(q_0 q_3 + q_1 q_2)}{1 - 2(q_2^2 + q_3^2)} \end{bmatrix}$$

Figure 3 shows that our results are in agreement with the numerical results in [28].

We remark that, even though we will be focusing on a two-dimensional simulation for the swimmer, the previous validation confirms that such solver can be used also on a 3D experiment. The resulting angular velocity is the one required in (4) to obtain the body orientation.

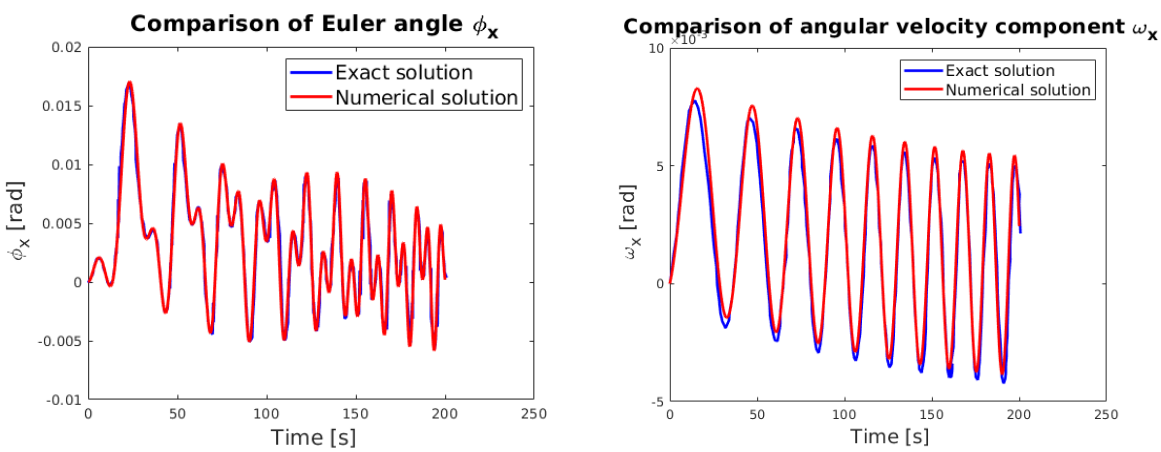


Figure 3: Comparison of the sampled solution from [28] and our numerical solution from equations (9). On the left, the results on Euler angle ϕ_x are compared. On the right, the results on the component ω_x of the angular velocity are compared.

4.3 Validation of force and torque computations

Given (u, p) solution of Stokes equations, this part addresses the numerical computation of fluid forces and torques. Since the swimmer motion depends on them, it is fundamental to validate the techniques we use

to compute them. Two methods are compared: the first method consists in approximating the integrals describing surface forces and torques

$$\int_{\partial\Omega} \sigma(t,x)\vec{n}(t,x)\,\mathrm{d}S(x), \qquad \int_{\partial\Omega} \sigma(t,x)\vec{n}(t,x)\wedge x\,\mathrm{d}S(x),$$

via surface quadratures, while the second method consists in exploiting the variational formulation to resort to volume integrals

$$\begin{aligned} \int_{\partial\Omega} \sigma(t,x)\vec{n}(t,x)\,\mathrm{d}S(x) &= \int_{\Omega} f\,\mathrm{d}x + \int_{\Omega} \sigma(t,x) : \nabla_x u\,\mathrm{d}x, \\ \int_{\partial\Omega} \sigma(t,x)\vec{n}(t,x)\wedge x\,\mathrm{d}S(x) &= \int_{\Omega} f\wedge x\,\mathrm{d}x + \int_{\Omega} (\sigma(t,x) : \nabla_x u)\wedge x\,\mathrm{d}x, \end{aligned} \tag{10}$$

that are subsequently approximated via 3D volumic quadratures. Feel++ library was used to perform these computations. The numerical results will be compared to approximate and exact solutions. We considered two test cases: the first one is the 3D Stokes example of a translating and rotating sphere, for which the approximate solutions are provided in [23, p.67-68], while the second one is a 3D example taken from [14]. In the moving sphere case, the experimental setting is composed of a sphere of radius $r = 0.01$ placed in the middle of a cubic box of size 1, $\mu = 1$ and $U = \omega = 2$. The approximated forces and torques, given by the formulas $F = -6\pi\mu r U$ and $T = -8\pi\mu r^3\omega$, give a qualitative target for the numerically derived forces and torques.

The results regarding the 3D Stokes case are collected in Tables 1 to 4. Tables 1 and 2 show the numerical results that were obtained when using X_h^2/M_h^1 finite elements on a piecewise linear approximation of the geometry: Table 1 shows that the surface quadratures give better approximation of forces while Table 2 show that torques are better approximated when using volumic quadratures. Tables 3 and 4 show the numerical results that were obtained when X_h^3/M_h^2 finite elements on a piecewise quadratic approximation of the geometry were used: in this case volumic quadrature performs better than surface quadratures in both cases.

h_{min}	h_{max}	Surface quad.	Volume quad.	Stokes
0.00099	0.1958	-0.397501	-0.395855	-0.376991
0.00052	0.1546	-0.394835	-0.395245	-0.376991
0.00036	0.1211	-0.393037	-0.393478	-0.376991

Table 1: Forces comparison for the translating and rotating sphere with piecewise linear approximation of the geometry and X_h^2/M_h^1 finite elements. In the last column, the numerical result of the Stokes formula $F = -6\pi\mu r U$ is reported.

h_{min}	h_{max}	Surface quad.	Volume quad.	Stokes
0.00099	0.1958	$-4.97575 \cdot 10^{-5}$	$-5.0582 \cdot 10^{-5}$	$-5.02655 \cdot 10^{-5}$
0.00052	0.1546	$-4.96165 \cdot 10^{-5}$	$-4.98904 \cdot 10^{-5}$	$-5.02655 \cdot 10^{-5}$
0.00036	0.1211	$-5.03447 \cdot 10^{-5}$	$-5.03904 \cdot 10^{-5}$	$-5.05423 \cdot 10^{-5}$

Table 2: Torques comparison for the translating and rotating sphere with piecewise linear approximation of the geometry and X_h^2/M_h^1 finite elements. In the last column, the numerical result of the Stokes formula $T = -8\pi\mu r^3\omega$ is reported.

h_{min}	h_{max}	Surface quad.	Volume quad.	Stokes
0.00099	0.1958	-0.39264	-0.392629	-0.376991
0.00052	0.1546	-0.392439	-0.392408	-0.376991

Table 3: Forces comparison for the translating and rotating sphere with piecewise quadratic approximation of the geometry and X_h^3/M_h^2 finite elements. In the last column, the numerical result of the Stokes formula $F = -6\pi\mu r U$ is reported.

h_{min}	h_{max}	Surface quad.	Volume quad.	Stokes
0.00099	0.1958	$-5.02279 \cdot 10^{-5}$	$-5.02783 \cdot 10^{-5}$	$-5.02655 \cdot 10^{-5}$
0.00052	0.1546	$-5.02507 \cdot 10^{-5}$	$-5.02577 \cdot 10^{-5}$	$-5.02655 \cdot 10^{-5}$

Table 4: Torques comparison for the translating and rotating sphere with piecewise quadratic approximation of the geometry and X_h^3/M_h^2 finite elements. In the last column, the numerical result of the Stokes formula $T = -8\pi\mu r^3\omega$ is reported.

The second benchmark that was considered regarded the exact computation of forces from an exact and fully three-dimensional solution to Navier-Stokes equations (where we neglected the time variation) [14]

$$u = -a \begin{bmatrix} e^{ax}\sin(ay+dz) + e^{az}\cos(ax+dy) \\ e^{ay}\sin(az+dx) + e^{ax}\cos(ay+dz) \\ e^{az}\sin(ax+dy) + e^{ay}\cos(az+dx) \end{bmatrix}, \tag{11}$$

$$p = -\frac{a^2}{2}[e^{2ax} + e^{2ay} + e^{2az} + 2\sin(ax+dz)\cos(az+dx)e^{ay+az} \\ + 2\sin(ay+dz)\cos(ax+dy)e^{az+ax} + 2\sin(az+dx)\cos(ay+dz)e^{ax+ay}] \tag{12}$$

in the cube $[-1,1]^3$, with $a = \frac{\pi}{4}$ and $d = \frac{\pi}{2}$.

The exact forces were calculated using the symbolic mathematics package *Sympy*, and they were compared to forces issued from surface and volume quadratures. The results in Figure 4 show that, after fixing a high quadrature order for both surface and volume quadratures (13 in our case), forces are better approximated when using volume quadratures on (10) than surface quadratures. These two benchmarks encouraged us to use the volume formulation to compute forces and torques for the Scallop simulation in the next section.

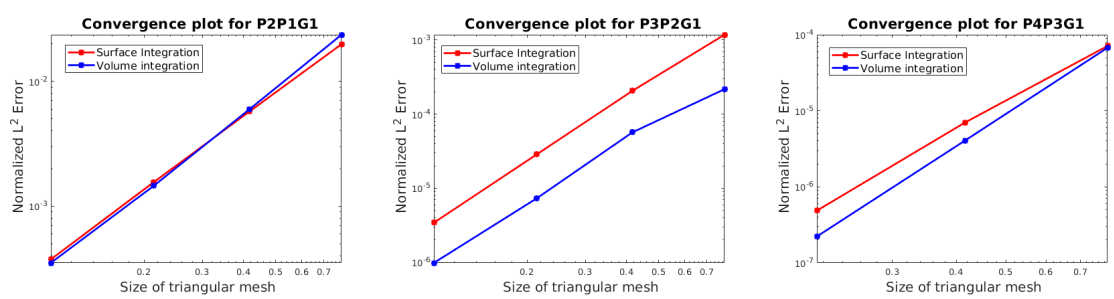


Figure 4: Comparison in L^2 norm between exact and approximated forces, normalized by the L^2 norm of the exact force. These pictures show that volume quadratures give a better approximation of forces and torques as the mesh size decreases. All the figures were obtained using a linear approximation of the geometry (G1). The left figure shows the results for X_h^2/M_h^1 finite elements. The center figure shows the results for X_h^3/M_h^2 finite elements. The right figure shows the results for X_h^4/M_h^3 finite elements.

Figure 5: Screenshots of the scallop's body configurations over a period. The opening and closure of the swimmer's valves produce the motion. Since it has only one degree of freedom, the scallop does not experience a net advancement over a period.

4.4 Scallop theorem

In this section we will address the simulation of the so called "Scallop theorem". This theorem was first presented by Purcell in [33], and later generalized in [13] for other types of swimmers, highlighting the underlying geometrical properties of the problem. Swimming is characterized by periodical strokes, and the scallop theorem specifies when they do not provide a net advancement. Scallop theorem, as in [33], was specialized to Stokes flows, and stated that a swimmer which performs time-reversible shape changes will not be able to advance over a period. A particular case, from which the theorem takes his name, is the case of one-hinged swimmers. Having just one degree of freedom, swimmers like the scallop are forced to perform time-reversible shape changes and are not able to obtain a net advancement over a period of time. Figure 5 shows an example of these shape changes.

The aim of this part is to verify that our numerical solver is able to reproduce the scallop theorem. Problem (1) is solved and coupled with (4) to obtain the swimmer's displacement and change in orientation. Here Ω^0 is the complement in $[0, 0.1] \times [0, 0.1]$ of the scallop and the deformation velocity on the swimmer's boundary point x is given by $v(t, x) = \frac{\mathrm{d}A}{\mathrm{d}t}(t)x$, where $A(t)$ is the rotation matrix imposing the desired opening and closing of the valves.

We imposed on the boundary of the swimmer a displacement field which caused its hinges to open and close in a periodical fashion. We also allow a slight deformation of the body, to see whether any changes can be experienced with respect to the invariance predicted in [13, Section 3.5].

The computational domain coincides with Ω^0, hence it consists in a pierced square whose hole has the shape of the swimmer. The mesh is conformal to the swimmer's geometry and the approximation is linear: interior and boundary edges are straight.

Figure 6 shows that the scallop theorem is not exactly satisfied due to the domain discretization, but the errors over a period, and the errors cumulated between two consecutive periods, are negligible. We chose a period $T = 2\,s$ and a swimmer with total valve opening equal to $14\,mm$. The smallest triangulation edge size in is of $0.025\,mm$.

We also located the scallop close to a no slip boundary of the computational domain. In that case the scallop theorem is not satsfied since the wall breaks the symmetry of the whole system. Figure 7 shows the displacement over a complete stroke period as a function of the proximity to the wall. In that case we noticed that the displacement at the end of the period was sensibly larger than what Figure 6 showed, and it exemplifies both the attractive effect that walls have on swimmers (Figure 7 right) and the enhanced displacement parallel to no slip boundaries (Figure 7 left). Let us quote the recent study [6] that overcomes the scallop theorem by opening and closing the shell with different speeds.

The pseudocode for the simulation of the swimmer motion is presented in Algorithm 1.

5 Conclusion and perspectives

In this note we present a solver able to provide the dynamics of a derforming swimmer moving into a Stokes flow. Here all the framework could fit a 3D model but the numerical results are given in 2D. All the numerical computations of this note relied on the finite element library Feel++. We applied a method proposed in [26] to extract rigid body motion from computation of fluid stresses. We validated our code and model using

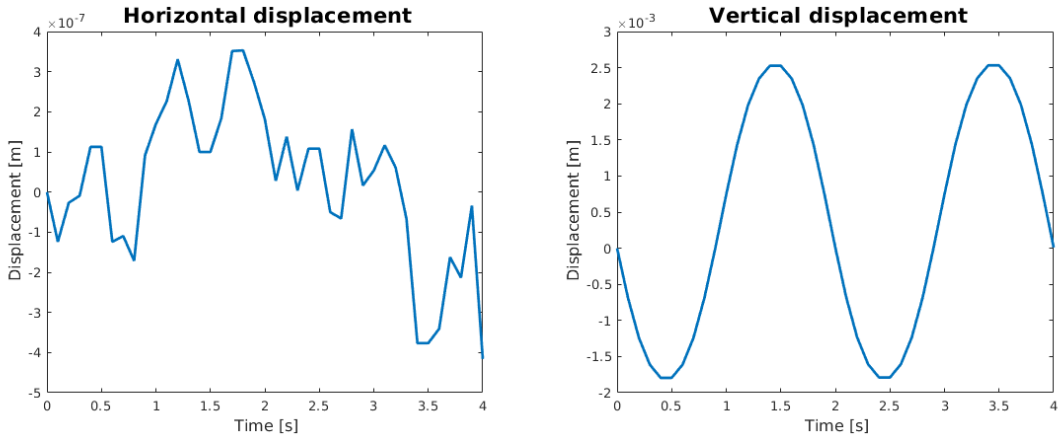


Figure 6: Position of the swimmer's center of mass in the laboratory reference frame. In this case the swimmer is located far from the no slip boundaries of the computational domain. The figure on the left shows its horizontal displacement, along the x axis: it's globally negligible and imputable to numerical errors. The figure on the right shows the swimmer's vertical displacement: it is periodic of period $T = 2$, and the net motion value on a period is also imputable to numerical errors.

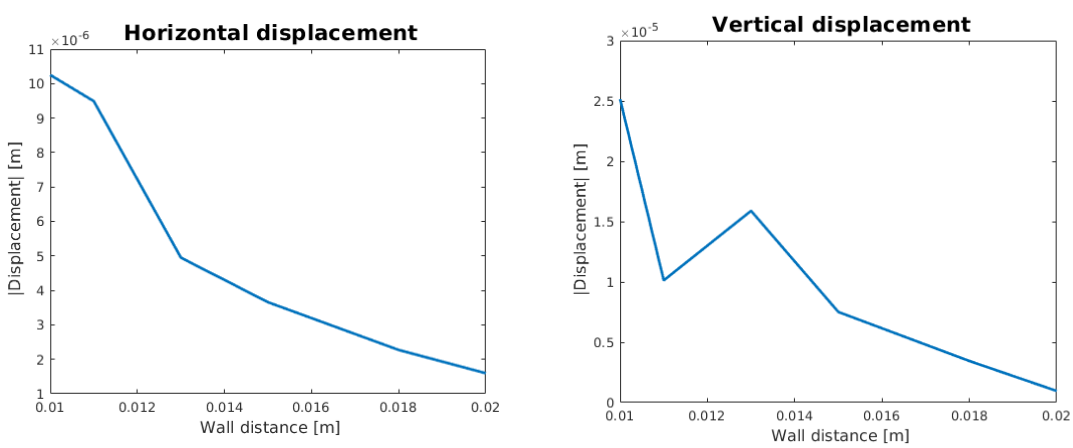


Figure 7: Absolute value of the net displacement of the swimmer's center of mass, as a function of its distance to a wall. When the swimmer departs closer to the no slip boundary, it experiences a larger displacement over a time period, both in the horizontal and vertical direction.

Algorithm 1 Solution procedure for the swimming problem

```
Choose Δt, d, T_final
Initialize X, q, V, ω
if d = 2 then
  N_Stokes = 4
else if d = 3 then
  N_Stokes = 7
end if
while time ≤ T_final do
  Displace the swimmer according to X, q, V, ω and Remesh
  for 1 ≤ i ≤ N_Stokes do
    Solve Stokes problem i from subsection 1.2
    Compute surface forces and torques issued from Stokes problem i
  end for
  Find V, ω by using (2) and (3)
  Find X, q by solving (4)
  time ← time + Δt
end while
```

different benchmarks. The goal of this project was to lay the foundations of a numerical method which allowed the simulation of swimmers with more complex geometries.

The passage to several swimmers or to 3D are beyond the scope of this paper. The technique could be applied to several swimmers, but the basis onto which the Stokes solution has to be expanded is larger and the computational cost might increase noticeably. Another issue is that, in the Stokes regime, forces can become infinite when two swimmers are close, and collision avoidance strategies would need to be implemented.

References

[1] N. Aguillon, A. Decoene, B. Fabrèges, B. Maury, and B. Semin, *Modelling and simulation of 2d stokesian squirmers*, ESAIM: Proceedings, 38 (2013), pp. 36–53.

[2] D. Ahmed, C. Dillinger, A. Hong, and B. J. Nelson, *Artificial acousto-magnetic soft microswimmers*, Advanced Materials Technologies, 2, p. 1700050.

[3] F. Alouges, A. DeSimone, L. Giraldi, and M. Zoppello, *Self-propulsion of slender microswimmers by curvature control: N-link swimmers*, International Journal of Non-Linear Mechanics, 56 (2013), pp. 132 – 141. Soft Matter: a nonlinear continuum mechanics perspective.

[4] F. Alouges, A. Desimone, L. Giraldi, and M. Zoppello, *Can magnetic multilayers propel artificial microswimmers mimicking sperm cells?*, Soft Robotics, 2 (2015), pp. 117–128.

[5] F. Alouges and L. Giraldi, *Enhanced controllability of low reynolds number swimmers in the presence of a wall*, Acta Applicandae Mathematicae, 128 (2012).

[6] F. Bagagiolo, R. Maggistro, and M. Zoppello, *Swimming by switching*, Meccanica, 52 (2017), pp. 3499–3511.

[7] M. Bercovier and M. Engelman, *A finite element for the numerical solution of viscous incompressible flows*, Journal of Computational Physics, 30 (1979), pp. 181 – 201.

[8] M. Bergmann and A. Iollo, *Modeling and simulation of fish-like swimming*, Journal of Computational Physics, 230 (2011), pp. 329–348.

[9] ——, *Bioinspired swimming simulations*, Journal of Computational Physics, 323 (2016).

[10] F. Brezzi and M. Fortin, *Mixed and hybrid finite elements methods*, Springer series in computational mathematics, Springer-Verlag, 1991.

[11] T. Chambrion, L. Giraldi, and A. Munnier, *Optimal strokes for driftless swimmers: A general geometric approach*, ESAIM: Control, Optimisation and Calculus of Variations, (2017).

[12] T. Chambrion and A. Munnier, *Locomotion and control of a self-propelled shape-changing body in a fluid*, Journal of Nonlinear Science, 21 (2009).

[13] T. Chambrion and A. Munnier, *Generalized scallop theorem for linear swimmers*, 2010.

[14] C. Ethier and D. A. Steinman, *Exact fully 3d navier–stokes solution for benchmarking*, International Journal for Numerical Methods in Fluids, 19 (1994), pp. 369 – 375.

[15] L. J. Fauci and A. McDonald, *Sperm motility in the presence of boundaries*, Bulletin of Mathematical Biology, 57 (1995), pp. 679 – 699.

[16] R. Featherstone, *Rigid Body Dynamics Algorithms*, Springer-Verlag, Berlin, Heidelberg, 2007.

[17] L. Giraldi, P. Lissy, C. Moreau, and J. Pomet, *Addendum to "local controllability of the two-link magneto-elastic microswimmer"*, IEEE Transactions on Automatic Control, 63 (2018), pp. 2303–2305.

[18] L. Giraldi, P. Martinon, and M. Zoppello, *Optimal design of purcell's three-link swimmer*, Phys. Rev. E, 91 (2015), p. 023012.

[19] L. Giraldi and J. Pomet, *Local controllability of the two-link magneto-elastic micro-swimmer*, IEEE Transactions on Automatic Control, 62 (2017), pp. 2512–2518.

[20] V. Girault and P. Raviart, *Finite element methods for Navier-Stokes equations: theory and algorithms*, Springer series in computational mathematics, Springer-Verlag, 1986.

[21] J. Gray and G. J. Hancock, *The propulsion of sea-urchin spermatozoa*, Journal of Experimental Biology, 32 (1955), pp. 802–814.

[22] D. Gérard-Varet and L. Giraldi, *Rough wall effect on micro-swimmers*, ESAIM: Control, Optimisation and Calculus of Variations, 21 (2013).

[23] J. Happel and H. Brenner, *Low Reynolds number hydrodynamics: with special applications to particulate media*, Mechanics of Fluids and Transport Processes, Springer Netherlands, 1983.

[24] L. Landau and E. Lifshitz, *Mechanics*, Butterworth-Heinemann, Elsevier Science, 1976.

[25] J. Lohéac, *Contrôle en temps optimal et nage à bas nombre de Reynolds*, PhD thesis, 2012. Thèse de doctorat dirigée par Tucsnak, Marius et Scheid, Jean-François Mathématiques Université de Lorraine 2012.

[26] J. Lohéac and A. Munnier, *Controllability of 3d low reynolds swimmers*, ESAIM Control Optimisation and Calculus of Variations, 20 (2012).

[27] ——, *Controllability of 3d low reynolds number swimmers*, ESAIM: Control, Optimisation and Calculus of Variations, 20 (2014), p. 236–268.

[28] J. Longuski and P. Tsiotras, *Analytical solutions for a spinning rigid body subject to time-varying body-fixed torques, part i: Constant axial torque.*, Journal of Applied Mechanics-transactions of The Asme, (1993).

[29] A. Montino and A. Desimone, *Three-sphere low-reynolds-number swimmer with a passive elastic arm*, The European physical journal. E, Soft matter, 38 (2015), p. 127.

[30] A. Oulmas, N. Andreff, and S. Régnier, *3d closed-loop motion control of swimmer with flexible flagella at low reynolds numbers*, 09 2017.

[31] D. Pimponi, M. Chinappi, and P. Gualtieri, *Flagellated microswimmers: Hydrodynamics in thin liquid films*, 41 (2018).

[32] C. Prud'homme, V. Chabannes, V. Doyeux, M. Ismaïl, A. Samake, and G. Pena, *Feel++: A computational framework for galerkin methods and advanced numerical methods*, ESAIM Proceedings, 38 (2013), p. 27.

[33] E. M. Purcell, *Life at low reynolds number*, American Journal of Physics, 45 (1977), pp. 3–11.

[34] A. Shapere and F. Wilczek, *Geometry of self-propulsion at low reynolds number*, Journal of Fluid Mechanics - J FLUID MECH, 198 (1989), pp. 557–585.

[35] A. VV., *Surgical Robotics. Systems Applications and Visions*, Springer.