## International Journal of Heat and Fluid Flow 51 (2015) 88-109

Contents lists available at ScienceDirect



International Journal of Heat and Fluid Flow

journal homepage: www.elsevier.com/locate/ijhff



# Simulation of forced deformable bodies interacting with two-dimensional incompressible flows: Application to fish-like swimming



## Seyed Amin Ghaffari, Stéphane Viazzo, Kai Schneider, Patrick Bontoux\*

Aix-Marseille Université, CNRS, École Centrale Marseille, Laboratoire M2P2UMR 7340, 13451, Marseille, France

## ARTICLE INFO

Article history: Available online 3 December 2014

Keywords: Fluid interaction with forced deformable bodies Compact fourth-order direct Poisson solver Volume penalization Fish swimming/turning

## ABSTRACT

We present an efficient algorithm for simulation of deformable bodies interacting with two-dimensional incompressible fluid flows. The temporal and spatial discretizations of the Navier–Stokes equations in vorticity stream-function formulation are based on classical fourth-order Runge–Kutta scheme and compact finite differences, respectively. Using a uniform Cartesian grid we benefit from the advantage of a new fourth-order direct solver for the Poisson equation to ensure the incompressibility constraint down to machine zero over an optimal grid. For introducing a deformable body in fluid flow, the volume penalization method is used. A Lagrangian structured grid with prescribed motion covers the deformable body which is interacting with the surrounding fluid due to the hydrodynamic forces and the torque calculated on the Eulerian reference grid. An efficient law for controlling the curvature of an anguilliform fish, swimming toward a prescribed goal, is proposed which is based on the geometrically exact theory of nonlinear beams and quaternions. Validation of the developed method shows the efficiency and expected accuracy of the algorithm for fish-like swimming and also for a variety of fluid/solid interaction problems.

© 2014 Elsevier Inc. All rights reserved.

## 1. Introduction

The quantification and simulation of the flow around biological swimmers is one of the challenges in fluid mechanics (Sotiropoulos and Yang, 2014). At the same time bio-inspired design of swimming robots are in growth (El Rafei et al., 2008). The costs of experimental studies (Belkhiri, 2013) lead the researchers to develop efficient predictive numerical algorithms for hydrodynamic analyses of fish swimming. Difficulties of numerical simulations of fish-like swimming are due to different reasons. One problem is efficient quantification of the kinematics of different species (more than 32,000) which seems to be far from the simple laws proposed in different studies. Efficient simulation of incompressible flows is also an important problem, where the efficiency of the elliptic solver is crucial. The third bottleneck in numerical simulations of fish-like swimming is the coupling of the fluid solver with deformable, moving and rotating bodies. Fishes swim by exerting force and torque against the surrounding water. This is normally done by the fish contracting muscles on either side of its body in order to generate moving waves from head to tail. These waves generally are getting larger as they go toward the tail

\* Corresponding author. *E-mail address:* bontoux@L3M.univ-mrs.fr (P. Bontoux). (Wikipedia contributers, 2014). The resultant force exerted on the water by such motion generates a force (even oscillatory) which pushes the fish forward. Most fishes generate thrust moving their body and fins. In general these movements can be divided into undulatory and oscillatory motions. Mechanisms of locomotion using body and fins are divided into groups that differ in the fraction of their body that is displaced laterally (Breder, 1926). Anguilliform swimmers are long and slender, in which there is little increase in the amplitude of the flexion wave as it passes along the body. In carangiform swimmers, there is a more remarkable increase in wave amplitude along the body with the vast majority of the work being done by the rear half of the fish. In thunniform fishes almost all the lateral movement is in the tail. Ostraciiform fishes have no appreciable body wave when they employ caudal locomotion, only the tail fin itself oscillates rapidly to create thrust. However there are other minorities (Wikipedia contributers, 2014). The tail beat creates a reversed Kármán street of vortices and generates thrust, leaving thus a momentumless wake back. By varying the frequency and amplitude of the oscillation a variety of wakes, like classical Kármán, two pairs (2P) (Van Rees et al., 2013), two pairs plus two single (2P+2S), etc. Schnipper et al. (2009) can be observed (Williamson and Roshko, 1988). Anguilliform fishes add a constant curvature to their backbone for turning, i.e., they use their body like a rudder for torque generation. Yeo

et al. (2010) studied numerically the straight swimming/cruising and sharp turning manoeuvres in two-dimensions. It was shown by Yeo et al. (2010) that a carangiform-like swimmer execute a sharp turn through an angle of 70° from straight coasting within a space of about one body length. Gazzola et al. (2012) investigated the C-start escape patterns of a larval fish by using a remeshed vortex particle method and the volume penalization. The deformation of the fish, based on the mid-line curvature values, is optimized via an evolutionary strategy by Hansen et al. (2003) to maximize the escape distance. Bergmann and Iollo (2011) performed numerical simulations of fish rotation and swimming toward a prescribed goal. They considered the average profile of the fish backbone aligns over a circle with an estimated radius to perform a rotation. The radius of the circle tends to infinity  $(r \rightarrow \infty)$  in a forward gait. The considered fish by Bergmann and Iollo (2011) is constructed by a complex valued mapping like the Kutta-Joukowski transform superposed to the fish backbone with prescribed undulatory motion. Here we will present a simple law for turning control of an anguilliform fish. Our rotation control law (Bontoux et al., 2014) is similar to that presented by Yeo et al. (2010), and Bergmann and Iollo (2011), in which the feedback is based on the angle between the line-of-sight and the direction of surge. But instead of adding a radius to the backbone, we envisage to use curvature which seems to be more efficient. We use the method proposed by Boyer et al. (2006) which is based on quaternions for efficient description of the fish backbone kinematics.

We apply the rotation control to two-dimensional swimming. Even if due to the shape and deformation style of the fish-like swimmers the surrounding flow is fully three dimensional, most of the fundamental features of swimming are included in twodimensional analyses. For incompressible flows the Navier-Stokes equations can be reformulated in terms of vorticity-velocity (Gazzola et al., 2011) or vorticity stream-function (Spotz and Carey, 1995). For two-dimensional problems the vorticity formulation is reduced to a scalar valued evolution equation. Hence only the vorticity transport equation has to be advanced in time. The choice of finite differences in this paper is related to the use of an immersed boundary method in which a Cartesian grid can be used. Therefore the use of finite differences is efficient and straightforward. Among finite difference methods high-order compact discretizations, (Hirsh, 1975; Lele, 1992), are more advantageous in terms of accuracy and reasonable cost. We refer to Abide and Viazzo (2005) and Boersma (2011) for high-order compact discretizations of the incompressible Navier-Stokes equations in primitive variables and to Bontoux et al. (1978), Roux et al. (1980), and Spotz and Carey (1995) for compact high-order solutions of the vorticity and stream-function formulation. Solving the incompressible Navier-Stokes equations typically implies an elliptic Poisson equation which is the most time consuming part of the algorithm. Direct methods like diagonalization or FFT based solvers can be used. Iterative methods, namely, point successive over relaxation (PSOR) with read-black sweeper, multigrid or Krylov subspace solvers are other alternatives. Using high-order discretizations iterative methods are less attractive because the resulted matrices are less sparse, thus the rates of convergence are slow. However iterative methods can cover all types of boundary conditions, we refer to Spotz and Carey (1995) for a fourth-order compact discretization of the Poisson equation. On the other hand, in direct methods the memory limitation is restrictive for simulations on a fine grid. Therefore decoupling of the directions by FFT based methods can be advantageous, even if this method implies some limitations in the boundary conditions. We propose a direct fourth-order solver for the Poisson equation which is a combination of a compact finite difference with a sine FFT. The main advantages of our method are fourth-order accuracy, efficiency, the possibility to parallelize and convergence down to zero machine precision over an optimal grid. Other advantages and limitations of the proposed solver are discussed in the paper. A difficulty in numerical simulations of fish swimming is the analysis of fluid/solid interaction, which can be handled by strong or loose coupling according to implicit or explicit time advancement, cf. (Sotiropoulos and Yang, 2014) for a detailed discussion. We use the volume penalization method, known also as Darcy-Brinkmann penalization (Brinkmann, 1947), proposed by Arquis and Caltagirone (1984), Angot et al. (1999) and Khadra et al. (2000), which belongs to the diffuse-interface immersed boundary methods (IBMs). It consists of modeling the immersed body as a porous medium, thus getting rid of the Dirichlet boundary conditions by considering both the fluid and the body as one domain with different permeabilities. So one can consider a rectangular solution domain in which the body is immersed and can even move. The penalization method leads to between first and second order accuracy near the body and is an efficient method in dealing with deformable, moving and rotating bodies immersed in a fluid. A development to deal with rigid bodies colliding with each other in incompressible flows is performed by Coquerelle and Cottet (2008). An extension to include elasticity of the solid interacting with fluid via the volume penalization method is represented by Engels et al. (2013). One advantage of this class of penalization schemes for fluid-structure interaction problems is that it enables the use of time and space adaptivity via multiresolution analysis as recently demonstrated by Gazzola et al. (2014) and Ghaffari et al. (2014). We refer to the review of Mittal and Iaccarino (2005) for a complete classification and description of immersed boundary methods.

In the present work, we will focus on some numerical aspects of efficient turning laws for anguilliform swimmers, a topic which is less studied so far. To this end the geometrically exact theory of nonlinear one-dimensional beams based on guaternions (Boyer et al., 2006) is adapted to the backbone kinematics description. Starting by the code developed by Sabetghadam et al. (2009) we apply compact finite differences to the vorticity stream-function formulation of the Navier-Stokes equations including the penalization term. An efficient direct method is presented for solving the Poisson equation. Thus different numerical aspects of the algorithm like accuracy in space and the error introduced by the penalization method will be examined. The code is developed in FORTRAN and is open access (Ghaffari). The paper is organized as follows. First our methodology including the governing equations, discretization, kinematics of an anguilliform swimmer and the algorithm for fluid interaction with forced deformable bodies will be presented. Next a validation of the algorithm will be carried out, the errors will be assessed and their convergence will be studied. Then the results for swimming and rotation control are reported. Finally, the results will be discussed and some guidelines for future works will be addressed.

#### 2. Methodology

#### 2.1. Governing equations of incompressible flow

The governing equations of incompressible flows are the Navier–Stokes equations. In two-dimensional problems the vorticity and stream-function formulation in comparison to the primitive variable formulation has the advantage that it not only eliminates the pressure, but also ensures a divergence-free velocity field (mass conservation, i.e.,  $\nabla \cdot \mathbf{u} = 0$ ) if the Poisson Eq. (2) is properly satisfied (Bontoux et al., 1978; Roux et al., 1980). One encounters two scalar valued quantities, i.e., the vorticity  $\omega$  and the stream-function  $\psi$ , instead of the velocity vector and the pressure field, thus it makes the computations more efficient. With this formulation, it is possible to use a collocated grid without adding any explicit numerical dissipation, which reduces the arithmetics

considerably, while keeping the expected accuracy. Moreover, there is no need for pressure grid staggering, neither fully nor partially. The boundary conditions of the elliptic Eq. (2) are of Dirichlet type instead of Neumann for Eq. (5), which can lead to a singularity. Therefore we continue with this formulation, but the concepts can also be extended to the primitive variable formulation. By taking the curl of the Navier–Stokes equations, one obtains the vorticity transport equation:

$$\partial_t \omega + (\mathbf{u} \cdot \nabla) \omega = \nu \nabla^2 \omega + \nabla \times \mathbf{F}, \quad \mathbf{x} \in \Omega \in \mathbb{R}^2$$
(1)

where  $\omega(\mathbf{x}, t) = \nabla \times \mathbf{u} = v_x - u_y$  denotes the vorticity component which is normal to the considered two-dimensional plane,  $\Omega$  is the spatial domain of interest, given as an open subset of  $\mathbb{R}^2$ , which can be bounded or unbounded in general,  $\mathbf{u}(\mathbf{x}, t)$  is the velocity field,  $v = \mu/\rho_f > 0$  is the kinematic viscosity of the fluid,  $\rho_f$  is the density and  $\mathbf{F}(\mathbf{x}, t)$  is a source term. For a complete description of a particular problem, the above equation needs to be complemented to describe an initial/boundary value problem (IBVP). The vorticity transport Eq. (1) is parabolic and the velocity components are  $(u, v) = (\partial_v \psi, -\partial_x \psi)$ , with  $\psi$  satisfying the Poisson equation

$$-\nabla^2 \psi = \omega \tag{2}$$

which is an elliptic equation. The penalization (Arquis and Caltagirone, 1984; Angot et al., 1999 and Khadra et al., 2000) term is representative of the immersed body

$$\mathbf{F} = -\eta^{-1}\chi(\mathbf{u} - \mathbf{u}_P) \tag{3}$$

where  $\mathbf{u}_{P}(\mathbf{x}, t)$  is the velocity field of the immersed body. The Navier–Stokes equations are written for unit mass of the fluid, therefore the dimension of the source term **F** is acceleration per unit mass of the fluid, i.e.,  $[LT^{-2}]$ . The penalization parameter  $\eta$  is the permeability coefficient of the immersed body with dimension [T]. The mask (characteristic) function  $\chi$  is dimensionless and describes the geometry of the immersed body

$$\chi(\mathbf{x},t) = \begin{cases} 1 & \mathbf{x} \in \Omega_b \\ 0 & \mathbf{x} \in \Omega_f \end{cases}$$
(4)

where  $\Omega_f$  represents the domain of the fluid and  $\Omega_b$  represents the immersed body in the domain of the solution. The solution domain  $\Omega = \Omega_f \cup \Omega_b$  is governed by the Navier–Stokes equations in the fluid regions and by Brinkmann (1947) law in the penalized regions, when  $\eta \rightarrow 0$ . An equation for the pressure can be derived by applying the divergence operator to the momentum equations and making use of the mass conservation,

$$\nabla \cdot (\nabla p) = -\rho \nabla \cdot [(\mathbf{u} \cdot \nabla)\mathbf{u}] - \frac{\rho}{\eta} \nabla \cdot [\chi(\mathbf{u} - \mathbf{u}_p)]$$
(5)

For fluid/solid interaction problems the simulations start with the body  $\mathbf{u}_P(\mathbf{x}, 0) = 0$ , and fluid at rest, i.e.,  $\omega(\mathbf{x}, 0) = \psi(\mathbf{x}, 0) = 0$  and free-slip boundary conditions are imposed at the surrounding walls  $(\psi|_{\partial \Omega} = \omega|_{\partial \Omega} = \partial p / \partial \mathbf{n}|_{\partial \Omega} = 0)$ .

## 2.1.1. Spatial discretization

In the present investigation an explicit central second order and an implicit fourth order compact finite difference method (Hirsh, 1975) is used for discretization of the spatial derivatives. The advantage of compact methods over explicit finite differences is illustrated in terms of the scaled modified wavenumber  $w = k\Delta x$ in Fig. 1. For a given periodic function  $f(x) = e^{ikx}, x \in [0, 2\pi]$  with known exact derivatives  $f'(x) = ike^{ikx}, f''(x) = -k^2 e^{ikx}$ , a numerical approximation of the derivatives at point  $x_j$  has the form  $f'(x_j) = ik'e^{ikx_j}$  and  $f''(x_j) = -k'^2 e^{ikx_j}$ . The difference between the exact and numerical approximation of the wavenumber is a measure of the discretization error which is purely dispersive for the first derivative and dissipative for the second derivative. if the considered function is periodic and the discretization is central. The scaled modified wavenumbers are computed via different explicit and implicit differentiation methods and are illustrated in Fig. 1 for the first and second derivatives. For the second derivative the scaled modified wavenumbers w'' are compared in Fig. 1(b) with analytical values given by Lele (1992). A good agreement between the numerical approximations of the scaled modified wavenumbers and the analytical values can be observed. It must be noted that the error in terms of the modified wavenumber is not necessarily sensitive to the formal order of the truncation error obtained by Taylor expansion analysis. The desired characteristics of finite difference schemes are better studied by directly optimizing the scheme in Fourier space rather than looking for the lowest truncation error. For example spectral like five-diagonal finite difference schemes designed by Lele (1992) or Kim (2007) are formally fourth order, see Fig. 1(a). Given the values of a function f on a uniformly spaced mesh  $x_i = (i - 1)h$ , for (i = 1, ..., N) with  $h = L_x/(N - 1)$ , following (Hirsh, 1975; Lele, 1992) a fourth-order approximation of the first and second derivatives are obtained by the classical Padé schemes:

$$f'_{i-1} + 4f'_i + f'_{i+1} = 3(f_{i+1} - f_{i-1})/h$$
(6)

$$f_{i-1}'' + 10f_i'' + f_{i+1}'' = 12(f_{i+1} - 2f_i + f_{i-1})/h^2$$
(7)

for i = 2, ..., N - 1, near the boundaries a third-order forward/backward stencil can be used, we refer to Lele (1992) for more details. A



**Fig. 1.** Plots of the scaled modified wavenumber for the first (left) and second (right) derivative, w'(w) and w''(w), respectively, versus the scaled wavenumber  $w = k\Delta x$  for different central finite difference methods presented by Lele (1992) and Kim (2007).

direct solver (based on LU-decomposition) can be applied to the tri/ penta-diagonal system of linear equations along each line. The computational cost of a tridiagonal implicit method is in general three times the one of an explicit method and for a pentadiagonal linear system it is twice that of the tridiagonal one.

#### 2.1.2. Time integration

A classical fourth-order Runge–Kutta (RK4) method is used for time integration of the penalized vorticity transport Eq. (1). By assembling all spatial derivatives in the operator  $k(\omega, \psi)$ , one has

$$\omega^{n+1} = \omega^n + \frac{\Delta t}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$
(8)

where

$$k(\omega,\psi) = -\partial_y \psi \ \partial_x \omega + \partial_x \psi \ \partial_y \omega + \nu \nabla^2 \omega + \partial_x F_y - \partial_y F_x \tag{9}$$

at each time step Eq. (9) must be evaluated four times where Eq. (2) must be solved for updating the stream-function  $(\omega_i = \omega^n + \alpha_i \Delta t k_{i-1} \text{ and } -\nabla^2 \psi_i = \omega_i)$  with  $\alpha_1 = 1/2, \alpha_2 = 1/2$  and  $\alpha_3 = 1$ . Details are given in the time integration part of Algorithm 1. For technical discussions of Runge–Kutta methods we refer to Press et al. (1992). However,  $\Delta t$  is limited by the *CFL* (Courant–Friedrichs–Lewy) condition which implies that

$$\frac{U\Delta t}{\Delta x} \leqslant CFL \approx \frac{\sigma_i}{w'_{\text{max}}} \tag{10}$$

where *U* is an advection velocity (or a phase speed). In the presence of nonlinearity in space more attention must be payed. Viscous terms imply an additional constraint of the form

$$\frac{v\Delta t}{\Delta x^2} \leqslant VSL \approx \frac{\sigma_r}{w''_{\text{max}}}$$
(11)

on the time-step, where  $\sigma_r = 2.9$  and  $\sigma_i = 2.85$  are real and imaginary limits of the stable region in the complex plane, of the RK4 method. Here  $w'_{max} = 1.74$  and  $w''_{max} = 6$  are the maximum values of the scaled modified wavenumbers for the first and second derivatives calculated via the fourth-order Padé scheme, plotted in Fig. 1. It can be seen that  $w'_{max} \in [1, \pi]$  and  $w''_{max} \in [4, \pi^2]$  for the different approximations of the spatial derivatives. For an explicit secondorder discretization we have  $w'_{max} = 1$  and  $w''_{max} = 4$ . Therefore with the use of a high-order method for the spatial discretization smaller time-steps must be used. In the presence of moving bodies the displacement of the moving body must not exceed the grid spacing, i.e.,  $\Delta t \leq \Delta \mathbf{x}/\mathbf{u}_{B}$ . Moreover, by using the explicit penalization method another constraint,  $\Delta t \leq \eta$ , must be respected. Among the four above-mentioned constraints, the smallest  $\Delta t$  must be chosen. In Gazzola et al. (2011) an implicit penalization method (via operator splitting) is used for simulation of fluid-solid interaction problems, the time accuracy is reported to be first-order. This is independent of the time accuracy of the underlying method for time integration of the Navier-Stokes equations. We use explicit penalization, i.e., the penalization term is kept in the right hand side of the vorticity transport Eq. (9). However the accuracy and the larger stability bound of RK4 is still attractive to enhance the time step of the flow solver. The overall accuracy of explicit RK4-penalization is observed to be better than implicit RK4-penalization via operator splitting in the simulations of the fish in forward gait. A rigorous error analysis must be done. However, implicit penalization is unconditionally stable and allows for smaller penalization parameters  $\eta$ . We refer to section the penalization model in Coquerelle and Cottet (2008) for a discussion on the time integration and the appendix of Morales et al. (2014).

## 2.1.3. Fourth-order fast Poisson solver

In solving the incompressible Navier–Stokes equations, an elliptic Poisson equation is frequently encountered which is the

most time consuming part of the algorithm. The common case is the pressure Poisson equation normally used with homogeneous Neumann boundary conditions, for the pressure correction in projection methods. In the vorticity stream-function formulation, Eq. (2) has to be solved with Dirichlet boundary condition for vorticity and stream-function. Free slip ( $\omega = 0$ ) boundary conditions in a closed rectangular domain ( $\psi = 0$ , all around) is applied in all the test cases studied in the present investigation. Numerical tests reveal that there is no significant difference between no-slip and free-slip boundary conditions, in dealing with fluid structure interaction problems, see the discussion in Section 3.2. In the presence of periodic boundary conditions, FFT based direct solvers can be used to efficiently solve the Poisson equation with high accuracy. Even if the flow is not periodic in all directions, like in most of the practical problems, in accordance with the boundary conditions for the elliptic equation (homogeneous Dirichlet/Neumann) sine or cosine FFTs can be used in one or two directions. see Fig. 2 and the discussions by Kim and Moin (1985), Orlandi (2000), Laizet et al. (2010).

We propose a direct fourth-order solver for the Poisson Eq. (2) which is a combination of a compact finite difference with a sine FFT (suitable for imposing free-slip boundary condition at least in one direction). The advantages of our method are fourth-order accuracy, convergence down to machine zero over an optimal grid. compact tridiagonal stencil, possibility of extension to three dimensions, reduced arithmetics and memory usage in comparison to iterative methods. Moreover, the parallelization is straightforward because the operations in different directions are decoupled. Nearly linear strong scaling (speed up) and efficiency is reported by Laizet et al. (2010) for a direct solver by decoupling of the operators in different directions. They introduced a dual domain decomposition (or pencil) method, in which information along a line is accessible for a CPU by alternative decomposition of the domain in three directions. The limitation of our method (in addition to the boundary conditions) is the use of a uniform grid in the direction in which the FFT is applied. When the solver of the parabolic part is based on finite-differences, one uses typically a FDM discretization in one direction without loss of accuracy and efficiency via a direct tridiagonal solver. The advantage of this



**Fig. 2.** (top) The trigonometric basis functions for a complex FFT of a periodic function. (mid) The trigonometric basis functions for a *sine* FFT of a function with homogeneous Dirichlet boundary conditions. (down) The trigonometric basis functions for a *cosine* FFT of a function with homogeneous Neumann boundary conditions. Picture from Press et al. (1992).

approach is the possibility of applying general boundary conditions in one direction and using a refined mesh. For the second-order version of this solver we refer to Orlandi (2000). For deriving a compact fourth-order collocated discretization (with  $N_x \times N_y$  grid points) of the Poisson equation  $-\nabla^2 \psi = \omega$ , in the x-direction

$$\frac{\partial^2 \psi}{\partial x^2} = \delta_x^2 \psi - \frac{\Delta x^2}{12} \frac{\partial^4 \psi}{\partial x^4} + O(\Delta x^4)$$
(12)

can be used, where  $\delta_x^2$  represents a central second-order approximation of the second derivative. Replacing  $\psi_{xx}$  by (12) in the Poisson Eq. (2) we obtain

$$\left(\delta_x^2 - \frac{\Delta x^2}{12}\frac{\partial^4}{\partial x^4} + \partial_{yy}\right)\psi = -\omega \tag{13}$$

Because of the presence of the  $\Delta x^2$  factor behind the fourth-order derivative, this term cannot be dropped and must be evaluated with second-order accuracy. Therefore, the whole approximation yields fourth-order accuracy. The fourth-order derivative can be evaluated by using the original Poisson equation  $-\nabla^2 \psi = \omega$ , and successive differentiation with respect to x (i.e.,  $\partial_{xx} \partial_{xx} \psi = -\partial_{xx} \partial_{yy} \psi - \partial_{xx} \omega$ ). Replacing  $\partial_{xx}$  by  $\delta_x^2$ , we find

$$\left(\delta_x^2 + \frac{\Delta x^2}{12}\delta_x^2\partial_{yy} + \partial_{yy}\right)\psi = -\omega - \frac{\Delta x^2}{12}\delta_x^2\omega \tag{14}$$

By applying a Fourier transform in the *y*-direction on Eq. (14) and replacing second derivatives  $\partial_{yy}\psi$  by  $-k_y^2\hat{\psi}$  in Fourier space, we have

$$\left(\delta_x^2 - \frac{\Delta x^2}{12}\delta_x^2 k_y^2 - k_y^2\right)\hat{\psi} = -\hat{\omega} - \frac{\Delta x^2}{12}\delta_x^2\hat{\omega}$$
(15)

Usually the exact wavenumber is replaced by the modified wavenumber  $k_y^2$  which permits to adapt the spectral approximation of the second derivative with the considered finite difference method (Orlandi, 2000). For a fourth-order explicit finite difference discretization, an analytical relation for the scaled modified wavenumber of the second derivative is given by Lele (1992) as follows

$$k_y^{\prime 2} = \frac{1}{\Delta y^2} \left[ \frac{8}{3} \left( 1 - \cos\left(\frac{k_y \pi}{N_y}\right) \right) - \frac{1}{6} \left( 1 - \cos\left(\frac{2k_y \pi}{N_y}\right) \right) \right]$$
(16)

Comparison with numerical values in Fig. 1(b) confirms that Eq. (16) is exact. The final tridiagonal system to be solved in Fourier space for each wavenumber of  $\psi$  in the *y*-direction is

$$\beta\hat{\psi}_{i+1,m} - (2\beta + k_y^2)\hat{\psi}_{i,m} + \beta\hat{\psi}_{i-1,m} = -(\hat{\omega}_{i+1,m} + 10\hat{\omega}_{i,m} + \hat{\omega}_{i-1,m})/12$$
(17)

for  $i = 2, ..., N_x - 1$ , where  $\beta = \Delta x^{-2} - k_y^2/12$ . In summary, first a one-dimensional direct-FFT of the forcing function is performed along all the lines,  $j = 1, ..., N_y$  in the *y*-direction. Next for each line in the x-direction the tri-diagonal system (17) must be solved to find the solution  $\psi$  in wavenumber space. Finally an inverse-FFT of the solution is performed line by line in the y-direction. For the real data with zero value at the boundaries (homogeneous Dirichlet, i.e.,  $\psi = \omega = 0$ , corresponding to free-slip boundary conditions), the natural Fourier transform to use is the sine transform, see Fig. 2 from Press et al. (1992). The direction of FDM and FFT can be changed to consider no-slip boundary conditions in the y-direction. In order to take into account inflow/outflow boundary conditions the mean flow must be reduced from the total velocity field  $\mathbf{u} = \mathbf{U} - \mathbf{U}_{\infty}$  in the vorticity transport Eq. (1) to impose  $\psi = 0$  at the boundaries. This is equivalent to move the grid with  $\mathbf{U}_{\infty}$  and writing the Navier-Stokes equations in a moving reference frame for the perturbed velocity field **u**, instead of a Galilean inertial frame (Rossinelli et al., 2010).

For validation of the developed fourth-order Poisson ( $\nabla^2 u = f$ ) solver an exact solution (see Fig. 3) with Dirichlet boundary

conditions is considered. The solution is obtained with  $N^2 = 33^2$ grid points via the fourth-order direct solver and is illustrated in Fig. 3(a). The corresponding error contours E(x, y) = |u(x, y)| $-u_{\text{exact}}(x,y)|, (x,y) \in \Omega$  in comparison with the exact solution are illustrated in Fig. 3(b). The convergence of different errors for second and fourth order direct Poisson solvers are illustrated in Fig. 3(c). The CPU-time scaling in log-log scale for different methods including, second and fourth order direct, multi-grid (MG) and point successive over relaxation (PSOR) with red-black sweeper are compared in Fig. 3(d). The cost of computations (in terms of CPU-time) of direct and multi-grid methods are proportional to the number of grid points ( $N^2$  in two-dimensions) but for iterative methods this is increasing exponentially  $CPU_{time} = 5 \exp(0.01 \text{ N})$ , which is very restrictive for computations with fine grids. The multi-grid solver developed by Pakneiad (2010) is the best in terms of CPU-time, but by optimizing the FFT the proposed direct method can do better. The memory allocation of the multi-grid solver developed by Pankejad (2010) is restrictive with fine grids, the finest possible resolution on the available machine is 1024<sup>2</sup>. The resolution of the finest possible grid of the proposed solver on the available machine is 4096<sup>2</sup>. From parallelization view point the multi-grid solver is the most difficult but the iterative methods are the easiest to be parallelized. The proposed direct method can be parallelized by the pencil rotation method as done by Laizet et al. (2010) for a direct method, where nearly linear strong scaling (speed up) is reported.

#### 2.2. Algorithm of fluid-structure interaction

A summary of the algorithm for the fluid interaction with a forced deformable body, applied to fish-like swimming, is given in Algorithm 1. Each part of the algorithm is discussed in detail in the following.

## 2.2.1. Kinematics of the fish

The swimming mechanism in the majority of anguilliform and carangiform fishes can be modeled with a sinusoidal wave enveloped by a profile, lying over the backbone of the fish, which moves from the head to the tail. The geometrically exact theory of nonlinear beams, is developed by Simo (1985) and extended for fish vertebral by Boyer et al. (2006). In this theory, the beam is considered as a continuous assembly of rigid sections of infinitesimal thickness, i.e., a one-dimensional Cosserat medium. We summarize the kinematics of the fish backbone in three dimensions, for interested readers and future developments, but all the cases in this paper are limited to two dimensions. Following Boyer et al. (2006), El Rafei et al. (2008) and Belkhiri (2013) starting with the head orientation, position and velocities as boundary conditions, the kinematics of the backbone for anguilliform fishes can be determined by integration along the arclength  $\xi \in [0, l_{\text{fish}}]$ . The variation of the orientation along the backbone in terms of quaternions are obtained by

$$\frac{\partial Q}{\partial \xi} = \frac{1}{2} M^{\vee}(K) Q \tag{18}$$

where  $Q = (\cos \frac{\phi}{2}, a_x \sin \frac{\phi}{2}, a_y \sin \frac{\phi}{2}, a_z \sin \frac{\phi}{2})^T$  are unit vectors, normalized  $(q_0^2 + q_1^2 + q_2^2 + q_3^2)^{1/2} = 1$  quaternions that represent the body frame orientation with respect to the inertial frame and  $M^{\vee}(K)$  is an anti-symmetric tensor

$$M^{\vee}(K) = \begin{bmatrix} 0 & -k_1 & -k_2 & -k_3 \\ k_1 & 0 & k_3 & -k_2 \\ k_2 & -k_3 & 0 & k_1 \\ k_3 & k_2 & -k_1 & 0 \end{bmatrix}$$
(19)



**Fig. 3.** (a) Solution of the Poisson equation  $(\nabla^2 u = f)$  with the forcing term  $f(x,y) = -(n^2\pi^2/L_y^2 + 1)\cos(x)\sin(n\pi y/L_y)$ ,  $(x,y) \in \Omega = [0, 2\pi]^2$  and Dirichlet boundary conditions  $u_b(x,y) = \cos(x)\sin(n\pi y/L_y)$ ,  $(x,y) \text{ on } \partial\Omega$  (n = 3), via the proposed direct fourth-order solver. (b) The corresponding error contours  $E(x,y) = |u(x,y) - u_{\text{exact}}(x,y)|$ ,  $(x,y) \in \Omega$  in comparison with exact solution. (c) Error analysis for direct Poisson solvers via second and fourth order compact methods. (d) CPU-time scaling of different iterative (Multi-Grid/Point Successive Over Relaxation) and direct methods.

where  $k_2$  and  $k_3$  in  $K = (k_1, k_2, k_3)^T$  stand for the fish backbone transversal curvature and  $k_1$  represents the rate of rotation (twist) of the section around the backbone with the normal aligned with the  $\xi$ -direction. The geometry  $R = (x, y, z)^T$  in the Galilean reference frame is given by

$$\frac{\partial R}{\partial \xi} = Rot(Q)\Gamma \tag{20}$$

where  $\Gamma = (\gamma_1, \gamma_2, \gamma_3)^T$  represents the local transversal shearing of the sections whose first component is the stretching rate along the  $\xi$ -direction. The rotation matrix in terms of the quaternions is then given by

$$Rot = 2 \begin{bmatrix} q_0^2 + q_1^2 - \frac{1}{2} & q_1q_2 - q_0q_3 & q_1q_3 + q_0q_2 \\ q_1q_2 + q_0q_3 & q_0^2 + q_2^2 - \frac{1}{2} & q_2q_3 - q_0q_1 \\ q_1q_3 - q_0q_2 & q_2q_3 + q_0q_1 & q_0^2 + q_3^2 - \frac{1}{2} \end{bmatrix}$$
(21)

The variation of mean linear  $V = (v_1, v_2, v_3)^T$  and angular  $\Omega = (\omega_1, \omega_2, \omega_3)^T$  velocities in the local frame, i.e., the frame attached to the body are given by

$$\frac{\partial}{\partial \xi} \begin{bmatrix} V\\ \Omega \end{bmatrix} = -\begin{bmatrix} K^{\vee} & \Gamma^{\vee}\\ 0 & K^{\vee} \end{bmatrix} \begin{bmatrix} V\\ \Omega \end{bmatrix} + \begin{bmatrix} \dot{\Gamma}\\ \dot{K} \end{bmatrix}$$
(22)

where (·) represents the time derivative, ( $\lor$ ) stands for the anti-symmetric matrix constructed from a given vector, e.g.,

$$K^{\vee} = \begin{bmatrix} 0 & -k_3 & k_2 \\ k_3 & 0 & -k_1 \\ -k_2 & k_1 & 0 \end{bmatrix}$$
(23)

The acceleration can also be deduced from the time derivative of Eq. (22). For more details we refer to Boyer et al. (2006), El Rafei et al. (2008), and Belkhiri (2013). To find the velocities in the frame attached to the body from the velocities  $V_G$  in the Galilean reference frame and inverse,

$$(v_1, v_2, v_3)^T = Rot^T (v_x, v_y, v_z)^T$$
 (24)

can be used. By considering  $N(1,...,N_{points})$  discrete points on the fish backbone, Eqs. (18), (20) and (22) altogether must be integrated in space by a proper numerical method ( $N_{eq} = 13$  in 3D). We use a fourth-order Runge–Kutta method for integration and comparisons with a first-order Euler method show that RK4 is more precise especially when the number of points along the fish backbone is less than  $N_{points} = 30$ .

#### 2.2.2. Lagrangian structured grid

The first choice to start the parameterization of the swimmer body is a symmetric shape. The geometry of a two-dimensional swimmer can be characterized by the half width  $w(\xi)$  of the body along its arclength (midline)  $\xi \in [0, l_{fish}]$ . Following the work of Kern and Koumoutsakos (2006) and Carling et al. (1998), the half width  $w(\xi)$  is defined as

$$w(\xi) = \begin{cases} \sqrt{2w_h\xi - \xi^2} & 0 \leqslant \xi < s_b \\ w_h - (w_h - w_t) \left(\frac{\xi - s_b}{s_t - s_b}\right)^2 & s_b \leqslant \xi < s_t \\ w_t \frac{l_{\text{fish}} - \xi}{l_{\text{fish}} - s_t} & s_t \leqslant \xi \leqslant l_{\text{fish}} \end{cases}$$
(25)

where  $l_{\text{fish}}$  is the body length,  $w_h = s_b = 0.04 l_{\text{fish}}$ ,  $s_t = 0.95 l_{\text{fish}}$  and  $w_t = 0.01 l_{\text{fish}}$ . The shape of the fish before deformation is plotted in Fig. 4. In the mid part of the fish a linear function can also be used as done by Gazzola et al. (2011). A structured grid formed by normal to backbone lines with thickness given by (25) covers the body. Two examples are shown in Figs. 6 and 7. The velocity components of each point on the Lagrangian grid  $V_{\text{shape}}$  with (*I*,*J*) indexes are given by

$$\vec{V}_{\text{shape}}(I,J) = \vec{V}_{\text{BN}}(I) + \vec{\Omega}_{\text{BN}}(I) \times \mathbf{r}(I,J)$$
(26)

where  $\overline{V}_{\text{BN}}$  and  $\overline{\Omega}_{\text{BN}}$  are the linear and angular velocities of the backbone, respectively given by Eq. (22). The radius ( $|\mathbf{r}| < w$ ) is measured over the transversal lines of the structured grid normal to the backbone. Fig. 6 shows an example of the Lagrangian grid covering the fish after deformation in which the corresponding velocities of each point are also illustrated. The information of the Lagrangian structured grid covering the deformable body must be transfered to the Eulerian–Cartesian grid by interpolation to find  $\chi(i,j)$  and  $\mathbf{u}_p(i,j)$ . To determine  $\chi(i,j)$  on the Eulerian grid whose first point  $(x, y)_{(1,1)} = (0, 0)$  is located at the origin, the coordinates of each point on the Lagrangian grid  $\mathbf{X}_{\text{shape}}(I,J)$  are divided by  $\Delta x$  and  $\Delta y$ . After applying a correction to the integer part of the results they give the indexes (i,j) of the mask function  $\chi$  on the Eulerian grid for which  $\chi = 1$  is assigned.

After determining the mask function  $\chi(i,j)$  on the Eulerian grid, following (Forestier, 2000; Minguez, 2008; Kolomenskiy and Schneider, 2009) the mask function is mollified by the Shuman (1957) filter

$$\bar{\chi}_{ij} = (2\chi_{ij} + \chi_{i+1,j} + \chi_{i-1,j} + \chi_{i,j+1} + \chi_{i,j-1})/6$$
(27)

which is equivalent to raised *cosine* filter in Fourier space, we refer to Pasquetti et al. (2008) for more details. The effect of smoothing with Eq. (27) is demonstrated in Fig. 5 over a box function  $\chi(x) = 1, x \in [4.5, 5.5]$  which is represented by a red-solid line. The mollified box function  $\bar{\chi}$  is plotted with a green-dashed line. An example of the transfered geometry  $\chi$  to the Eulerian grid, after smoothing by Eq. (27) is illustrated in Fig. 7, where the boundary of the Lagrangian grid is also added over the contours of the smoothed mask function  $\bar{\chi}$ . It can be seen that it lies between maximum and minimum values of the mask function. The smoothing of the mask function  $\bar{\chi}$  reduces the stiffness of the vorticity transport equation thus larger time steps can be used. Moreover, it increases the regularity of the pressure and velocity field. Furthermore, in



Fig. 4. Shape of the fish given by Eq. (25) before deformation.



**Fig. 5.** A box function  $\chi(x) = 1, x \in [4.5, 5.5]$  (red-solid) and mollified box function  $\bar{\chi}$  (green-dashed) with Eq. (27). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

dealing with moving boundaries the oscillations of the hydrodynamic coefficients are less, when the mask function is smoothed. However, without filtering of the mask function, by applying the explicit second-order finite difference method the solution converges. But by using fourth and higher order discretizations smoothing of the mask function is necessary, if not Gibbs phenomenon or divergence in the solution are expected, especially when the mask function  $\chi$  is moving. Note also that the interpolated velocity field  $\mathbf{u}_p$  over the Eulerian grid is not divergence-free, we refer to Gazzola et al. (2011) for a complete theoretical and numerical discussion about this subject. In the present investigation, we do not consider this issue under the assumption that the body is slender. We use a two-dimensional linear interpolation, to transfer the velocities of the Lagrangian grid given by Eq. (26) to the Eulerian grid, by considering

$$f(x,y) = axy + bx + cy + d \tag{28}$$

and using the four nearest points of the Lagrangian grid Eq. (28) leads to a 4 × 4 linear system for each point with  $\bar{\gamma} \neq 0$  over the Eulerian grid. To determine the unknowns the system is solved by a direct method, i.e., Gauss-Jordan elimination (Press et al., 1992). For all points in the interior of the fish we have  $\bar{\chi}(i, j) = 1$  on the Eulerian grid. For the points of the Eulerian grid in which  $\bar{\chi} = 1$ the four nearest points of the Lagrangian grid are used to find the coefficients of the linear system formed by (28). In some points of the Eulerian grid, due to mollifying of the mask function  $\chi$  by Eq. (27) we have  $0 < \bar{\chi} < 1$ , therefore the interpolation automatically becomes an extrapolation. Some points are completely outside of the original Lagrangian shape. Just one point at the start and the end singularities of the Lagrangian grid can be used to find the penalized velocities  $\mathbf{u}_{p}$  over the Eulerian grid, if not the interpolation matrix will have a zero determinant (singular). However, the start and the end points are used in the determination of the mask function. An example of the interpolated velocity components on the Eulerian grid is illustrated in Fig. 7.

The spacing of the grid points on the Lagrangian grid must be fine enough in comparison to  $\Delta X$  and  $\Delta Y$  to accurately represent the deformation of the body over the Eulerian grid, i.e.,  $\Delta X \leq \Delta x$ . However, the ratio  $\Delta x/\Delta X$  cannot be determined exactly because  $\Delta X$  are varying even if  $\Delta x$  and  $\Delta y$  are fixed. Nevertheless in Figs. 6 and 7 the Lagrangian and the Eulerian grids are schematically illustrated for a fine and a coarse Lagrangian grid. If the Lagrangian grid is very fine, the computational effort in the procedure of evolving the mask function  $\chi$  and determining the corresponding velocities



**Fig. 6.** (left) The Lagrangian structured grid  $(Im_b \times Jm_b = 121 \times 19)$  superimposed on the Eulerian one. (right) The Lagrangian structured grid covering the fish after deformation and the corresponding velocity vectors of each point, colored by absolute velocity  $\sqrt{u^2 + v^2}$ .



**Fig. 7.** (left) Interpolated mask function  $\bar{\chi}$  from a Lagrangian grid ( $Im_b \times Jm_b = 61 \times 9$ ) where  $\Delta \mathbf{x} < \Delta \mathbf{X}$ , as can be seen insufficient resolution in the Lagrangian grid results in  $\bar{\chi} = 1$  in very few points inside the fish. (right) Boundary of the Lagrangian grid (black line), interpolated smoothed mask function  $\bar{\chi}$  (colored isolines), as can be seen the boundary of the Lagrangian grid ( $Im_b \times Jm_b = 121 \times 19$ ) lies between minimum and maximum values of the mask function. The velocity components forming a jet at the tail while the fish is turning. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

 $\mathbf{u}_{n}$  will increase. The additional cost does not lead to considerable enhancement in the accuracy of the mask function  $\gamma$  or the interpolated velocities of the body  $\mathbf{u}_{p}$  over the Eulerian grid. However, a very fine Lagrangian grid leads to singular matrices in the interpolation procedure via Eq. (28) because the four points chosen for interpolation will be very close. For a very fine Lagrangian grid, zero order interpolation must be used, i.e., the velocities of the nearest point on the Lagrangian grid must be assigned to the Eulerian grid. On the other hand if a very coarse Lagrangian grid is used the information of the body will be lost. Especially the rotational velocity field due to the deformation of the body which has a great importance in the accuracy of the simulations, will be inaccurate and even divergence of the simulations is expected. Moreover, the values of the mask function will not reach the value one inside the fish with insufficient resolution of the Lagrangian grid, see Fig. 7. The geometry also will be not accurate near singular points (like the tail) or boundaries with high curvature (like the head). The hydrodynamic coefficients can also be inaccurate whenever a coarse grid is used for the Lagrangian grid. An optimal value is proposed for the size of the Lagrangian grid;

$$\frac{\Delta \mathbf{x}}{10} < \Delta \mathbf{X} < \frac{\Delta \mathbf{x}}{2}$$

## 2.2.3. Hydrodynamic coefficients evaluation

With the use of the volume penalization method (Arquis and Caltagirone, 1984; Angot et al., 1999; Khadra et al., 2000) the hydrodynamic forces and the torque acting on the body, which are usually evaluated via surface integrals of the stress tensor  $\sigma(\mathbf{u}, p) = \mu (\nabla \mathbf{u} + (\nabla \mathbf{u})^T) - p\mathbf{I}$ , can be computed directly by integrating the penalized velocity over the considered volume (surface in two-dimensions). Thus the hydrodynamic forces in [Newton] are given by

$$\mathbf{F}^{\star} = \oint_{\partial \Omega_{s}} \boldsymbol{\sigma} \cdot \mathbf{n} \, \mathrm{d}l = \lim_{\eta \to 0} \frac{\rho_{f}}{\eta} \int_{\Omega_{s}} \chi(\mathbf{u} - \mathbf{u}_{B}) \, \mathrm{d}s + \rho_{f} S_{pen} \ddot{\mathbf{X}}_{ref}$$
(29)

for the unit mass  $(m = \rho_f S_{pen})$  of the fluid. By definition  $\mathbf{F} = \mathbf{F}^{\star}/m$ , we have

$$\mathbf{F}_{H} \approx \frac{1}{\eta S_{pen}} \int_{\Omega_{s}} \chi(\mathbf{u} - \mathbf{u}_{B}) \, \mathrm{d}s + \ddot{\mathbf{X}}_{ref} \tag{30}$$

The torque in two-dimensions can be evaluated by

$$M_{ref} = \oint_{\partial \Omega_s} \mathbf{r} \times \boldsymbol{\sigma} \cdot \mathbf{n} \, \mathrm{d}l$$
  
= 
$$\lim_{\eta \to 0} \frac{\rho_f}{\eta} \int_{\Omega_s} \chi \mathbf{r} \times (\mathbf{u} - \mathbf{u}_B) \, \mathrm{d}s + \frac{\rho_f}{\rho_b} J_{ref} \ddot{\theta}_{ref}$$
(31)

in  $[N \cdot m]$ , where  $\mathbf{r} = (\mathbf{x} - \mathbf{X}_{ref})$  is the distance vector from the reference point,  $J_{ref} = \int \mathbf{r}^2 dm$  is the polar moment of inertia  $(J = I_{zz})$  taken around the reference point which can be the center of gravity (cg) of the immersed body,  $\mathbf{n}$  is the unit outward vector normal to  $\partial \Omega_s$ ,  $\theta$  is the angle of rotation with respect to the reference point. The dots denote derivatives with respect to time and  $S_{pen}$  is the surface of the penalized area.

#### 2.2.4. Denoising of the hydrodynamic coefficients

In dealing with fluid/solid interaction problems, the oscillations of the hydrodynamic forces and the torque during successive iterations computed from Eqs. (30) and (31) cause some troubles in correctly predicting the accelerations. The hydrodynamic forces and the torque acting on the body are used to evaluate the linear and angular accelerations which in turn have an impact on the predicted velocity vector and the trajectory of the solid. The oscillations are due to the nature of the penalization method see the discussion by Kolomenskiy and Schneider (2009), insufficient resolution, the approximative nature of Eqs. (30) and (31). The oscillations are like a noise and lead to invalid results and even divergence of the simulations. An efficient method to eliminate them is to apply a low-pass filter like exponential smoothing which is usually used in denoising of time series. This filter is used by Kern and Koumoutsakos (2006) to denoise the hydrodynamic forces and the torque. Simple exponential smoothing does not perform well if there is a trend in the data (e-Handbook of Statistical Methods, 2012). In such situations, several methods were devised like second-order (double) exponential smoothing (Holt, 1957)

$$\hat{F}^n = \alpha F^n + (1 - \alpha)(\hat{F}^{n-1} + b^{n-1}) \quad n = 3, 4, \dots$$
 (32)

$$b^{n} = \beta(\hat{F}^{n} - \hat{F}^{n-1}) + (1 - \beta)b^{n-1} \quad (\alpha, \beta) \in [0, 1]$$
(33)

where  $\hat{F}^1 = F^1$ , for n = 2 one can use Eqs. (32) and (33) with  $\alpha = \beta = 1$ . Then  $\alpha = 1 - (1 - \delta)^2$  and  $\beta = \delta^2 / \alpha$  can be used in which  $\delta$  is a small band. Imposing b = 0 in Eq. (32) leads to first-order filtering. A comparison of the first and second order filtering of the hydrodynamic coefficients is done in the Section 3.3 for the falling ellipse in the fluttering regime, see Fig. 16. According to our experience  $\delta = 10^{-3}$  performs well for denoising of the hydrodynamic coefficients of the moving bodies. However  $\delta = 10^{-3}$  has a strong damping effect, larger values, i.e.,  $\delta = 5 \times 10^{-3}$  have less damping effect but there is a risk of divergence in the simulations. A sensitivity analysis must be done for each test case, see also the discussion of the results in Sections 3.3 and 3.4. By using a projection/fictitious domain approach the instantaneous linear and angular velocities of the fish can be recovered directly from the flow (Gazzola et al., 2011), avoiding the evaluation of the hydrodynamic coefficients and thus the use of denoising schemes. Denoising of the hydrodynamic coefficients needs a preliminary sensitivity analysis and can be considered as a drawback for the present methodology.

#### 2.2.5. Body dynamics

The dynamics of an arbitrary solid or deformable body moving in a viscous incompressible fluid is governed by Newton's second law

$$\Sigma(\mathbf{F}_H + \mathbf{F}_G) = m \mathbf{X}_{ref} \tag{34}$$

where the applied forces can be split into two components; the hydrodynamic forces  $\mathbf{F}_H$  and the forces due to gravity  $\mathbf{F}_G = S_{pen}(\rho_b - \rho_f)\mathbf{g}$ . Newton's law can be integrated directly to give the position of the center of gravity as a function of time. Holding **F** constant over the discrete physical time step  $(t^n, t^{n+1})$  yields

$$\Delta \mathbf{X}_{ref} = \frac{1}{2} \frac{\mathbf{F}^n}{m} \Delta t^2 + \mathbf{V}^n \Delta t \tag{35}$$

and  $V^{n+1} = V^n + \ddot{X}\Delta t$ . The rotational motion is described by Euler's equation

$$\Sigma M_{ref} = \frac{d}{dt} (J_{ref} \dot{\theta}) \tag{36}$$

where *M* is the applied torque around the reference point. If the reference point does not coincide with the center of gravity (cg) the torque due to the gravity force (buoyancy) must be added to  $\Sigma M_{ref}$  in Eq. (36). In the presence of the body forces, choosing (cg) as the reference point can simplify the evaluation of the exerted torque, i.e., only the torque due to the hydrodynamic forces  $\mathbf{F}_{H}$ , must be integrated around the reference point. In the present investigation, the center of gravity is choosen as the reference point in the simulations of the falling cylinder and ellipse. However, in the simulations of the swimming fish  $(\rho_b = \rho_f)$  the buoyancy is equal to zero and thus plays no role. Choosing the head as the reference point can simplify the integration of the backbone kinematics via Eqs. (18), (20) and (22), without evaluation of the torque due to buoyancy. Time integration of Eq. (36) regardless of changes in the moment of inertia and  $M_{ref}$ , yields the new angle of the body with respect to a given reference

$$\Delta\theta = \frac{1}{2}\ddot{\theta}^n \Delta t^2 + \dot{\theta}^n \Delta t \tag{37}$$



Fig. 8. Flowchart of the fluid/solid interaction (FSI) algorithm.

where  $\ddot{\theta} = M/I$  and  $\dot{\theta}^{n+1} = \dot{\theta}^n + \ddot{\theta}\Delta t$  (the dot denote derivation with respect to time). Eqs. (35) and (37) describe a motion with three degrees of freedom (3DOF) for the considered body. In these equations second-order terms can be eliminated as done in Gazzola et al. (2011) but we keep these terms. Eqs. (30) and (31) provide the exerted forces and the torque necessary to integrate the system of ODEs formed by Eqs. (34) and (36). Denoising of the hydrodynamic coefficients is done according to Eq. (32). Appropriate initial conditions are necessary. In the present computations we use a first-order scheme for time integration of the dynamics equations which seems to be adequate because of the error introduced by the volume penalization method which is between first and second order. The same time integration method is also used in Kolomenskiv and Schneider (2009) and Gazzola et al. (2011) for the dynamics of the body where the volume penalization is also used. A summary of the algorithm for the fluid/structure interaction is given in Algorithm 1. The flowchart is illustrated in Fig. 8.

Algorithm 1. Fluid/structure interaction

- 1. START FROM AN INITIAL CONDITION
- 2. BODY KINEMATICS
  - (a) (Just for the fish) Create Eel's backbone by integrating Eqs. (18), (20) and (22)
  - (b) (Just for the fish) Cover the shape by a Lagrangian structured grid & compute velocities at each point with Eq. (26)
  - (c) Compute the mask  $\chi(i,j)$  and smooth it by Eq. (27)
  - (d) Compute the moment of inertia *J* around the reference point
  - (e) Compute the velocity components of the body  $u_p(i,j), v_p(i,j)$  on the Eulerian grid by interpolation (Lagrange  $\rightarrow$  Euler)

3. TIME INTEGRATION OF FLOW FIELD VIA RK4

(a) 
$$\omega_0 = \omega^n$$
 ,  $\psi_0 = \psi^n$ 

For *i* = 1, 2, 3

(b) Compute  $k_i(\omega, \psi)^{i-1}$  from Eq. (9)

(c) 
$$\omega_i = \omega^n + \alpha_i \Delta t k_i$$

- (d) Solve Eq. (2);  $-\nabla^2 \psi_i = \omega_i$  for updating (u, v)
- End For
- (e) Compute  $k_4(\omega_3, \psi_3)$  from Eq. (9)
- (f) Update vorticity from Eq. (8)
- (g) Solve Eq. (2);  $-\nabla^2 \psi^{n+1} = \omega^{n+1}$
- 4. Solve for the body dynamics
  - (a) Compute the hydrodynamic coefficients of the body from Eqs. (30) and (31)
  - (b) Denoise the coefficients by Eq. (32)
  - (c) Compute the displacements from Eq. (35)
- (d) Compute the rotation from Eq. (37)
- 5. WRITE NECESSARY DATA TO FILE
- 6. If  $T < T_{end}$ , Go to step 2
- **7.** End

## 3. Validation

In this section first the spatial error of the solver including the penalization term is verified using a Taylor–Couette flow for which an analytical solution is available. Then the ability of the algorithm for dynamical analysis of falling bodies, due to terrestrial gravity field, in a quiescent fluid is examined. Finally, a test case of fish swimming in forward gait is compared with the results of Gazzola et al. (2011).

#### 3.1. Spatial convergence for Taylor-Couette flow

For a rigorous study of the error due to the penalization term added to the Navier-Stokes equation in vorticity and stream-function formulation an exact solution is necessary. The Taylor-Couette configuration is a good choice, first and foremost, because of known Dirichlet boundary conditions everywhere, and secondly, because of the presence of curved walls contrary to other existing analytical solutions usually available for Cartesian domains which thus coincide with the underlying Cartesian grid used to discretize the governing equations. Although the solver is adapted to a Cartesian domain the mask function which represents the penalized area for the Taylor-Couette flow is curved (see Fig. 9) as it is the case for flow around an ellipse or complex geometries which will be considered in the following. Here an explicit second-order finite difference method is used for discretization of the governing equations including the curl of the penalization term  $\nabla \times \mathbf{F}$ . Taylor-Couette flow (Taylor, 1923) consists of a viscous fluid confined between two concentric cylinders with radii (R1, R2) in rotation with different angular velocities ( $\Omega_1, \Omega_2$ ). For Taylor numbers  $Ta = R_1(\Omega_2 - \Omega_1)^2 (R_2 - R_1)^3 v^{-2}$  below the critical value  $Ta_c \approx 1708$ , the flow is steady and purely azimuthal, i.e.,  $u_z = u_r = 0$ . This state is known as circular Taylor–Couette flow and for which an analytical solution exists (Monin and Yaglom, 1971). The solution is given in cylindrical coordinates, the azimuthal velocity is

$$u_{\theta}(r) = Ar + B/r, \quad (r,\theta) \in [R_1,R_2] \times [0,2\pi]$$

where  $A = (\Omega_2 R_2^2 - \Omega_1 R_1^2)/(R_2^2 - R_1^2)$  and  $B = R_1^2 R_2^2 (\Omega_1 - \Omega_2)/(R_2^2 - R_1^2)$  are known. The vorticity between the two cylinders is constant  $(\omega_z = 2A)$  and the stream-function is given by  $\psi(r) = -Ar^2/2$   $-B\ln(r) + c_0$  where  $c_0$  must be determined with respect to an arbitrary reference point. By using the volume penalization method, the velocity components must be enforced in the penalized regions from known angular velocities (i.e.,  $\Omega_1$  and  $\Omega_2$ ),

$$u_{\theta}(r) = r\Omega, \quad (r, \theta) \in [0, R_1] \cup [R_2, R_{max}] \times [0, 2\pi]$$

The vorticity inside the rotating regions is constant and is equal to twice of the domain angular velocity ( $\omega_z = 2\Omega$ ) and the streamfunction is given by,  $\psi(r) = \Omega/2r^2 + c$ , where *c* must be determined



**Fig. 9.** Schematic representation of the penalized unit square domain for modeling the Taylor–Couette flow with the volume penalization method ( $\chi = 0$  represents the fluid domain and  $\chi = 1$  is imposed in the solid domains). The radius of the inner cylinder is  $R_1 = 0.15$  and that of the outer cylinder is  $R_2 = 0.4$ . The angular velocity of the inner cylinder is  $\Omega_1 = 0.2$  and that of the outer is equal to zero, the kinematic viscosity is set to  $\nu = 0.01$ , thus resulting in  $Ta = R_1(\Omega_2 - \Omega_1)^2(R_2 - R_1)^{3}\nu^{-2} \approx 1$ .

for each domain in accordance with  $c_0$ . A unit square domain is considered as the solution domain, the time-step of the RK4 method is calculated by the constraints presented in the Section 2.1.2 and the kinematic viscosity is fixed to v = 0.01. The radii are chosen  $R_1 = 0.2$  and  $R_2 = 0.4$ , respectively. At t = 0 the fluid domain is at rest and the inner-cylinder is set into movement with a fixed angular velocity ( $\Omega_1 = 0.2$ ) while the angular velocity of the outer cylinder is kept equal to zero ( $\Omega_2 = 0$ ). The Taylor number for this configuration (Ta = 0.64) is below the critical value, thus the flow is purely azimuthal. The  $L_1$ -error  $||u^{\text{exact}} - u^{\text{num}}_n||$  for u which is the x-component of the considered velocity field, is calculated for different penalization parameters  $\eta$  and resolutions (N in x and y directions). The simulations are carried out until a steady state is reached, so that the error is independent of the time discretization. The simulations are stopped when the time  $t_{end} = 10$  is reached. Original and mollified mask function at the midline y = 0.5 are illustrated in Fig. 10(a), comparison of the computed vorticity  $\omega$ . stream-function  $\psi$  and the *u* velocity component with exact solutions for N = 128 grid points in each direction are plotted in Fig. 10(a) and (b). The convergences of the  $L_1$ -error of u versus the grid resolution, for different penalization parameters  $\eta$  are shown in Fig. 10(c), where between first and second order convergence can be seen. Suppose  $u_{\eta}^{\text{num}}$  denotes the numerical solution of the penalized equation, for quantifying the numerical error of  $u_{\eta}^{\text{num}}$  compared to  $u^{\text{exact}}$  (the solution to the original Navier–Stokes problem), the error can be estimated by

$$\|u^{\text{exact}} - u^{\text{num}}_{\eta}\| \leq \underbrace{\|u^{\text{exact}} - u_{\eta}\|}_{O(\sqrt{\eta})} + \underbrace{\|u_{\eta} - u^{\text{num}}_{\eta}\|}_{O(\Delta x^{p})}$$
(38)

The first term at the right-hand side is the error due to the penalization term and the second term represents the discretization error (p being the formal order of accuracy of the numerical method used to discretize the equation). Here  $\|\cdot\|$  is an appropriate norm. A compromise between the two errors is to choose  $\Delta x \approx \sqrt{\eta}$ , which leads to a first-order convergence  $||u^{\text{exact}} - u_n^{\text{num}}|| \leq O(\Delta x)$  (Nguyen van yen et al., 2014). The convergence of the  $L_1$ -error of u versus different penalization parameters  $\eta$  is shown in Fig. 10(d) for different grid resolutions, where the order  $\sqrt{\eta}$  convergence can be observed. For these calculations the expected formal accuracy is p = 2 and thus the convergence is between first and second order in space as a function of the resolution *N*, confirming the theoretical analysis of Carbou and Fabrie (2003) and the numerical results of Morales et al. (2014). We also observe a saturation of the convergence error for larger *N*, due to dominance of the penalization error. An optimal resolution can be found for each  $\eta$  and vice versa. As can be seen in Fig. 10(c) by using a fine grid a smaller  $\eta$  is needed. In general for fine grids, decreasing  $\eta$  leads to an accuracy enhancement but for



**Fig. 10.** (a) Original and mollified mask function for the Couette flow, comparison of computed vorticity  $\omega$  with the exact solution at y = 0.5 with N = 128 grid points in each direction. (b) Comparison of the computed stream-function  $\psi$  and the *u* velocity component with the exact solutions at y = 0.5 with N = 128 in each direction. (c)  $L_1$ -error of *u* versus spatial resolution for different values of  $\eta$ , where *N* represents the grid resolution in each direction. (d)  $L_1$ -error of *u* versus the penalization parameter  $\eta$  for different resolutions *N*.

an explicit penalization, the time step  $\Delta t = O(\eta)$  is limited by  $\eta$  as discussed in Section 2.1.2.

#### 3.2. Fluid/solid interaction via a falling cylinder

In this section we attempt to perform a simulation of a twodimensional cylinder falling (due to the gravity) in a quiescent fluid to validate the two-way fluid/solid interaction. We compare our results with those of Gazzola et al. (2011) and Namkoong et al. (2008) which have the same physical parameters. A rigid 2D cylinder of diameter D = 0.005 m with  $\rho_b = 1.01 \rho_f$ , is released from rest in a fluid with density  $\rho_f =$  996 kg/m<sup>3</sup> and kinematic viscosity  $v = 8 \times 10^{-7} \text{ m}^2/\text{s}$  and accelerates due to gravity  $(g = -9.81 \text{ m/s}^2)$  until it reaches its asymptotic terminal velocity. The size of the domain is set to  $(x, y) \in [0, 0.04 \text{ m}] \times [0, 0.32 \text{ m}]$ . The spatial resolutions in our simulations are set to  $512 \times 4096$ and  $1024 \times 8192$ , the penalization parameter  $\eta \in [10^{-4}, 10^{-3}]$ , the time step  $\Delta t \in [10^{-4}, 10^{-3}]$  and the filter parameter for denoising of the hydrodynamic coefficients is  $\delta \in [0.001, 0.005]$ . Second and fourth order discretizations are used in the simulations. In the simulations of Gazzola et al. (2011) the resolution is  $1024 \times 8192$ , the penalization parameter  $\eta = 10^{-4}$  and the Lagrangian *CFL* is set to 0.01.

The snapshots of the vorticity isolines generated by the falling cylinder in a fully quiescent and slightly perturbed fluid are illustrated in Figs. 11 and 12, respectively. A qualitative agreement with the simulations of Gazzola et al. (2011) can be observed. Comparison of the vorticity structures at t = 13 between the simulation slightly perturbed initial condition  $(\mathbf{u}' \approx 0.001)$ with  $\times$  random number  $\times u_{\infty}$ ) represented in Fig. 12(h) and that of fully quiescent initial condition represented in Fig. 11(d), shows that the presence of perturbations in the initial condition can trigger the transition in the early stages of falling, i.e.,  $t \approx 3$ . It is particularly important to obtain comparable results with other simulations with different numerical methods where the added numerical dissipation is not necessarily the same. Without adding any initial perturbation, the transition can be triggered (e.g., at  $t \approx 10$ ) by the numerical errors which perform like a perturbation (see Fig. 11). This kind of transition is not controlled, it depends on grid resolution and the numerical implementation and explains the delayed streamwise velocity overshoot and the different transient flow fields.

Fig. 13 shows the time evolution of the streamwise and lateral velocities obtained with the present method and those of Namkoong et al. (2008) and Gazzola et al. (2011). As can be seen the streamwise velocity shows the same dynamics as the reference simulations. In particular the streamwise velocity obtained by simulation with perturbed initial condition overshoots above the terminal velocity and then slows down when the vortices start shedding. It can also be seen that there is no significant difference between the streamwise velocities by imposing no-slip and freeslip boundary conditions at the boundaries of the rectangular Cartesian grid. After transition the amplitude of the oscillations of the lateral velocity  $u_{\text{lateral}} = \pm 0.002 \text{ m/s}$  is in agreement with those of Namkoong et al. (2008) and Gazzola et al. (2011) but a phase shift can be observed due to a short delay in the transition in our simulation. The terminal streamwise velocity in the simulation with perturbed initial condition is  $u_{\text{streamwise}} = 0.024 \text{ m/s}$  resulting in Reynolds number  $Re \approx 150$  while using unperturbed initial condition  $u_{\text{streamwise}} = 0.025 \text{ m/s}$  resulting in Reynolds number  $Re \approx 156$ . In the former an overshoot can be observed in the streamwise velocity while in the later the overshoot takes place in a larger time interval or it is entirely eliminated. The terminal velocity differs maximum 5% from the reference terminal velocity in the case of perturbed initial conditions and coincides in the case of an unperturbed initial condition. The differences are due to different



**Fig. 11.** Vorticity isolines (dashed lines are used for negative values) of the falling cylinder in a quiescent fluid,  $g = -9.81 \text{ m/s}^2$ ,  $\rho_b/\rho_f = 1.01$ , D = 0.005 m,  $(x, y) \in [0, 0.04 \text{ m}] \times [0, 0.32 \text{ m}] = [0, 8D] \times [0, 64D]$ ,  $2^{\text{md}}$ -order solver is used, free-slip BC is imposed,  $\Delta t = 1.25 \times 10^{-4}$ , the resolution is set to  $512 \times 4096$ , the penalization parameter  $\eta = 10^{-3}$ , the filter parameter for denoising of the hydro-dynamic coefficients is  $\delta = 0.001$ ,  $v = 8 \times 10^{-7} \text{ m}^2/\text{s}$  and  $Re \approx 156$ .

Poisson solvers which is unbounded in the simulation of Gazzola et al. (2011), the boundary conditions which is free-slip and nopenetration in our simulations, different penalization parameters and resolutions. In the authors' viewpoint the take-home message here is that the near one relative solid/fluid density leads to a small buoyancy where invalid approximation of the hydrodynamic coefficients especially in the early stages of the fall yields the simulation to a failure. To cope with this challenge the process of denoising of the hydrodynamic coefficients with a proper filter parameter  $\delta_{\text{filter}}$  is devised in the proposed algorithm to eliminate the non physical oscillations of the hydrodynamic coefficients.

## 3.3. Validation of the solid dynamics via a falling ellipse

For further validation of the proposed algorithm to deal with rotating objects interacting with incompressible flows, the sedimentation of an ellipse due to terrestrial gravity field is considered in this section. Different behaviors like steady falling, fluttering, tumbling and chaotic motion can be observed by varying the ellipse aspect ratio a/b, density ratio  $\rho_b/\rho_f$  and the viscosity v of the fluid. These parameters can be summarized in a dimensionless moment of inertia

$$J_{
m cg}^* = 2J_{
m cg}/(\pi a^4 
ho_f) = (a^2 + b^2)(b/2a^3)(
ho_b/
ho_f)$$

and the Reynolds number  $Re = u_t L/v$ , where  $u_t$  is the sedimentation average velocity estimated by

$$u_t = \sqrt{4bg(\rho_b/\rho_f - 1)} \tag{39}$$



**Fig. 12.** Vorticity isolines (dashed lines are used for negative values) of the falling cylinder in a slightly perturbed fluid,  $g = -9.81 \text{ m/s}^2$ ,  $\rho_b/\rho_f = 1.01$ , D = 0.005 m,  $(x, y) \in [0, 0.04 \text{ m}] \times [0, 0.32 \text{ m}] = [0, 8D] \times [0, 64D]$ , performed by the 4th-order solver, free-slip BC is imposed,  $\Delta t = 1.25 \times 10^{-4}$ , resolution 4096 × 512, penalization parameter  $\eta = 5 \times 10^{-4}$ ,  $\Delta t = 1.25 \times 10^{-4}$ ,  $\delta_{\text{filter}} = 10^{-3}$ ,  $\nu = 8 \times 10^{-7} \text{ m}^2/\text{s}$  and  $Re \approx 150$ .



**Fig. 13.** Comparison of the streamwise  $u_{\text{streamwise}}$  and lateral  $u_{\text{lateral}}$  velocities of the falling cylinder via different methods/parameters with reference simulations. Symbols indicate the simulations performed by Gazzola et al. (2011) (red triangles) and Namkoong et al. (2008) (green circles). Solid and dashed lines represent the results with the proposed algorithm (4096 × 512 and  $\eta = 10^{-3}$ ), performed by the 4th-order solver with a perturbed IC and free-slip BC (blue solid), 2nd-order solver with unperturbed IC and free-slip BC (blue solid), 2nd-order solver with unperturbed IC and no-slip BC (purple dash-dot) and the finest grid (8192 × 1024 and  $\eta = 10^{-4}$ ) performed by the 2nd-order solver with perturbed IC and free-slip BC (cyan dash-dot-dot). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

in Gazzola et al. (2011). Kolomenskiy and Schneider (2009) replaced the coefficient 4 in the definition of the reference velocity Eq. (39) by  $\pi$ . In our opinion the definition of the reference velocity by Eq. (39) is questionable and needs more investigation. Using Eq. (39) for evaluation of the reference velocity leads to under prediction of the Reynolds number. We think that the average velocity  $u_t = (\bar{U}_{cg}^2 + \bar{V}_{cg}^2)^{1/2}$  in the final stage of the fall would be a better

choice. For the moment we prefer to use the ellipse aspect ratio, density ratio and the viscosity of the fluid as influencing parameters, for classification of the ellipse behavior.

The results of three simulations performed by the second order solver for the falling ellipse corresponding to steady fall, fluttering and tumbling are reported in the following. The domain of the solution for steady fall and fluttering is  $(x, y) \in [0, 5L] \times [0, 20L]$  where

L = 2a = 1 and H = 2b = 0.2 are the major and minor diameters of the ellipse, respectively. The resolution of the grid is  $N_x \times N_y = 512 \times 2048$ . For simulation of the tumbling regime a larger domain and a finer grid are needed. Therefore  $(x, y) \in [0, 10] \times [0, 10]$  and  $N_x \times N_y = 2048 \times 2048$  is used. Decreasing the kinematic viscosity from v = 0.03 m<sup>2</sup>/s to v = 0.01 and v = 0.003 results in different falling regimes. Snapshots of vorticity isolines of the falling ellipse in steady, fluttering and tumbling regimes are illustrated in Fig. 14. Other parameters used in the simulations are as follows; the polar moment of inertia around the center of gravity  $I_{zz} = J_{cg} = 0.25\pi ab(a^2 + b^2)\rho_b = 0.0157$ , the initial position  $(x_0, y_0) = (0.5L_x, L_y - 3a)$  and the initial angle of the major diameter with respect to the horizon is  $\theta_0 = \pi/4$ . The density ratio is set to  $\rho_b = 1.538\rho_f$ , the filter parameter for denoising of the hydrodynamic coefficients  $\delta = 0.001$ , the gravity in the y-direction g = -9.81 m/s<sup>2</sup> and the penalization parameter is  $\eta = 10^{-3}$ .



**Fig. 14.** Vorticity isolines (dashed lines are used for negative values) of the falling ellipse in different regimes,  $I^* = 0.16$ ,  $\rho_b/\rho_f = 1.538$ , g = -9.81, a/b = 0.5/0.1,  $X_0^{\text{rg}} = L_x/2$ ,  $Y_0^{\text{rg}} = L_y - 3a$ ,  $\theta_0 = \pi/4$ ,  $\delta_{\text{filter}} = 10^{-3}$ ,  $\eta = 10^{-3}$ .

A qualitative agreement of the (cg) trajectories in different falling regimes with the simulations of Gazzola et al. (2011) can be observed in Fig. 15. The differences in the trajectories are because of the slightly different parameters used and the chaotic behavior of ellipse in the tumbling regime. The amplitude of the oscillations



**Fig. 15.** Comparison of the (cg) trajectories of the falling ellipse between the results of the present investigation and those of Gazzola et al. (2011) (coordinates are reported in cord lengths in (a) from Gazzola et al. (2011)).

in the fluttering regime is also sensitive to the used parameters. The corresponding hydrodynamic coefficients and velocity components of the falling ellipse in the fluttering regime are plotted in Fig. 16. A comparison of the first and second order filtering of the hydrodynamic coefficients is shown in Fig. 16(a) - (c). As can be seen the second-order filtering is more efficient for denoising of the hydrodynamic forces in comparison to the firstorder filtering. The hydrodynamic coefficients in the fluttering regime show an oscillatory behavior with a principal frequency  $f_1 \approx 0.24$ . However in the side force a harmonic frequency with  $f_2 = 2f_1 \approx 0.48$  can be seen which is due to the shedding of the vortices. The chosen reference point in the simulation of the falling ellipse is the center of gravity (cg) for calculation of the polar moment of inertia, rotation angle and the torque. This choice is advantageous for simplification of the Euler Eq. (36) to not include the torque due to the buoyancy. For the simulations of the swimming fish ( $\rho_b = \rho_f$ ) the buoyancy is equal to zero. Thus, without evaluation of the torque due to the body forces in Eq. (36), the reference point can move to the head, which is more suitable for construction of the fish geometry and its kinematics, starting by the information of the head as initial conditions for Eqs. (18), (20) and (22).

## 3.4. Fish in forward gait

Anguilliform swimming presented by Gazzola et al. (2011) is considered for validation of the proposed algorithm to deal with deformable bodies interacting with incompressible flows. A periodic swimming law is defined by fitting the backbone of the fish to a given curve y(x,t) while keeping the backbone length  $l_{\rm fish}$  fixed. Let  $\xi$  be the arclength over the curvilinear coordinate of the deformed backbone ( $0 \le \xi \le l_{\rm fish}$ ). For points being uniformly distributed with  $\Delta \xi = l_{\rm fish}/(N-1)$ , over the backbone, *y* is given by

$$y(x,t) = a(x)\sin(2\pi(x/\lambda + ft))$$
(40)

where  $\lambda$  is the wavelength of the imposed deformation, *f* represents the frequency of the backbone beat and the envelope a(x) is given by

$$a(x) = a_0 + a_1 x + a_2 x^2 \tag{41}$$

where *x* is defined by inverting the arclength integral, i.e.,  $\Delta x = \Delta \xi / \sqrt{1 + (\partial y / \partial x)^2}$ . The wavelength of the fish is defined in accordance with the geometry of the backbone in the Cartesian coordinate. The pointwise curvature of the backbone is needed to use the geometrically exact theory of nonlinear beams, described in Section 2.2.1. One must switch from the Cartesian system to the curvature, thus the second derivative of Eq. (40) gives

$$k(\xi, t) = (2a_2 - (2\pi/\lambda)^2 a(\xi)) \sin(2\pi(\xi/\lambda + ft)) + (4\pi(a_1 + 2a_2\xi)/\lambda) \cos(2\pi(\xi/\lambda + ft))$$
(42)

where  $a(\xi) = a_0 + a_1\xi + a_2\xi^2$ . Using the curvature of the backbone provides a general framework which is independent of the Cartesian coordinates, this is especially interesting to prevent the



**Fig. 16.** (a) Hydrodynamic coefficients of a falling ellipse in the fluttering regime,  $l^* = 0.16$ ,  $\rho_b/\rho_f = 1.538$ , a/b = 1/5 and v = 0.01 before denoising. (b) After applying the first-order filter Eq. (32) with b = 0 and  $\alpha = 0.2$ . (c) After applying the second-order filter via Eqs. (32) and (33) with  $\delta = 0.001$ . (d) The corresponding velocity components.

ambiguity in definition of the geometry when the fish performs a complete rotation. The parameters used by Kern and Koumoutsakos (2006) and Gazzola et al. (2011) for the kinematics of the fish are as follows;  $\lambda = 1, f = 1, a_2 = 0, a_1 = 0.125/(1 + c), a_0 = 0.125c/(1 + c)$  and c = 0.03125. The profile of the fish is given by Eq. (25) and plotted in Fig. 4. The buoyancy is equal to zero, i.e.,  $\rho_b = \rho_f$ . The viscosity of the fluid is set to  $v = 1.4 \times 10^{-4}$  resulting in an approximative Reynolds number  $Re \approx 3800$ , with an asymptotic mean velocity  $U_{\text{forward}} \approx 0.52$ .

The simulations of Gazzola et al. (2011) are carried out on a rectangular domain  $(x, y) \in [0, 8l_{fish}] \times [0, 4l_{fish}]$  with resolution 4096  $\times$  2048 and a penalization parameter  $\eta = 10^{-4}$ . We perform our simulations on a rectangular domain  $(x, y) \in [0, 10l_{fish}]$  $\times [0, 5l_{fish}]$  by imposing a penalization parameter inside the body equal to  $\eta = 10^{-3}$  with resolutions  $2048 \times 1024 - 1024 \times 512$ and  $\Delta t = 10^{-3}$ . The centroid of the fish is initially positioned at  $x_{cg} = 0.9L_x$  and  $y_{cg} = 0.5L_y$ . Two snapshots of vorticity isolines at t = 1 and t = 9 with the aforementioned parameters are illustrated in Fig. 19. The forward velocities of the center of gravity (cg) of the fish, computed with different methods and parameters are compared with those of Kern and Koumoutsakos (2006) and Gazzola et al. (2011) in Fig. 18. We impose two degrees of freedom fixing the angular velocity of the fish around the center of gravity equal to zero. But this does not result in a motion without slaloming. Deformation of the fish in addition to the lateral displacement creates slaloming. The overall angular velocity of the fish comes from two components, one accounting for the vanishing rotational impulse and the other imparted by the flow. Both components are evolving in time in Gazzola et al. (2011) and are not equal to zero. Fixing the angular velocity of the fish around (cg) is the difference between our simulation and that of Gazzola et al. (2011) but the results are in good agreement. The simulations start with the body  $\mathbf{u}_{P}(\mathbf{x}, \mathbf{0}) = \mathbf{0}$  and fluid at rest, i.e.,  $\omega(\mathbf{x}, \mathbf{0}) = \psi(\mathbf{x}, \mathbf{0}) = \mathbf{0}$ . Freeslip boundary conditions are imposed at the four surrounding walls  $(\psi|_{\partial 0} = \omega|_{\partial 0} = 0)$ . The motion of the fish is initialized by gradually increasing the amplitude of the backbone through a sinusoidal function (plotted in Fig. 17), from zero to its designated value during the first period T = 1/f in the reference simulations, (Kern and Koumoutsakos, 2006; Gazzola et al., 2011). Here we do not consider this and start by a sudden movement given by Eq. (40). That is the reason why a deviation from the reference solution can be seen in the first period. This deviation will continue systematically until the asymptotic velocity is reached at t = 7.

The reference simulation of Kern and Koumoutsakos (2006) is based on a body fitted finite volume method which is first-order in time and second-order in space. The Navier-Stokes equations were solved using the commercial package STAR-CD which uses arbitrary Lagrangian-Eulerian grids. The solution of the Newton's equations of motion and the deformation and displacement of the Lagrangian grid are implemented in user defined subroutines linked to STAR-CD. The implemented explicit coupling procedure is a staggered integration algorithm proposed by Farhat and Lesoinne (2000). The simulation of Gazzola et al. (2011) is based on a remeshed vortex particle code coupled with Brinkman penalization which handles arbitrarily deforming bodies and especially the corresponding divergent velocity field inside the body. A projection method is used by Gazzola et al. (2011), the resulting Poisson equations for rotational (solenoidal) and potential (divergent) components of the velocity fields are solved in an unbounded domain, FFT based solver over Cartesian grids. A second-order finite difference discretization in two dimensions and a fourthorder finite difference discretization in three dimensions are used for all spatial derivatives. The time step is adapted by a Lagrangian CFL condition. The difference on the final forward velocity of the fish reported by Gazzola et al. (2011) by taking into account the divergence of the velocity field inside the fish due to deformation of the fish is visible in Fig. 18. Even though the average divergence over the fish volume is zero (i.e. the volume is conserved), locally inside the fish, the velocity field is not divergence free. We do not deal with this issue in this paper under the assumption that the body is slender.

In our simulations a grid independent solution is obtained with  $2048 \times 1024$  grid points. The difference of the forward velocity in two simulations with  $2048 \times 1024$  and  $1024 \times 512$  grid points can be seen in Fig. 18. Filtering of the hydrodynamic coefficients is necessary to prevent the simulation from divergence and non-physical results. We use a second-order exponential filtering (32) instead of the first-order filtering used by Kern and Koumoutsakos (2006) (see the discussion in Sections 2.2.4 and 3.3). This process is like adding a damper to the system therefore a proper value for  $\delta$  must be chosen via numerical tests to obtain reliable and physical results. We propose values in the range of  $\delta \in [0.0001, 0.01]$  for fluid/solid interaction problems, however this



**Fig. 17.** Smooth step function  $Cr(t) = t' - \sin(2\pi t')/(2\pi), t \in [t_i, t_f]$  with  $t' = (t - t_i)/(t_f - t_i), t_i = 0, t_f = 1$  for gradually starting the motion proposed by Boyer et al. (2006). At t = 0 and t = 1 the left-and right-hand limits are equal for the function Cr and its first Cr' and second Cr'' derivatives.



**Fig. 18.** Forward velocity *U* of a 2D anguilliform swimmer's ( $\lambda = f = 1$ ). Solid lines indicate the reference simulations performed by (green) Kern and Koumoutsakos (2006) and (pink and brown) Gazzola et al. (2011). Dashed lines represent the results with the proposed algorithm. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



(b) t = 9

**Fig. 19.** Snapshots of vorticity isolines obtained during a simulation where  $(x, y) \in [0, 10l_{\rm fish}] \times [0, 5l_{\rm fish}]$  by imposing a penalization parameter inside the body equal to  $\eta = 10^{-3}$ ,  $\Delta t = 10^{-3}$ , with resolution 2048 × 1024, the filter parameter  $\delta_{\rm filter} = 0.005$  for denoising of the hydrodynamic forces, the Reynolds number is  $Re \approx 3800$ .

can also depend on the manner of non-dimensionalization of the forces. In Fig. 18 the effect of filtering with two filter parameters, i.e.,  $\delta = 0.001$  and  $\delta = 0.005$ , can be seen. The simulations with a smaller filter parameter, e.g.,  $\delta = 0.001$ , are more stable but instead will lead to smaller amplitudes in the oscillations of the terminal velocity. A sensitivity analysis is thus necessary.

#### 4. Application and results

In this section we attempt to propose an efficient law for rotation control of an anguilliform swimmer. Fish maneuvering to attain a predefined fixed goal is done by adding a constant curvature  $k_{\text{offset}}(\theta_{\text{des}}, t)$  all along the fish backbone  $\xi \in [0, l_{\text{fish}}]$ , to the primary propulsion mode, i.e.,

$$k_3 = k(\xi, t) + k_{\text{offset}} \tag{43}$$

However, the change of the added curvature  $k_{\text{offset}}$  given by Eq. (45) must be gradually, i.e.,  $O(\Delta t)$  to perform a physically reasonable motion. For the fish in forward gait  $k_{\text{offset}}$  is set equal to zero. To

perform a rotation, a desired curvature  $k_{des}$  must be evaluated by the following relation,

$$k_{\rm des}(\theta_{\rm des}) = \begin{cases} -\operatorname{sgn}(\theta_{\rm des}) \ k_{\rm max} & |\theta_{\rm des}| \ge \theta_{\rm limit} \\ -\operatorname{sgn}(\theta_{\rm des}) \ k_{\rm max} \ \left(\frac{\theta_{\rm des}}{\theta_{\rm limit}}\right)^2 & \text{else} \end{cases}$$
(44)

where sgn represents the sign function, i.e.,  $sgn(\theta_{des}) = \theta_{des}/|\theta_{des}|$ , positive and negative values of  $\theta_{des}$  in the head frame will push the fish to turn left and right, respectively. For a schematic representation of  $\theta_{des}$  see Fig. 20. In each time step, first a desired angle  $\theta_{des}$  should be calculated according to the position and direction of the head by considering the goal. Next by using Eq. (44) a desired curvature  $k_{des}$  must be found. Then  $k_{offset}$  will be evaluated with the following relation,

$$k_{\text{offset}}^{n+1}(k_{\text{des}}) = \begin{cases} k_{\text{offset}}^n + \Delta k & k < k_{\text{desired}} \\ k_{\text{offset}}^n - \Delta k & \text{else} \end{cases}$$
(45)

where  $\Delta k = \Delta t \pi / T$ . After that  $k_{\text{offset}}$  must be added to the backbone curvature given by Eq. (43) for performing a rotation. Finally knowing the direction, position and the velocities of the head by considering  $Im_b = 251$  discrete points on the backbone of the fish, Eqs. (18), (20) and (22) altogether must be integrated in space to give the position and the velocities of the backbone. In the lateral direction,  $Im_{h} = 39$  points are used to construct the Lagrangian grid which covers the fish. By choosing  $k_{\text{max}} = \pi$  in Eq. (44) the fish lies over a semicircle when it turns with its maximum curvature. As in Bergmann and Iollo (2011) we use  $\theta_{\text{limit}} = \pi/4$ . The time derivative of the curvature dk/dt is needed in Eq. (22) for velocity calculation and can be calculated numerically. A simulation is performed to show the ability of the proposed law for rotation control of a swimmer toward a predefined goal. The domain size is  $(x, y) \in [0, 5l_{fish}]$  $\times$  [0, 5 $l_{\text{fish}}$ ], the resolution is set to 1024  $\times$  1024, the penalization parameter  $\eta = 10^{-3}$ , filter parameter  $\delta = 0.005$ , tail beat frequency f = 1, wavelength of deformation  $\lambda = 1, a_2 = 0, a_1 = 0.125/(1 + c),$  $a_0 = 0.125c/(1+c)$  and c = 0.03125. The profile of the fish is given by Eq. (25) and plotted in Fig. 4. The kinematic viscosity is  $v = 1.4 \times 10^{-4}$ , initial position of the head is  $(x_0, y_0) =$  $(0.1L_x, 0.5L_y)$  and initial angle of the head is  $\theta_0 = 0$ . Fig. 21 shows snapshots of vorticity isolines obtained during a simulation of swimming fish toward a predefined goal which is located at  $(x_f, y_f) = (0.9L_x, 0.5L_y)$ . The simulations start with the body  $\mathbf{u}_{P}(\mathbf{x}, \mathbf{0}) = \mathbf{0}$  and surrounding fluid at rest, i.e.,  $\omega(\mathbf{x}, \mathbf{0}) = \psi(\mathbf{x}, \mathbf{0})$ = 0. Free-slip boundary conditions are imposed at surrounding walls ( $\psi|_{\partial\Omega} = \omega|_{\partial\Omega} = 0$ ). The motion of the fish is initialized by gradually increasing the curvature of the backbone, given by Eq. (43), through a sinusoidal function (plotted in Fig. 17), from zero to its designated value during the first period T. After reaching the vicinity ( $r_{\text{goal}} = 0.5 l_{\text{fish}}$ ) of the goal the curvature of the backbone, given by Eq. (43), will tend to zero (see Fig. 22) by multiplying it with the following function,

$$C(t) = \frac{t_f - t}{t_f - t_i} + \frac{1}{2\pi} \sin(2\pi \frac{t - t_i}{t_f - t_i}), \quad t \in [t_i, t_f]$$
(46)



**Fig. 20.** Schematic representation of desired angle for curvature control in rotation,  $\theta_{des} = \theta_{goal} - \theta_{Head}$  is the difference of the angles between head direction and the line passing through the target and the head ( $-\pi < \theta_{des} < \pi$ ), picture adopted from Bergmann and Iollo (2011) with slight modification.



**Fig. 21.** Snapshots of vorticity isolines obtained during a simulation of swimming fish toward a predefined target which is located at  $(x_f, y_f) = (0.9L_x, 0.5L_y)$ . At t = 0 the fish and the surrounding fluid are in rest. After reaching the vicinity  $(r = 0.5l_{fish})$  of the target the curvature of the backbone will tend to zero by Eq. (46). The domain of the solution is  $(x, y) \in [0, 5l_{fish}] \times [0, 5l_{fish}]$ , the resolution of the Eulerian grid  $1024 \times 1024$ , resolution of the Lagrangian grid  $(Im_b \times Jm_b = 251 \times 39)$ ,  $\eta = 5 \times 10^{-4}$  and kinematic viscosity is equal to  $v = 1.4 \times 10^{-4}$ . Samples of the backbone of the fish are plotted in Fig. 24.



**Fig. 22.** Snapshots of pressure isolines obtained during a simulation of swimming fish (represented by black contour corresponding to  $\chi = 0.2$ ) toward a predefined goal which is located at  $(x_f, y_f) = (0.9L_x, 0.5L_y)$ . At t = 0 the fish and the surrounding fluid are at rest. After reaching the vicinity  $(r = 0.5l_{fish})$  of the target the curvature of the backbone will tend to zero by Eq. (46). The domain of the solution is  $(x, y) \in [0, 5l_{fish}] \times [0, 5l_{fish}]$ , the resolution of the Eulerian grid  $1024 \times 1024$ , resolution of the Lagrangian grid  $(Im_b \times Jm_b = 251 \times 39)$ ,  $\eta = 5 \times 10^{-4}$  and kinematic viscosity is equal to  $\nu = 1.4 \times 10^{-4}$ .

which is the mirror of the function presented in Fig. 17, with  $t_i = t_{\text{reached}}, t_f = t_{\text{reached}} + T$  for gradually decreasing the curvature of the backbone during one period. Samples of the backbone of the fish are plotted in Fig. 24. As can be seen in Fig. 21, the values of the vorticity start from zero and go up very fast  $\omega \in [-200, 220]$  during the rotation. In the forward gait the range of the vorticity is  $\omega \in [-60, 70]$  and finally it goes down by stopping



**Fig. 23.** Saddle points denoted by green dashed circles and vortices denoted by purple solid circles forming dipoles (i.e., strong jets) during the rotation.

the stroke in the vicinity of the goal to be in the range of  $\omega \in [-28, 25]$ . Saddle and center points in the separated flow from the fish are seen in Fig. 23 successively. These are the common characteristics of separated flows. For evaluation of the pressure field, the Poisson Eq. (5) can be simplified for the current application as follows

$$\nabla^2 p = 2(u_x v_y - u_y v_x) - \nabla \cdot \left[\eta^{-1} \chi(\mathbf{u} - \mathbf{u}_p)\right]$$
(47)

where Neumann Boundary conditions  $\partial p/\partial \mathbf{n}|_{\partial\Omega} = 0$  must be imposed at the boundaries of the rectangular domain. By using a second-order forward finite difference discretization one has

$$p_1 = (4p_2 - p_3)/3$$

over the left boundary. Similar backward/forward relations can be derived for right, up and down boundaries. A point successive over relation (PSOR) method (Press et al., 1992) with red-black sweeper is used for calculation of the pressure field every 500 iterations. The value of pressure in the center of the cavity is set to one,  $p_{\infty} = 1$ , at each iteration, i.e.,

$$p(N_x/2, N_y/2) = p_{\infty}$$

which avoids the singularity due to imposed Neumann boundary conditions. Snapshots of pressure isolines are illustrated in Fig. 22. High and low pressure regions in the right and the left sides of the fish can be seen alternatively. As expected the pressure contours are normal to the boundary of the fish and the boundaries of the computational domain. The centers of the vortices correspond to low pressure regions. The deviation of the pressure from  $p_{\infty} = 1$ goes up to  $p \in [-21, 27]$  after starting the stroke at t = 5 and goes down instantaneously when the fish reaches the vicinity of the goal (t = 15), thus stopping the stroke. This is in clear contradiction to the vorticity field which is very persistent even after stopping the stroke and proves the global nature of the pressure field against the local nature of the vorticity field. A high pressure region is seen between the head and the tail of the fish at t = 2.25 when it turns with the maximum authorized curvature  $k = \pi$  forming a c-shape which corresponds to what observed by Gazzola et al. (2012). Cbent maneuver before the escape is explained to be effective in trapping and accelerating larger volumes of fluid by Gazzola et al. (2012). Despite industrial self-propelled objects in which the



**Fig. 24.** Samples of the backbone of a swimming fish toward a predefined goal which is located at  $(x_f, y_f) = (0.9L_x, 0.5L_y)$  obtained during a simulation,  $t \in [0, 15]$ . After reaching the vicinity  $(r = 0.5I_{nsh})$  of the goal the curvature of the backbone will tend to zero by Eq. (46). The snapshots of the corresponding vorticity and pressure isolines are plotted in Figs. 21 and 22, respectively. Starting from rest the fish performs a 180° rotation within an area of about 1.3 its length.

maximum pressure occurs at the head facing the free-stream, in the swimming fish the high and low pressure regions occur in either side of the fish alternatively. However, at the final stage of the motion after stopping the stroke, a high-pressure region at the head of the fish is observed at t = 15 in Fig. 22. Smoothing of the mask function  $\bar{\chi}$  by Eq. (27) results in a smooth pressure field, there is no oscillation inside and around the fish and the pressure distribution is regular. With the proposed law for rotation which adds a time-dependent curvature (which is constant all along the camber line) to the primary propulsion mode, starting from rest the fish executes a sharp 180° turn within an area of about 1.3 its body length.

#### 5. Conclusion

In this paper an efficient algorithm for simulation of deformable bodies interacting with two-dimensional incompressible fluid flows is presented. By using a uniform Cartesian grid a direct fourth-order solver for the solution of the Poisson equation is proposed. In order to introduce a deformable body in fluid flow, the volume penalization method is applied to the solution of the Navier–Stokes equations as a forcing term. Even if the penalization method is shown to have between first and second order accuracy in space, an advantage of this method is that the evaluation of the hydrodynamic coefficients is straightforward. Proper denoising of the hydrodynamic coefficients is crucial in dealing with fluid/solid interaction problems via the volume penalization method. An efficient law for curvature control of an anguilliform swimmer toward a predefined goal is proposed which is based on geometrically exact theory of nonlinear beams. With the proposed law, the motionless fish executes a sharp 180° turn within an area of about 1.3 its body length. Validation of the developed method shows the efficiency and the expected accuracy of the algorithm for rotation control of an anguilliform swimmer and also for a variety of fluid/solid interaction problems. Some perspectives for future works are adding a multi-resolution analysis to the algorithm for grid adaptation, we refer to Gazzola et al. (2014) and Ghaffari et al. (2014), enhancement of the rotation control law, parallelization and extension to three dimensions. The FORTRAN code is open source and is accessible upon request (Ghaffari). Interested users are first encouraged to try the second-order solver over the finest possible grid, then to investigate the effect of increasing the order from second to fourth over the same or a coarser grid. However, increasing the accuracy order of the immersed boundary method is a challenging task. For high-order IBMs implemented in finite difference solvers we refer to Linnick and Fasel (2005), Bonfigli (2011), and Seo and Mittal (2011).

## Acknowledgments

We give our thanks to Professor Boyer and colleagues for useful discussions and sending their code describing the kinematics of the backbone of a fish in forward gait, to Professor Gazzola and colleagues for sending the needed data for comparisons and to the referees for providing constructive comments that greatly improved the manuscript.

#### References

- Abide, S., Viazzo, S., 2005. A 2D compact fourth-order projection decomposition method. J. Comput. Phys. 206, 252–276.
- Angot, P., Bruneau, C.-H., Fabrie, P., 1999. A penalization method to take into account obstacles in incompressible viscous flows. Numer. Math. 81 (4), 497– 520.
- Arquis, E., Caltagirone, J.P., 1984. Sur les conditions hydrodynamiques au voisinage d'une interface milieu fluide – milieux poreux: application à la convection naturelle. Comptes Rendus de l'Académie des Sciences, Paris 2 (299), 1–4.

- Belkhiri, A., 2013. Modélisation dynamique de la locomotion compliante: Application au vol battant bio-inspiré de l'insecte. Thèse de doctorat, École Nationale Supérieure des Mines de Nantes, 2013.
- Bergmann, M., Iollo, A., 2011. Modeling and simulation of fish-like swimming. J. Comput. Phys. 230, 329–348.
- Boersma, B.J., 2011. A 6th order staggered compact finite difference method for the incompressible Navier–Stokes and scalar transport equations. J. Comput. Phys. 230 (12), 4940–4954.
- Bonfigli, G., 2011. High-order finite-difference implementation of the immersedboundary technique for incompressible flows. Comput. Fluids 46, 2–11.
- Bontoux, P., Forestier, B., Roux, B., 1978. Analyse et optimisation d'une méthode de haute précision pour la résolution des équations de Navier–Stokes instationnaires. J. de Mécanique Appliquée 2 (3), 291–316.
- Bontoux, P., Viazzo, S., Schneider, K., Ghaffari, S.A., 2014. An efficient algorithm for simulation of forced deformable bodies interacting with incompressible flows. In: Application to Fish Swimming, ECCM V and 11th World Congress on Computational Mechanics, vol. II, Barcelona, Spain, pp. 787–798.
- Boyer, F., Porez, M., Khalil, W., 2006. Macro-continuous computed torque algorithm for a three-dimensional Eel-like robot. IEEE Trans. Rob. 22 (4), 763–775.
- Breder, C.M., 1926. The locomotion of fishes. Zoologica 4, 159–297.
- Brinkmann, H.C., 1947. A calculation of the viscous force exerted by a flowing fluid on a swarm of particles. Appl. Sci. Res. 1, 27–34.
  Carbou, G., Fabrie, P., 2003. Boundary layer for a penalization method for viscous
- Carloot, G., Fabie, F., 2005. Boundary layer for a penanzation method for viscou incompressible flow. Adv. Differen. Equat. 8, 1453.
- Carling, J., Williams, T.L., Bowtell, G., 1998. Self-propelled anguilliform swimming: simultaneous solution of the two-dimensional Navier–Stokes equations and Newton's laws of motion. J. Exp. Biol. 201, 3143–3166.
- Coquerelle, M., Cottet, G.-H., 2008. A vortex level set method for the two-way coupling of an incompressible fluid with colliding rigid bodies. J. Comput. Phys. 227, 9121–9137.
- El Rafei, M., Alamir, M., Marchand, N., Porez, M., Boyer, F., 2008. Multi-variable constrained control approach for a three-dimensional Eel-like robot. In: IEEE/ RSJ 2008 International Conference on Intelligent Robots and Systems, IROS, Nice, France.
- Engels, T., Kolomenskiy, D., Schneider, K., Sesterhenn, J., 2013. Two-dimensional simulation of the fluttering instability using a pseudospectral method with volume penalization. Comput. Struct. 122, 101–113.
- Farhat, C., Lesoinne, M., 2000. Two efficient staggered procedures for the serial and parallel solution of three-dimensional nonlinear transient aeroelastic problems. Comput. Methods Appl. Mech. Eng. 182, 499–516.
- Forestier, M., 2000. Etude par méthode spectrale de sillages tridimensionnels en fluide stratifié. Thèse de Doctorat, Université de Nice-Sophia.
- Gazzola, M., Chatelain, P., van Rees, W.M., Koumoutsakos, P., 2011. Simulations of single and multiple swimmers with non-divergence free deforming geometries. J. Comput. Phys. 230, 7093–7114.
- Gazzola, M., Van Rees, W.M., Koumoutsakos, P., 2012. C-start: optimal start of larval fish. J. Fluid Mech. 698, 5–18.
- Gazzola, M., Hejazialhosseini, B., Koumoutsakos, P., 2014. Reinforcement learning and wavelet adapted vortex methods for simulations of self-propelled swimmers. SIAM J. Sci. Comput. 36 (3), B622–B639.
- Ghaffari, S.A., Schneider, K., Viazzo, S., Bontoux, P., 2014. Multi-resolution analyses of incompressible flows interaction with forced deformable bodies interacting. Application to fish swimming. ECFD IV and 11th World Congress on Computational Mechanics, vol. V, Barcelona, Spain, pp. 5002–5013.
- Hansen, N., Mulller, S.D., Koumoutsakos, P., 2003. Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES). Evol. Comput. 11 (1), 1–18.
- Hirsh, R.S., 1975. High order accurate difference solutions of fluid mechanics problems by a compact differencing technique. J. Comput. Phys. 19, 90–109.
- Holt, C.C., 1957. Forecasting trends and seasonal by exponentially weighted averages, Office of Naval Research Memorandum, Carnegie Institute of Technology, 52.
- Kern, S., Koumoutsakos, P., 2006. Simulations of optimized anguilliform swimming. J. Exp. Biol. 209, 4841–4857.
- Khadra, K., Angot, P., Parniex, S., Caltagirone, J.P., 2000. Fictitious domain approach for numerical modelling of Navier–Stokes equations. Int. J. Numer. Meth. Fluids 34, 651–684.
- Kim, J.W., 2007. Optimised boundary compact finite difference schemes for computational aeroacoustics. J. Comput. Phys. 225, 995–1019.
- Kim, J., Moin, P., 1985. Application of a fractional-step method to incompressible Navier–Stokes equations. J. Comput. Phys. 59, 308–323.
- Kolomenskiy, D., Schneider, K., 2009. A Fourier spectral method for the Navier-Stokes equations with volume penalization for moving solid obstacles. J. Comput. Phys. 228 (16), 5687–5709.
- Laizet, S., Lamballais, E., Vassilicos, J.C., 2010. A numerical strategy to combine highorder schemes, complex geometry and parallel computing for high resolution DNS of fractal generated turbulence. Comput. Fluids 39-3, pp 471–484.
- Lele, S.K., 1992. Compact finite difference schemes with spectral-like resolution. J. Comput. Phys. 103, 16–42.
- Linnick, M.N., Fasel, H.F., 2005. A high-order immersed interface method for simulating unsteady incompressible flows on irregular domains. J. Comput. Phys. 204, 157–192.
- Minguez, M., 2008. Large Eddy Simulation of Turbulent Flows Over Complex Geometries: Application to a Simplified Car Model. PhD thesis, Université Paul Cézanne, Aix-Marseille III.
- Mittal, R., Iaccarino, G., 2005. Immersed boundary methods. Annu. Rev. Fluid Mech. 37, 239–261.

Monin, A.S., Yaglom, A.M., 1971. Statistical Fluid Mechanics. MIT Press, Cambridge, MA.

Morales, J.A., Leroy, M., Bos, W.J.T., Schneider, K., 2014. Simulation of confined magnetohydrodynamic flows with Dirichlet boundary conditions using a pseudo-spectral method with volume penalization. J. Comput. Phys. 274, 64–94.

- Namkoong, K., Yoo, J.Y., Choi, H.G., 2008. Numerical analysis of two-dimensional motion of a freely falling circular cylinder in an infinite fluid. J. Fluid Mech. 604, 33–53.
- Nguyen van yen, R., Kolomenskiy, D., Schneider, K., 2014. Approximation of the Laplace and Stokes operators with Dirichlet boundary conditions through volume penalization: a spectral viewpoint. Numer. Math. 128, 301–338.
- NIST/SEMATECH e-Handbook of Statistical Methods, 2012. <a href="http://www.itl.nist.gov/div898/handbook/">http://www.itl.nist.gov/div898/handbook/</a>>.

Orlandi, P., 2000. Fluid Flow Phenomena: A Numerical Toolkit. Springer.

- Paknejad, A., 2010. Development of an Aggregation Based Algebraic Multi-Grid Method for Numerical Solution of Incompressible Flows, Master thesis, MUT. Pasquetti, R., Bwemba, R., Cousin, L., 2008. A pseudo-penalization method for high
- Reynolds number unsteady flows. Appl. Numer. Math. 58, 946–954.
- Press, H., Teukolsky, A., Vetterling, T., Flannery, P., 1992. Numerical Recipes. Cambridge University Press.
- Rossinelli, D., Bergdorf, M., Cottet, G.H., Koumoutsakos, P., 2010. GPU accelerated simulations of bluff body flows using vortex particle methods. J. Comput. Phys. 229, 3316–3333.
- Roux, B., Bontoux, P., Loc, T.P., Daube, O., 1980. Optimisation of Hermitian methods for Navier–Stokes Equations in the Vorticity and Stream-Function Formulation, West Germany. Springer-Verlag, Berlin, pp. 450–468, September 9–15, 1979. (A81-36526 16-34).
- Sabetghadam, F., Ghaffari, S.A., Dadashi, M., 2009. Implementation of vortex stretching into the two-dimensional Navier-Stokes equations via arbitrary

external straining. In: Advances in Turbulence XII, Springer Proceedings in Physics, vol. 132.

- Schnipper, T., Andersen, A., Bohr, T., 2009. Vortex wakes of flapping foil. J. Fluid Mech. 633, 411–423.
- Seo, J.H., Mittal, R., 2011. A high-order immersed boundary method for acoustic wave scattering and low-Mach number flow-induced sound in complex geometries. J. Comput. Phys. 230, 1000–1019.
- Shuman, F.G., 1957. Numerical methods in weather prediction: II smoothing and filtering. Mon. Wea. Rev. 85, 357–361.
- Simo, J.C., 1985. A finite strain beam formulation. The three-dimensional dynamic problem. Part I. Comput. Methods Appl. Mech. Eng. 49, 55–70.
- Sotiropoulos, F., Yang, X., 2014. Immersed boundary methods for simulating fluidstructure interaction. Prog. Aerosp. Sci. 65, 1–21.
- Spotz, W.F., Carey, G.F., 1995. High-order compact scheme for the steady streamfunction vorticity equations. Int. J. Numer. Meth. Eng. 38, 3497–3512.
- Taylor, G.I., 1923. Stability of a viscous liquid contained between two rotating cylinders. Philos. Trans. R. Soc. A 223, 289–343.
- Ghaffari S.A., The code is developed in FORTRAN and is accessible to all by sending an email to: ghaffari@L3m.univ-mrs.fr or s.amin.ghaffary@gmail.com.
- Van Rees, W.M., Gazzola, M., Koumoutsakos, P., 2013. Optimal shapes for anguilliform swimmers at intermediate Reynolds numbers. J. Fluid Mech. 722 (R3), 1–12.
- Wikipedia contributers, 2014. Fish Locomotion, Wikipedia, The Free Encyclopedia. Williamson, C.H.K., Roshko, 1988. Vortex formation in the wake of an oscillating cylinder. J. Fluids Struct. 2, 355–381.
- Yeo, K.S., Ang, S.J., Shu, C., 2010. Simulation of fish swimming and manoeuvring by an SVD-GFD method on a hybrid meshfree-Cartesian grid. Comput. Fluids 39, 403–430.