Numerical simulation of fluid–structure interaction with the volume penalization method

Thomas Engels [a,b,*], Dmitry Kolomenskiy [c], Kai Schneider [a], Jörn Sesterhenn [b]

[a] Laboratoire de Mécanique, Modélisation et Procédés Propres (M2P2), CNRS et Aix-Marseille Université, France
[b] Institut für Strömungmechanik und Technische Akustik (ISTA), TU Berlin, Germany
[c] Department of Mathematics and Statistics, McGill University, Montréal, Canada

ARTICLE INFO

ABSTRACT

We present a novel scheme for the numerical simulation of fluid–structure interaction problems. It extends the volume penalization method, a member of the family of immersed boundary methods, to take into account flexible obstacles. We show how the introduction of a smoothing layer, physically interpreted as surface roughness, allows for arbitrary motion of the deformable obstacle. The approach is carefully validated and good agreement with various results in the literature is found. A simple one-dimensional solid model is derived, capable of modeling arbitrarily large deformations and imposed motion at the leading edge, as it is required for the simulation of simplified models for insect flight. The model error is shown to be small, while the one-dimensional character of the model features a reasonably easy implementation. The coupled fluid–solid interaction solver is shown not to introduce artificial energy in the numerical coupling, and validated using a widely used benchmark. We conclude with the application of our method to models for insect flight and study the propulsive efficiency of one and two wing sections.

© 2014 Elsevier Inc. All rights reserved.

1. Introduction

The numerical simulation of fluid–structure interaction is of fundamental interest in computational fluid dynamics (CFD) given the challenging applications encountered, for example the numerical simulation and optimization of insect flight, swimming fish or sailing boats. The fluid can deform the solid, which in turn alters vortical structures, and vice versa. The key feature is thus that the fluid–solid interface is not known a priori, but rather is a part of the solution itself. Some systems can be simplified by considering rigid solid bodies, a rather strong simplification that still proves to be challenging by its own, if the obstacle moves or features a complicated shape. Aside from the experimental approach, the method of numerically solving the coupled problem proved suitable for a large variety of problems, from swimming fish [1,2], flying insects [3,4] to parachutes [5,6].

The time-varying interface in fluid–structure interaction (FSI) problems is more than an ingredient to be added to existing codes; it rather required developing entirely new tools, especially if large deformations are involved, which is the case in the present work. A broad classification of FSI algorithms can be made either by the coupling of fluid and solids, where monolithic and partitioned solvers can be distinguished, or by the fluid mesh, which can either be conformal or non-conformal.

---

* Corresponding author at: Laboratoire de Mécanique, Modélisation et Procédés Propres (M2P2), CNRS et Aix-Marseille Université, France.
   E-mail address: thomas.engels@mailbox.tu-berlin.de (T. Engels).

Monolithic solvers [7–9] describe both media, fluid and solid, using one set of equations that are solved using same numerical schemes for both components, with all interaction resolved. These approaches are usually very stable and accurate, but require a significant code development effort, and their highly problem-specific coding makes it difficult to provide them with new features. Partitioned solvers, on the other hand, use distinct approaches for fluid and solid, allowing the use of legacy codes that have been developed previously. Special care has to be devoted to treating the coupling condition, which may cause numerical instabilities [10–14].

Another distinctive feature of FSI solvers is the fluid mesh, which may be body-fitted or not. In the former group, Gomes et al. [15] used a structured, partitioned mesh that matches the interface of a flexible appendix in the wake of a cylinder, and studied the flow-induced vibrations both numerically and experimentally. A general difficulty of this family of methods is the generation of suitable mesh, a question which has been addressed for example by Wick [16] for structured, moving meshes or by Bathe et al. [17] based on unstructured grids. These methods have been used successfully for a wide range of problems [18–20].

Despite their success, alternatives to the classical methods have been developed. These approaches do not require body-fitted moving grids, but rather rely on a fixed grid. The obstacle is then taken into account by either locally modifying the discretization, or by adding specific terms to the underlying equations. Immersed boundary techniques which are still of rising interest in CFD are particularly attractive due to their simplicity in terms of implementation in existing solvers and their flexibility in terms of handling non-academic geometries moving or even deforming in time. This branch has been pioneered by Peskin [21–23] and has been popularized since around a decade and a half now. The key feature of these methods, which are also used in the present work, is their simplicity. Arbitrary geometries can easily be incorporated in existing fluid solvers, and the fixed fluid mesh allows using fast solvers. Such an approach has been used to simulate thrust production of plunging foils [24], investigate the stability of a foil in uniform stream [25] or for elastic swimmer models [26], to name but a few. An overview can also be found in [27]. The fluid–structure interaction solver presented in this paper belongs to this category, relying on the volume penalization method [28–30], which belongs to the family of immersed boundary and fictitious domain methods.

This article is a part of our on-going work on the development of a tool for numerical simulation of the flapping flight of insects. The basic Fourier pseudo-spectral method with volume penalization for modeling rigid flapping wings was proposed in [31], extended to three-dimensional flows in [32] and successfully employed to explore aspects of insect flight such as the dynamics of leading-edge vortices [33] and the force balance during takeoff [34].

Though the rigid wing assumption proved to be useful for understanding some basic aerodynamic mechanisms of insect flight, it is recognized that insect wings can undergo deformation that significantly affects the aerodynamic forces [35]. The mechanics of flapping flight is essentially three-dimensional. The relative importance of chordwise, spanwise and twist deformations depends on the species and the overall effect of the wing flexibility varies [36,3,37]. However, in some cases, the problem can be simplified by only considering one type of deformations. For example, flexible membrane wings driven by stiff leading edges have been used for micro air vehicles [38]. The efficiency of this design strongly depends on the chordwise flexibility, and, even if the flow is three-dimensional, a two-dimensional model can explain many experimental observations [39].

In view of the above, in this work we focus on modeling the chordwise flexibility only. We consider a two-dimensional approximation, and we develop a numerical method for studying the propulsion of flexible flapping foils. There have been a number of numerical methods proposed for this problem that differ in their treatment of the fluid, the solid model and the interface. For high Reynolds number fluid flows, vortex methods have been employed [40,41], as well as the solution of the Navier–Stokes equations using body-fitted grids [42,43]. Navier–Stokes solvers on Cartesian grids and immersed boundary methods have been used at lower Reynolds numbers [44–46]. Solid solvers based on finite element [43,4] of finite difference [40] methods have been developed.

In this work, we propose a technique for fluid–structure interaction based on the volume penalization method, which was proposed for modeling and computing flow around fixed obstacles with complicated geometries on simple numerical grids [28]. It is physically motivated by replacing the solid obstacle by a porous medium and its distinctive feature is the existence of a rigorous convergence proof [29] which shows that the solution of the penalized Navier–Stokes equations tends indeed towards the exact solution of Navier–Stokes imposing no-slip boundary conditions. In [31], a first extension to moving, rigid obstacles has been presented. The present article is the successor to this paper and develops techniques required for the computation of flexible obstacles.

The remainder of this article is organized as follows. First, we summarize the fluid solver and the volume penalization method we will use in this article. In the next step we address the question how the free penalization parameter of this method can be chosen in order to optimize its precision. In Section 4, we extend the method to deal with fluid–structure interaction and derive an adequate solid model to be used in the simulations. The method is then extensively validated. In the last section, we apply our method to simplified models for insect flight and study the propulsive efficiency of heaving foils. Finally, conclusions are drawn and an outlook on future work is given.
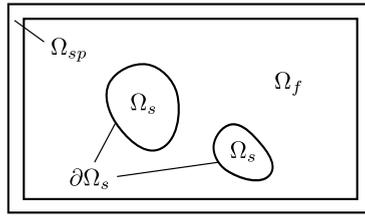
Fig. 1. Domain definitions. The original Dirichlet problem in the fluid domain Ω_f is replaced by the penalized problem in $\Omega = \Omega_f \cup \Omega_s \cup \Omega_{sp}$, where Ω is the two-dimensional torus $\mathbb{T}^2$. The solid domains Ω_s represent solid obstacles, while the sponge domain Ω_{sp} is used to prevent the obstacles' wakes from re-entering the domain.

2. Governing equations and numerical method

2.1. Governing equations

We consider a viscous, incompressible Newtonian fluid governed by the Navier–Stokes equations, in a two-dimensional, periodic domain Ω_f, with an immersed obstacle Ω_s, i.e. $\Omega_f = \mathbb{T}^2 \backslash \Omega_s$:

$$\partial_t \underline{u} + \begin{pmatrix} -\omega u_y \\ \omega u_x \end{pmatrix} = \underline{F}_p - \nabla q + \frac{1}{\text{Re}} \nabla^2 \underline{u} \tag{1a}$$

$$\nabla \cdot \underline{u} = 0 \tag{1b}$$

$$\underline{u}|_{\partial \Omega_{s(t)}}(s,t) = \underline{u}_s(s,t) \tag{1c}$$

$$\underline{u}(\underline{x}, t=0) = \underline{u}_0(\underline{x}) \tag{1d}$$

where s is a curvilinear coordinate along the fluid–solid interface $\partial \Omega_s$, $\underline{u}(\underline{x}, t)$ is the velocity, $q(\underline{x}, t)$ is the total pressure, $\underline{F}_p(t)$ is an external force, $\underline{\omega} = \nabla \times \underline{u}$ is the vorticity and $\text{Re} = u_\infty \ell / \nu$ is the Reynolds number, based on the free-stream velocity $\underline{u}_\infty$, a length scale ℓ and the kinematic viscosity ν. Normalization units are chosen such that the density ϱ is unity. The external force $\underline{F}_p$ is equivalent to the presence of an externally imposed pressure gradient. Indeed the term ∇q has by construction vanishing spatial average, since Ω_f is periodic. Thus, ∇q cannot model an external pressure gradient. Problem (1a)–(1d) can be challenging to solve numerically if the fluid–solid interface features a complicated, time-dependent geometry, which is the case in the present article.

To solve the Dirichlet problem (1a)–(1d) numerically, we use the volume penalization method [28,29].

Thus we consider a computational domain Ω, which is the two-dimensional torus $\mathbb{T}^2$. The original domain Ω_f is embedded into Ω, see Fig. 1. Eq. (1a) is supplemented by adding a penalization term to impose the Dirichlet boundary condition on $\partial \Omega_s$:

$$\partial_t \underline{u} + \begin{pmatrix} -\omega u_y \\ \omega u_x \end{pmatrix} = \underline{F}_p - \nabla q + \frac{1}{\text{Re}} \nabla^2 \underline{u} - \frac{\chi}{\varepsilon}(\underline{u} - \underline{u}_s) \tag{2a}$$

$$\nabla \cdot \underline{u} = 0 \tag{2b}$$

$$\underline{u}(\underline{x}, t=0) = \underline{u}_0(\underline{x}) \tag{2c}$$

where

$$\chi(\underline{x}, t) = \begin{cases} 1 & \underline{x} \in \overline{\Omega}_s \\ 0 & \underline{x} \in \Omega_f \end{cases} \tag{3}$$

is the mask function, $\underline{u}_s$ is the velocity field of the solid in Ω_s and $\varepsilon \ll 1$ is the so-called penalization parameter. Note that Eqs. (2a)–(2c) do no longer include Dirichlet boundary conditions, as all geometric information is now encoded in the χ-function. In view of our application to moving boundaries, we will allow the χ-function to smoothly fade from the fluid to the solid, using interim values between 0 and 1. This gives rise to a thin smoothing layer, as specified later. The penalization approach is physically motivated by the intuition that a solid wall can be modeled as a permeable solid with small permeability ε.

Using the penalization method, one enjoys a significant reduction in computational complexity, as the discretization scheme can be chosen independently of the geometry at hand. This, in turn, implies that Eqs. (2a)–(2c) can be solved on a regular Cartesian grid.

For two-dimensional flows exclusively considered in this work, it is convenient to work with the vorticity-velocity formulation, since the vorticity is scalar-valued in this case. The governing equation then reads

$$\partial_t \omega + (\underline{u} \cdot \nabla)\omega = \frac{1}{\text{Re}} \nabla^2 \omega - \nabla \times \left(\frac{\chi}{\varepsilon}(\underline{u} - \underline{u}_s) \right). \tag{4}$$

The velocity can be determined as $\underline{u} = \nabla^\perp \psi$, introducing the streamfunction ψ, which satisfies $\nabla^2 \psi = \omega$, and $\nabla^\perp = (-\partial_y, \partial_x)^T$ as the orthogonal gradient. The velocity $\underline{u}$ is defined up to an irrotational contribution that can be forced independently, i.e. $\underline{u} = \nabla^\perp \psi + \underline{U}$. Since the only periodic potential flow is a constant velocity, $\underline{U} = \underline{u}_\infty$ can be interpreted as the mean velocity. The mean flow $\underline{u}_\infty$ can be forced to a desired value, or it can be determined dynamically by imposing an external pressure force. In the latter case, an ordinary differential equation for $\underline{u}_\infty$ can be derived by spatially averaging Eq. (2a),

$$\underbrace{\int_{\mathbb{T}^2} \partial_t \underline{u}\, \mathrm{d}\underline{x}} + \underbrace{\int_{\mathbb{T}^2} \underline{\omega} \times \underline{u}\, \mathrm{d}\underline{x}}_{=0} = \underline{F}_p - \underbrace{\int_{\mathbb{T}^2} \nabla q\, \mathrm{d}\underline{x}}_{=0} + \frac{1}{\mathrm{Re}} \underbrace{\int_{\mathbb{T}^2} \nabla^2 \underline{u}\, \mathrm{d}\underline{x}}_{=0} - \int_{\mathbb{T}^2} \frac{\chi}{\varepsilon}(\underline{u} - \underline{u}_s)\, \mathrm{d}\underline{x}$$

$$\frac{\mathrm{d}\underline{u}_\infty}{\mathrm{d}t} = \underline{F}_p - \int_V \frac{\chi}{\varepsilon}(\underline{u} - \underline{u}_s)\, \mathrm{d}\underline{x}.$$

A Poisson equation for the total pressure q is obtained by taking the divergence of Eq. (2a), yielding

$$-\nabla^2 q = \nabla \cdot \begin{pmatrix} -\omega u_y \\ \omega u_x \end{pmatrix} - \nabla \cdot \left(\frac{\chi}{\varepsilon}(\underline{u} - \underline{u}_s) \right), \tag{5}$$

where we note that the penalization term itself is not divergence-free. The hydrostatic pressure p is then obtained by subtracting the kinetic energy from the total pressure, i.e. $p = q - \frac{1}{2}\underline{u} \cdot \underline{u}$.

2.2. Discretization

Eqs. (2a)–(2c) can be discretized with an arbitrary discretization scheme. In particular, we can enjoy a fast and accurate Fourier pseudo-spectral discretization on a regular Cartesian mesh [47,48]. Hence, we represent the vorticity and all other variables as truncated Fourier series:

$$\omega(\underline{x}, t) = \sum_{k_x=-N_x/2}^{N_x/2-1} \sum_{k_y=-N_y/2}^{N_y/2-1} \widehat{\omega}(\underline{k}, t) \exp(\iota \underline{k} \cdot \underline{x}) \tag{6}$$

where $\underline{k} = (k_x, k_y)^T$ is the wave vector, $\iota = \sqrt{-1}$ and the hat superscript denotes the Fourier coefficients. The fast Fourier transform (FFT) is used to compute the Fourier coefficients with $N \log(N)$ complexity, where $N = N_x N_y$. Domain lengths different from 2π can easily be rescaled. The gradient and Laplace operator can then be computed by multiplying with $\iota \underline{k}$ and $-|\underline{k}|^2$, respectively. Products are evaluated in physical space, thus avoiding expensive convolutions with $\mathcal{O}(N^2)$ complexity. Denoting the Fourier transform with $\mathcal{F}$, i.e. $\widehat{\omega} = \mathcal{F}(\omega)$, we can write the semi-discrete version of Eq. (4):

$$\partial_t \widehat{\omega} = -\mathcal{F}\left(\underline{u} \cdot \mathcal{F}^{-1}(\iota \underline{k}\widehat{\omega}) \right) - \frac{1}{\mathrm{Re}}|\underline{k}|^2 \widehat{\omega} - \iota \underline{k} \times \mathcal{F}\left(\frac{\chi}{\eta}(\underline{u} - \underline{u}_s) \right) \tag{7}$$

$$\widehat{\underline{u}} = \frac{\iota \underline{k}^\perp}{|\underline{k}|^2}\widehat{\omega} + \underline{u}_\infty, \quad |\underline{k}| \neq 0 \tag{8}$$

The Fourier coefficients are dealiased using the 2/3 rule. Due to the diagonality of the Laplacian in Fourier space, a Poisson equation can be solved without solving a linear system, as can be seen from Eq. (8). Eq. (7) is yet to be discretized in time. We can again enjoy the diagonality of the Laplace-operator to integrate the diffusive term exactly. This method is known as integrating factor technique. To simplify the notation, we rewrite Eq. (7) as

$$\partial_t \widehat{\omega} + \frac{1}{\mathrm{Re}}|\underline{k}|^2 \widehat{\omega} = \widehat{f}(\widehat{\omega}) \tag{9}$$

We can easily find the solution of the homogeneous problem. The solution of the complete equation (9) then reads

$$\widehat{\omega}(\underline{k}, t^{n+1}) = \widehat{\omega}(\underline{k}, t^n) \exp\left(-\frac{\Delta t}{\mathrm{Re}}|\underline{k}|^2 \right) + \int_{t^n}^{t^{n+1}} \exp\left(-\frac{\tau}{\mathrm{Re}}|\underline{k}|^2 \right) \widehat{f}(\omega(\underline{x}, t^{n+1} - \tau))\mathrm{d}\tau. \tag{10}$$

The Duhamel integral on the right hand side is discretized with an explicit Adams–Bashforth scheme of second order accuracy. The exact treatment of the diffusive term avoids introducing an additional stability condition on the time step Δt.

On the other hand, the penalty term introduces additional numerical stiffness and thus imposes a restriction on the time step. Using this time marching scheme, the stability condition for the time step is

$$\Delta t \leq \min\left(\eta, \mathrm{CFL}\frac{\Delta x^\alpha}{u_{\max}} \right), \quad 0 < \mathrm{CFL} < 1 \tag{11}$$

where $\alpha = 4/3$ [49]. Further details about this time marching scheme and the spatial discretization can be found in [50,31].

2.3. Hydrodynamic forces

The hydrodynamic forces acting on the solid obstacles can be computed by integrating the stress tensor over the surface:

$$\underline{F} = \oint_{\partial \Omega_s} \sigma \cdot \underline{n} \, \mathrm{d}\gamma. \tag{12}$$

$$\sigma = -\nabla p + \frac{\nu}{2}\left(\nabla \underline{u} + (\nabla \underline{u})^T\right) \tag{13}$$

If one is interested in the integral forces only, as opposed to the local distribution, volume integration can be employed. Volume integrals are computationally more efficient than surface integrals. We consider a control volume V, that may or may not coincide with an obstacle. Integrating (2a) over V and rearranging yields:

$$\underline{F} = \oint_{\partial V} \sigma \cdot \underline{n} \, \mathrm{d}\gamma = \int_V \frac{\chi}{\eta}(\underline{u} - \underline{u}_s)\mathrm{d}V + \int_V (\partial_t \underline{u} + \underline{u} \cdot \nabla \underline{u})\mathrm{d}V \tag{14}$$

where we used

$$\int_V \nu \nabla^2 \underline{u} - \nabla p \, \mathrm{d}V = \int_V \nabla \cdot \sigma \, \mathrm{d}V = \oint_{\partial V} \sigma \cdot \underline{n} \, \mathrm{d}\gamma.$$

If V coincides with an obstacle at rest, then (14) reduces to its first term, $\underline{F} = \int_V \frac{\chi}{\eta}\underline{u}\mathrm{d}V$, which is the classical formula [28]. If V coincides with a moving obstacle, then we are left with the additional second term known as unsteady correction. If the obstacle is rigid, it can be computed as $V_s \dot{\underline{u}}_s$, as described in [51], where V_s is the obstacles' volume and $\dot{\underline{u}}_s$ its acceleration. If the object is flexible, the unsteady correction has to be computed numerically. This can be done by either explicitly deriving the integral over the solid velocity field, $\mathrm{d}/\mathrm{d}t \int_V \chi \underline{u}_s \mathrm{d}V$, or by integrating the right hand side of the penalized Navier–Stokes equation (2a). In the present work, we choose the former way and numerically integrate the unsteady correction.

Analogously, the moment on the obstacle can be computed using

$$M = \oint_{\partial V} \underline{r} \times (\sigma \cdot \underline{n}) \, \mathrm{d}\gamma = \int_V \underline{r} \times (\partial_t \underline{u} + \underline{u} \cdot \nabla \underline{u})\mathrm{d}V + \int_V \underline{r} \times \frac{\chi}{\eta}(\underline{u} - \underline{u}_s)\mathrm{d}V \tag{15}$$

where $\underline{r} = \underline{x} - \underline{x}_c$ is the position vector relative to the center of gravity. Again the unsteady correction appears, which reduces to $\int_V \underline{r} \times (\partial_t \underline{u} + \underline{u} \cdot \nabla \underline{u})\mathrm{d}V = J_c \ddot{\Theta}_c$ in the case of rigid objects. Therein, J_c is the geometrical moment of inertia and Θ is the angle of rotation.

2.4. Wake removal techniques

The inherent periodicity of Fourier methods can be alleviated by again using the volume penalization method. Since we work with the vorticity formulation of the Navier–Stokes equations, we can directly use the penalty method to impose a Dirichlet boundary condition on the vorticity, thus preventing the wake from re-entering the domain. The corresponding domain Ω_{sp} is illustrated in Fig. 1. This can be done by adding another penalization term to Eq. (4):

$$\partial_t \omega + (\underline{u} \cdot \nabla)\omega = \frac{1}{\mathrm{Re}}\nabla^2 \omega - \nabla \times \left(\frac{\chi}{\varepsilon}(\underline{u} - \underline{u}_s)\right) - \frac{\chi_{sp}^{\omega}}{\varepsilon_{sp}^{\omega}}(\omega - \omega_0) \tag{16}$$

where χ_{sp}^{ω} is the mask function of the vorticity sponge. This technique will be used for simulations of unbounded flows, such as the simulations of the spanwise wing sections of flapping wings in Section 5. The desired vorticity profile ω_0 is usually zero.

In the case of a channel flow with a given Dirichlet condition on the velocity, one can use an alternative technique and directly impose a Dirichlet condition on the velocity, as it was already done for the obstacle:

$$\partial_t \omega + (\underline{u} \cdot \nabla)\omega = \frac{1}{\mathrm{Re}}\nabla^2 \omega - \nabla \times \left(\frac{\chi}{\varepsilon}(\underline{u} - \underline{u}_s) + \frac{\chi_{sp}^{u}}{\varepsilon_{sp}^{u}}(\underline{u} - \underline{u}_0)\right). \tag{17}$$

The difference between the actual obstacle and the sponge is that χ_{sp}^{u} does not depend on time. Note that the supplementary term in Eq. (16) does not appear in the Poisson equation for the pressure (5), while the one in Eq. (17) does.

In both cases, the respective penalization parameters can be chosen independently. In general, the upstream influence of the sponges should be minimized. This can be achieved by either choosing a large corresponding ε, or by smoothing the $\chi_{sp}^{\omega,u}$ functions accordingly. Both u and ω sponge techniques can also be combined, as discussed in Section 5.1.
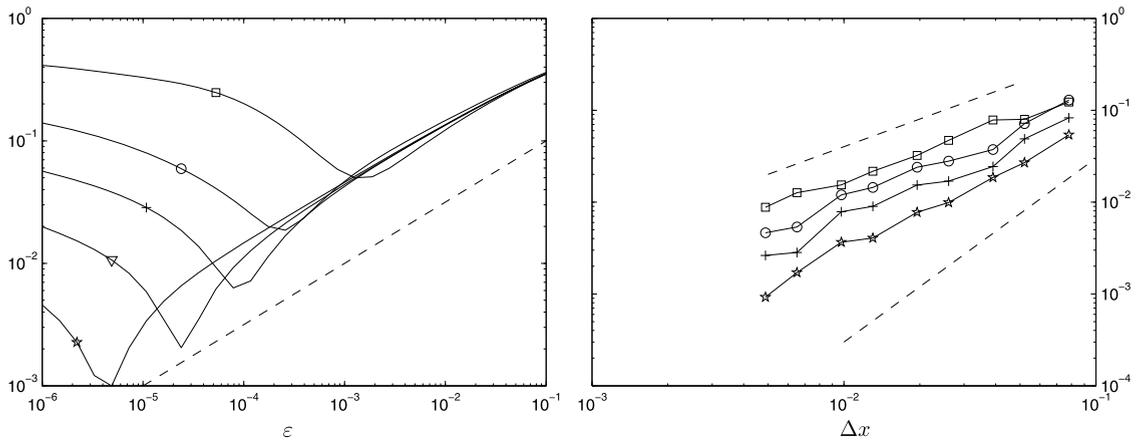
Fig. 2. Relative L_2 error $\|\underline{u}^{ex} - \underline{u}_\varepsilon^N\|/\|\underline{u}^{ex}\|$ for the Taylor–Couette flow. Left: convergence with respect to ε. Resolutions are 32^2 ($\square$), 64^2 ($\bigcirc$), 128^2 ($+$), 256^2 ($\triangledown$) and 512^2 ($\star$). The dashed line marks the $\sqrt{\varepsilon}$ slope. Right: convergence with respect to Δx. Here, ε is slaved to Δx: $\varepsilon = \varepsilon_0 (\Delta x/\Delta x_0)^2$ where the penalty parameter ε_0 at the coarsest resolution is 10^{-2} ($\square$), $2.5 \cdot 10^{-4}$ ($\bigcirc$), $5 \cdot 10^{-4}$ ($+$) and 10^{-3} ($\star$). Note an intermediate value ε yields best results. Dashed lines indicate first (top) and second (bottom) order convergence.

3. Convergence properties: how to choose ε?

The penalization method described previously is proved to converge to the solution of the original Dirichlet problem in the limit $\varepsilon \to 0$, with a convergence rate of $\mathcal{O}(\sqrt{\varepsilon})$ [28,29]. These proofs are obtained using rigorous mathematics in the continuous setting and thus neglect the aspect of discretization, which is important from a practical point of view.

The error of the numerical solution of the penalized problem (2a)–(2c) with respect to the original problem can be decomposed into two components,

$$\left\| \underline{u}^{ex} - \underline{u}_\varepsilon^N \right\| \leq \left\| \underline{u}^{ex} - \underline{u}_\varepsilon \right\| + \left\| \underline{u}_\varepsilon - \underline{u}_\varepsilon^N \right\|, \tag{18}$$

where $\underline{u}^{ex}$ is the exact solution of (1a)–(1d) and $\underline{u}_\varepsilon$, $\underline{u}_\varepsilon^N$ is the exact and numerical solution of (2a)–(2c), respectively. The first contribution in Eq. (18) is the penalization error which has been shown to decay as $\mathcal{O}(\sqrt{\varepsilon})$. The second contribution is the discretization error which decays as $\mathcal{O}(\Delta x^2)$, since $\underline{u}_\varepsilon$ is in C^1 only.

From the physical interpretation of ε as permeability and in view of the slow $\mathcal{O}(\sqrt{\varepsilon})$ convergence, a first intuition would be to choose a very small value, say slightly larger than the machine precision, i.e. $\varepsilon = 10^{-13}$. Unfortunately, the time step restriction (11) for our explicit time marching scheme prevents us from doing so. An apparent way out would be to treat the penalization term with an implicit time marching scheme. This approach has been discussed for the Burgers equation in [31]. In [52], an extension to the Navier–Stokes equation was presented, revealing the intimate coupling between ε and Δt: the error saturates around $\varepsilon \approx \Delta t$. Thus, there is no gain in choosing a very small ε and a larger time step; the error is almost the same as if one would have chosen both of the same order of magnitude. This behavior can be understood by the interpretation of the penalty term as a strong damping term on the velocity in Eq. (2a). It introduces a characteristic time scale, which is of order ε, that has to be resolved by the time marching scheme.

The central question when using the penalization method is thus how to appropriately choose ε for a given problem.

To answer it, one may consider the Taylor–Couette flow between two coaxial cylinders. Let the cylinders have radii $R_1 = 0.5$ and $R_2 = 1.0$. The inner one rotates with an angular velocity Ω, and the outer one is fixed. The exact solution for the steady state is an azimuthal velocity field [53]:

$$u_\Theta(r) = -\Omega \frac{R_1^2}{R_2^2 - R_1^2} r + \Omega \frac{R_1^2 R_2^2}{R_2^2 - R_1^2} \frac{1}{r}. \tag{19}$$

Our numerical solution is obtained using a $[0, 2.5] \times [0, 2.5]$ domain and a viscosity of $\nu = 0.1$. A discontinuous mask function is used. The initial condition is taken to equal the exact solution, and the computation is stopped when the time derivatives of both L_2 and L_∞ norms are smaller than 10^{-4}, which corresponds to a final time of around $T = 1.0$. This test case being computationally cheap, we can afford to test five resolutions, 32^2, 64^2, 128^2, 256^2 and 512^2 with 30 values of ε each, distributed logarithmically equidistant in $\varepsilon = [10^{-6}, 10^{-1}]$. The time step is $\Delta t = \min(10^{-4}, \varepsilon)$ to keep the time discretization error negligible.

Fig. 2 (left) illustrates the relative error in L_2 norm as a function of ε and the spatial resolution. It exhibits the coupling between the spatial resolution and the penalization parameter. First, one observes the $\sqrt{\varepsilon}$ slope predicted from theory [28] and the saturation of the error for small ε. The behavior is consistent with the loss of the C^1 regularity of the exact solution of the penalized problem in the limit of small ε. For each resolution, one can identify an optimal value of ε, that minimizes

the error. This is also consistent with [30], where the penalized Laplace and Stokes operators were analyzed analytically and the relation between ε and Δx has been shown to be

$$\varepsilon \propto \Delta x^2. \tag{20}$$

In view of the connection between ε and both the spatial and temporal discretization, it is clear that the strategy of choosing some fixed, small value of ε is sure to give sub-optimal results. Instead, one should choose ε according to Eq. (20). Fig. 2 (right) illustrates the spatial convergence for four different values of ε at the coarsest resolution (32^2). All curves exhibit first- to second order convergence, but the choice of the constant of proportionality in Eq. (20) modifies the error offset. The smallest error is obtained for $\varepsilon_0 = \varepsilon(N_x = 32) = 10^{-3}$.

Although the order of convergence remains the same, we can thus tune the penalization method by carefully choosing ε_0. Due to the action of viscosity ν, a small penalization boundary layer with a characteristic length scale of $\sqrt{\nu\varepsilon}$ forms inside the solid, see also [29]. This boundary layer can be used to get an idea on how to choose ε. When looking at the vorticity in the case of the Taylor–Couette flow, one notices that this boundary layer is far from being resolved when the error is minimal. As a practical guideline, one should consider a coarse resolution and start with a large ε, which is decreased until the boundary layer in the vorticity is no longer resolved. This value is then chosen as constant of proportionality. The resulting value depends on the Reynolds number, and the penalization method works better when ν is small, i.e. for larger Re.

It should be noted that it is likely to be impossible to find the optimal value of ε in cases where no reference solution exists.

Due to the restriction $\Delta t < \varepsilon \propto \Delta x^2$, a first order scheme in time would in principle be sufficient. However, the Adams–Bashforth scheme does not introduce a significant additional cost but ensures that the time discretization error is negligible in front of the other error contributions.

4. Fluid–structure interaction

In previous work, the penalization method has been extended to take moving rigid obstacles into account. The algorithm has been presented in [31], and we recall only the essential idea by considering an obstacle that undergoes pure translational motion with a velocity $\underline{u}_s$. Basically, the approach consists in solving a transport equation for the mask function,

$$\partial_t \chi + \underline{u}_s \cdot \nabla \chi = 0, \tag{21}$$

using the same discretization as for the Navier–Stokes equation. The gradient operator being diagonal in Fourier space, this equation can be solved exactly. Thus, the obstacle at time t^n is described by

$$\chi\left(\underline{x}, t^n\right) = \mathcal{F}^{-1}\left(e^{-i\underline{k}\cdot\underline{u}_s t} \mathcal{F}\left(\chi\left(\underline{x}, t^0\right)\right)\right). \tag{22}$$

Using Eq. (22) the obstacle can be displaced by less than one grid point. Always using the initial mask, $\chi(\underline{x}, t^0)$, avoids error accumulation. For general $\underline{u}_s$ the motion can be decomposed in a sequence of one-dimensional transformations [31]. Hence, arbitrary motion can be simulated with this approach. In order to avoid Gibbs oscillations, the discontinuous χ-function can be smoothed by solving a few steps of a diffusion equation prior to the simulation.

4.1. Penalization method for flexible obstacles

For fully flexible obstacles the approach of shifting the mask in Fourier space is no longer applicable, because the velocity field $\underline{u}_s$ in Eq. (21) is no longer constant. On the other hand, the discontinuous mask function cannot be translated in physical space by less than one grid point, and this jerky motion produces spurious oscillations in the hydrodynamic forces, as shown in [31]. One could solve the transport equation (21) with a different discretization scheme, but this would introduce additional difficulties such as numerical diffusion. A way out of this dilemma is to make use of the smoothing layer that has already been introduced.

In the present work, we use an analytical expression for the mask function, that can easily be resampled on the Eulerian grid. To this end we introduce a smoothed Heaviside function

$$H(x) = \frac{1}{2}\left[\operatorname{erf}\left(\frac{t - x}{\delta}\right) + \operatorname{erf}\left(\frac{t + x}{\delta}\right)\right] \tag{23}$$

where

$$\delta = c_{\mathrm{sm}}\,\Delta x \max\left(\frac{\partial H}{\partial x}\right) = \frac{c_{\mathrm{sm}}\,\Delta x}{\sqrt{\pi}}\left[\exp(-4) - 1\right].$$

The parameter c_{sm} defines the thickness of the smoothing layer in mesh widths. This particular choice for $H(x)$ is motivated by the fact that it provides a good compromise between compact support in Fourier and physical space.

To validate this approach we use the setup proposed by Wan et al. [54]. It consists of a circular cylinder of radius $R = 0.05$ immersed in a channel of length $L = 2.2$ and height $H = 0.41$. The coordinate of the cylinder center is given by
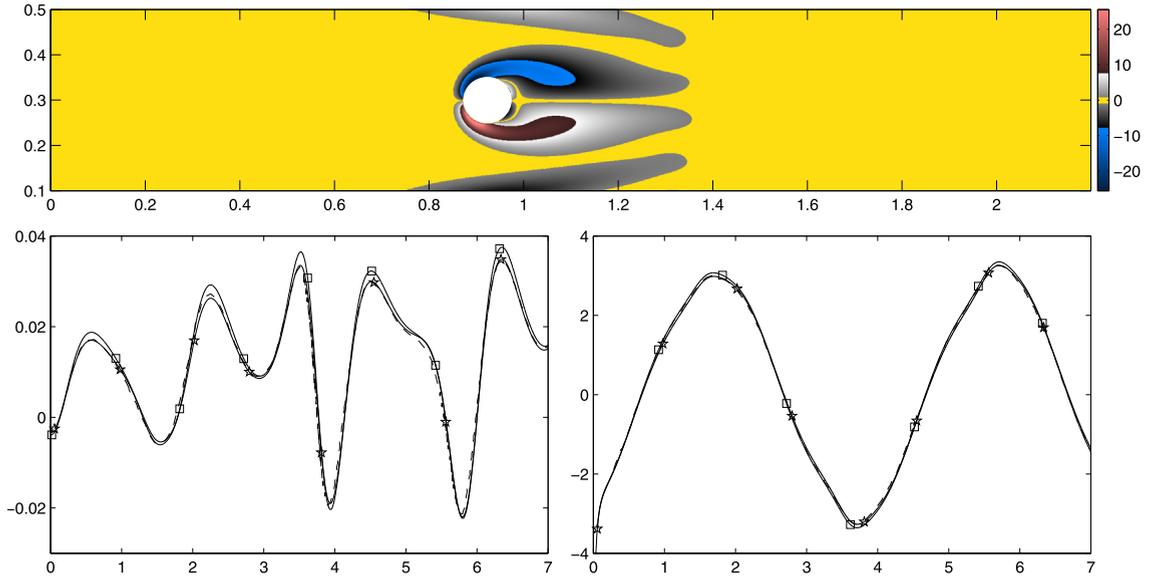
Fig. 3. Hydrodynamic forces acting on an oscillating cylinder in a channel. Top: snapshot of the vorticity field. Bottom: lift (left) and drag force (right). Solid lines are present results (1024 × 284: □, 2048 × 568: ⋆) and results of Wan et al. [54] are represented by dashed lines.
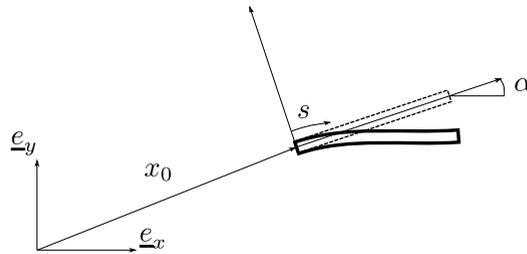

Fig. 4. Illustration of the solid model used in the present work. The beam is clamped with respect to the relative coordinate system, which is translated by x_0 and rotated by α.

$$\underline{x}_c(t) = \begin{pmatrix} x_0 + A \sin(2\pi f_0 t) \\ y_0 \end{pmatrix}$$

where $x_0 = 1.1$, $y_0 = 0.2$, $A = 0.25$ and $f_0 = 0.25$. The penalization parameter is $\varepsilon = 10^{-3}$ at the coarse resolution (1024 × 284) and $\varepsilon = 2.5 \cdot 10^{-4}$ at the finer one (2048 × 568). The smoothing parameter is $c_{sm} = 2$. The reference computation in [54] is computed with an FEM solver using a body-fitted grid in a moving reference frame. Fig. 3 illustrates the hydrodynamic forces acting on the cylinder. Good agreement can be observed. Quantitatively, the difference for the drag is 4.58% and 2.57%, and for the lift we find 12.05% and 7.04% for the coarse and fine resolution, respectively. The difference is computed by the L_2 norm of the time series of the difference, normalized by the L_2 norm of the reference signal. We can thus conclude that the smoothed mask function yields reasonable results.

4.2. Solid model

4.2.1. Model equations

The focus in the present work lies on the simulation of fluid–structure interaction. Thus, a suitable solid model is required. To this end, we introduce a one-dimensional model equation instead of solving the two-dimensional elastic continuum equations. We will show that this model performs well compared to the reference solution obtained by Turek et al. [18], while being computationally cheap and reasonably easy to implement.

Our model consists in a beam, which is a thin inextensible structure made out of linear elastic material. A key aspect is that we must account for large deflections and cannot use geometrically linear models, as their deflections have to be comparable to the beam thickness and thus to be very small. It is assumed that the beam is clamped at its leading edge, while the trailing one is free. In view of our application to biolocomotion, an arbitrary motion at the leading edge must be considered. This imposed motion is described by the position vector $\underline{x}_0(t)$ and the rotation angle $\alpha(t)$, that are both arbitrary functions of time, see Fig. 4. The beam is parametrized by its local deflection angle Θ and its longitudinal force T,

which are both functions of the arclength coordinate $s \in (0, 1)$ and time. The model equation is the extension of the one used in [55]. It is derived by considering force- and torque balance on a deformed beam element, yielding

$$\frac{\partial^2 T}{\partial s^2} - T\left(\frac{\partial \Theta}{\partial s}\right)^2 = -[p]^\pm \frac{\partial \Theta}{\partial s} - 2\eta \frac{\partial \Theta}{\partial s} \frac{\partial^3 \Theta}{\partial s^3} - \eta \left(\frac{\partial^2 \Theta}{\partial s^2}\right)^2 - \mu(\dot{\Theta} + \dot{\alpha})^2 - \frac{\partial [\tau]^\pm}{\partial s} \tag{24a}$$

$$\mu\ddot{\Theta} + \mu\ddot{\alpha} + \frac{\partial [p]^\pm}{\partial s} = -\eta \frac{\partial^4 \Theta}{\partial s^4} + \left(T + \eta\left(\frac{\partial \Theta}{\partial s}\right)^2\right) \frac{\partial^2 \Theta}{\partial s^2} + 2\frac{\partial T}{\partial s} \frac{\partial \Theta}{\partial s} + [\tau]^\pm \frac{\partial \Theta}{\partial s}, \tag{24b}$$

where

$$\mu = \frac{h \varrho_s}{\ell \varrho_f} \qquad \eta = \frac{E h^3}{12 \ell^3 \varrho_f u_\infty^2} \tag{25}$$

are the dimensionless density and rigidity, respectively. The external forces acting on the beam are the pressure jump across the beam $[p]^\pm$ and the viscous tensions $[\tau]^\pm$ on the surface. The Young's modulus, thickness, length and density are denoted by E, h, ℓ, ϱ_s, and the fluid density ϱ_f and the free-stream velocity $\underline{u}_\infty$ are used for normalization. Dots denote time derivatives. Note that unlike in our earlier work [56], Eqs. (24a)–(24b) account for the viscous tensions. The set of PDEs (24a)–(24b) has to be completed with the clamped-free boundary conditions:

$$\left.\begin{aligned} &\Theta = 0 \\ &\frac{\partial T}{\partial s} + \eta \frac{\partial^2 \Theta}{\partial s^2} \frac{\partial \Theta}{\partial s} = \mu(\ddot{x}_0 \cos\alpha + \ddot{y}_0 \sin\alpha + g\sin\alpha) - [\tau]^\pm \\ &T\frac{\partial \Theta}{\partial s} - \eta \frac{\partial^3 \Theta}{\partial s^3} = \mu(\ddot{y}_0 \cos\alpha - \ddot{x}_0 \sin\alpha + g\cos\alpha) + [p]^\pm \end{aligned}\right\} \quad \text{at } s = 0 \tag{26a}$$

$$\left.\begin{aligned} &T = 0 \\ &\frac{\partial \Theta}{\partial s} = 0 \\ &\frac{\partial^2 \Theta}{\partial s^2} = 0 \end{aligned}\right\} \quad \text{at } s = 1 \tag{26b}$$

where $\ddot{x}_0$ and $\ddot{y}_0$ are the components of the position vector $\underline{x}_0$ and g is the gravity, which acts always in negative y direction. From the local deflection angle Θ, the global deflection line $\underline{x}_s$ and velocity $\underline{u}_s$ can be obtained by computing:

$$\underline{x}_c(s) = \underline{x}_0 + \begin{pmatrix} \cos\alpha \int_0^s \cos\Theta\,ds - \sin\alpha \int_0^s \sin\Theta\,ds \\ \cos\alpha \int_0^s \sin\Theta\,ds + \sin\alpha \int_0^s \cos\Theta\,ds \end{pmatrix} \tag{27a}$$

$$\underline{u}_s(s) = \dot{\underline{x}}_0 + \begin{pmatrix} \cos\alpha(\int_0^s -\dot{\Theta}\sin\Theta\,ds - \dot{\alpha}\int_0^s \sin\Theta\,ds) - \sin\alpha(\int_0^s -\dot{\Theta}\sin\Theta\,ds + \dot{\alpha}\int_0^s \cos\Theta\,ds) \\ \cos\alpha(\int_0^s -\dot{\Theta}\sin\Theta\,ds + \dot{\alpha}\int_0^s \cos\Theta\,ds) + \sin\alpha(\int_0^s -\dot{\Theta}\sin\Theta\,ds - \dot{\alpha}\int_0^s \sin\Theta\,ds) \end{pmatrix}. \tag{27b}$$

4.2.2. Numerical solution

The set of Eqs. (24a)–(24b), (26a)–(26b), (27a)–(27b) are solved numerically using finite difference approximations of the differential operators. Special care has to be devoted to time integration. First, the equations are stiff and require an implicit time marching scheme. Second, the eigenvalues of the discrete (linearized) operator lie on the imaginary axis, which happens to be the stability limit of the second order Crank–Nicolson scheme which was used in previous works [56]. Hence we employ a second order backward differentiation scheme with variable time steps [57]:

$$w^{n+1} = \frac{(1+\xi)^2}{1+2\xi} w^n - \frac{\xi^2}{1+2\xi} w^{n-1} + \frac{1+\xi}{1+2\xi} \Delta t^n f(w^{n+1}) \tag{28}$$

where $\xi = \Delta t^n / \Delta t^{n-1}$. The vector $w = (\Theta\ \dot{\Theta})$ has been introduced to rewrite Eq. (24b) as a first order system and $f(w^{n+1})$ is the right-hand side of this first order system. All terms are treated implicitly and the inextensibility constraint (24a) is fulfilled at the new time level. The resulting non-linear system is solved using Newton–Raphson iterations, with a relative error below 10^{-10} as stopping criterion. The Jacobian is computed analytically and hard-coded in the solver, speeding up the computation. Typically, three iterations are performed until the stopping criterion is reached. The time step Δt is determined by the fluid.

4.2.3. Validation

Our numerical algorithm is verified to be of second order accuracy with respect to both Δx and Δt, by comparing with the solution on the finest grid. To study the model error, we compare our solver with the reference data for test case CSM3 proposed in [18]. This test considers a beam without surrounding fluid and without imposed motion. The motion is

Table 1
Results of the CSM3 test. The present results are obtained with the non-linear beam equation, while the reference solution is obtained by solving a full 2D continuum model.

	Frequency	x-displacement [$\times 10^{-3}$]		y-displacement [$\times 10^{-3}$]	
		mean	amplitude	mean	amplitude
Turek et al. [18]	1.0995	−14.3051	14.3057	−63.6077	65.1607
Present results	1.1012	−14.4442	14.4442	−63.3999	65.2898

driven by gravity. The normalized parameters (25) characterizing the solid are $\eta = 2.592 \cdot 10^{-2}$ and $\mu = 5.714 \cdot 10^{-2}$ and the normalized gravity is $g = 0.7$. We should point out that Young's modulus has to be corrected when passing to a plane stress state, $E = E_0/(1 - n^2)$, where E_0 is the usual Young's modulus and n is the Poisson ratio.

The results of the comparison are summarized in Table 1. The numerical resolution is sufficiently high for the model error to become dominant. Although our model is only one-dimensional, it agrees reasonably well with the reference solution.

4.3. Fluid–solid coupling

In order to couple the previously described fluid and solid models, two distinct tasks have to be performed. First, the fluid solver requires the current geometry of the problem, hence the mask function $\chi(t)$ has to be constructed from the current deflection line of the solid. Second, the fluid forces acting on the solid and determining its dynamics must be transferred from the Eulerian description of the fluid to the Lagrangian description of the solid. This requires interpolation.

The algorithm used to construct $\chi(\underline{x}, t)$ and $\underline{u}_s(\underline{x}, t)$ can be found in Appendix A.

To transfer the fluid forces we employ a regularized delta function for the Eulerian–Lagrangian transfer. Namely, we use the kernel proposed by Yang et al. [58]:

$$\delta(r) = \begin{cases} \frac{3}{8} + \frac{\pi}{32} - \frac{r^2}{4} & |r| \leq 0.5 \\ \frac{1}{4} + \frac{1-|r|}{8}\sqrt{-2 + 8|r| - 4r^2} - \frac{1}{8}\arcsin(\sqrt{2}(|r| - 1)) & 0.5 \leq |r| \leq 1.5 \\ \frac{17}{16} - \frac{\pi}{64} - \frac{3|r|}{4} + \frac{r^2}{8} + \frac{|r|-2}{16}\sqrt{16|r| - 4r^2 - 14} + \frac{1}{16}\arcsin(\sqrt{2}(|r| - 2)) & 1.5 \leq |r| \leq 2.5 \\ 0 & 2.5 \leq |r|, \end{cases}$$

where $r = x/\Delta x$. This kernel is designed to reduce spurious oscillations in the interpolated values, which is desirable for fluid–structure interaction. The two-dimensional function is constructed as tensor product of two functions of one variable,

$$\delta(\underline{x}) = \delta\left(\frac{x}{\Delta x}\right)\delta\left(\frac{y}{\Delta y}\right),$$

and the interpolated pressure on the interface then reads

$$p(\underline{x}) = \sum_j \sum_i p_{ij}\delta(\underline{x}_{ij} - \underline{x}). \tag{29}$$

The same technique is used to interpolate viscous tensions on the surface.

With the mask construction and the load transfer technique all the necessary tools to couple fluid and structure are available. In the present work we are concerned with solid materials that have a higher density than the surrounding fluid. Thus, we do not encounter the specific stability problems that arise if the solid density is too close to or below the fluid density. We can thus enjoy a fast sequential staggered coupling scheme that would be unstable for light structures due to the artificial added mass instability [14]. The algorithm used here reads

1. At time t^n, compute $\underline{u}^n$ and p^n from the given the vorticity field ω^n.
2. Construct the mask function χ^n and the solid velocity field $\underline{u}_s^n$ from the beam state at t^n.
3. Advance the fluid in time by solving Eq. (10) using the adaptive second order Adams–Bashforth scheme.
4. From ω^{n+1}, compute the pressure p^{n+1} on the surface of the beam given by its state at t^n.
5. Using the interpolation (29), transfer the Eulerian pressure fields p^n, p^{n+1} to the Lagrangian surface points on the beam.
6. Advance the beam in time by solving Eq. (28).
7. Set $t^n \rightarrow t^{n+1}$ and go to 1.

Because we use two different time marching schemes for the fluid and the solid, the resulting algorithm is only first order accurate. It could be improved by performing iterations. Yet, because of the influence of the penalization parameter ε, we can content ourselves with this algorithm.

4.4. Energy budget

A first physically relevant test is whether our coupled solver respects energy conservation. To this end, we consider a beam of unit length in a domain $\Omega = [0, 4] \times [0, 4]$. The resolution is $n_x \times n_y = 2048 \times 2048$ and the penalty parameter is $\varepsilon = 10^{-4}$. The test case is that of a beam immersed in a fluid that bends down due to the action of gravity. We test two Reynolds numbers, Re = 200 and 2000. As stated previously, the penalization method requires very small ε in the limit of small Re. Thus, we do not release the beam in a fluid at rest, because the Reynolds number at the beginning of the motion would be zero. Instead, a mean flow is forced and given some time to develop, before gravity is softly turned on at $t = 2.5$. This way, one can avoid dealing with an impulsively started motion.

The energy contained in the beam can be computed as the sum of the elastic, kinetic and potential energy:

$$E_{\text{solid}}^s = E_{\text{flex}}^s + E_{\text{kin}}^s + E_{\text{pot}}^s \tag{30}$$

$$E_{\text{flex}}^s = \frac{1}{2} \int_0^1 \eta \Theta_s^2 \, ds \tag{31}$$

$$E_{\text{kin}}^s = \frac{1}{2} \mu \int_0^1 \underline{u}_s \cdot \underline{u}_s \, ds \tag{32}$$

$$E_{\text{pot}}^s = \mu g \int_0^1 y \, ds. \tag{33}$$

The energy budget for the penalized Navier–Stokes equation can be obtained by dot-multiplying Eq. (2a) and integrating over the computational domain $\Omega = \mathbb{T}^2$:

$$\frac{\mathrm{d}}{\mathrm{d}t} E_{\text{kin}}^f = -\frac{1}{\text{Re}} \int_\Omega |\nabla \underline{u}|^2 \mathrm{d}\Omega - \int_\Omega \frac{\chi}{\varepsilon} (\underline{u} - \underline{u}_s) \cdot \underline{u} \mathrm{d}\Omega$$

$$\dot{E}_{\text{kin}}^f = -\frac{1}{\text{Re}} \int_\Omega \omega^2 \mathrm{d}\Omega - \int_\Omega \frac{\chi}{\varepsilon} \underline{u}_s (\underline{u} - \underline{u}_\infty - \underline{u}_s) \mathrm{d}\Omega$$

$$\qquad - \int_\Omega \frac{\chi}{\varepsilon} \underline{u}_\infty (\underline{u} - \underline{u}_s) \mathrm{d}\Omega - \int_\Omega \frac{\chi}{\varepsilon} (\underline{u} - \underline{u}_\infty - \underline{u}_s)^2 \mathrm{d}\Omega$$

$$\dot{E}_{\text{kin}}^f = \dot{E}_{\text{diss}}^f + \dot{E}_{\text{solid}}^f + \dot{E}_{\text{mean}}^f + \dot{E}_{\text{porous}}^f \tag{34}$$

where the penalization term has been split into three contributions: the energy that enters the fluid through the solid motion $\dot{E}_{\text{solid}}^f$ and the mean flow $\dot{E}_{\text{mean}}^f$, and the energy that is dissipated in the penalization term $\dot{E}_{\text{porous}}^f$, which can be interpreted as "porous dissipation".

For our numerical method, we require $\dot{E}_{\text{solid}}^f = \dot{E}_{\text{solid}}^s$, i.e. the coupling should not introduce artificial energy. Fig. 5 illustrates a snapshot of the vorticity field (left) and the time series of the different terms in Eq. (34). One can observe good agreement between $\dot{E}_{\text{solid}}^f$ and $\dot{E}_{\text{solid}}^s$, the relative L_2 difference is about 7%. The remaining difference is attributed to finite-size effects in the model, since the underlying solid model is one-dimensional, but it is extended to a thin, yet two-dimensional structure.

4.5. Quantitative validation

To validate our numerical method quantitatively, we consider the benchmark proposed by Turek et al. [18,19]. It consists of three stages, one of which has already been used to evaluate the model error of the beam equation. The remaining stages are the CFD test, in which a rigid obstacle is considered, and the FSI test with a flexible object. For each stage, three test cases are proposed, and here we consider only the CFD3 and FSI3 tests.

The setup is illustrated in Fig. 6. A circular cylinder of radius $R = 0.05$ with an appendage of length $\ell = 0.35$ and thickness $h = 0.02$ is immersed in a channel of size $L_x \times L_y = 2.5 \times 0.41$ with a parabolic velocity profile imposed at the inflow. The center of the cylinder is placed at $(0.2, 0.2)$, where $(0, 0)$ is in the bottom-left corner of the domain, which intentionally yields an asymmetric setup. In our approach, the channel walls as well as the obstacle are modeled with the penalization method. The walls have the same thickness as the beam structure. The inflow condition is met using the velocity sponge approach, see Eq. (17), which is imposed in Ω_{sp}.

We should note that in this setup, the distance between the object and the inflow is smaller than the upstream influence of the object itself. Thus, this configuration cannot be reproduced considering a long channel with a developed laminar flow, as it would be done experimentally.
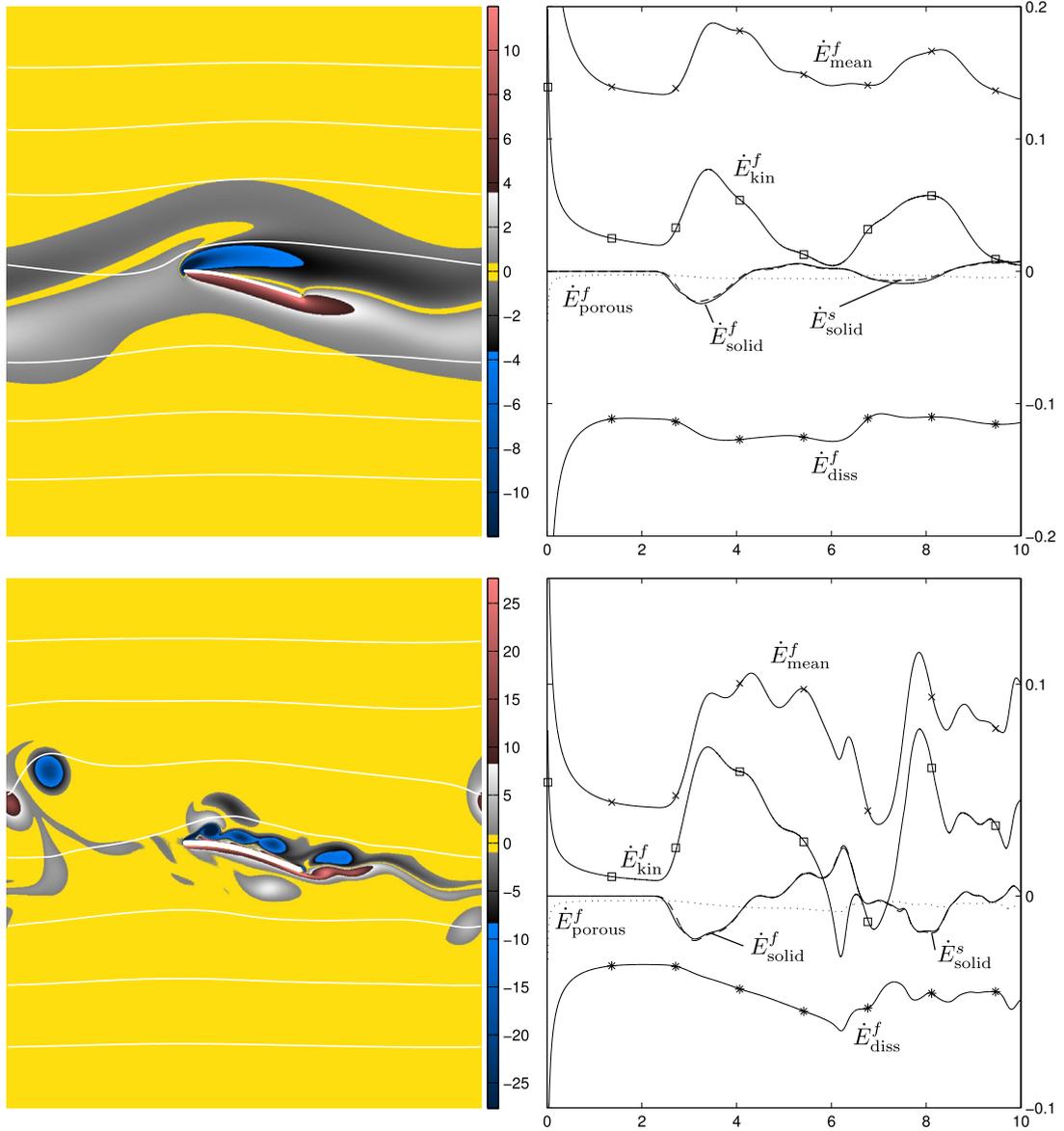
Fig. 5. Energy test case for the Reynolds numbers 200 (top) and 2000 (bottom). Left column: Snapshot of the flow field, shown is vorticity with super-imposed white streamlines. Right column: energy budget according to Eq. (34). The beam remains fixed until $t = 2$. The startup singularity is due to the impulsively started flow. The curves for $\dot{E}^f_{\mathrm{solid}}$ and $\dot{E}^s_{\mathrm{solid}}$ show good agreement in both cases, indicating a good conservation of energy.
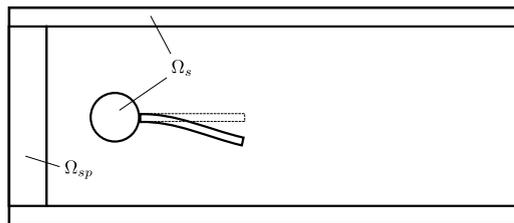
Fig. 6. Setup for the benchmark proposed by Turek et al. [18]. A cylinder with an attached appendage is immersed in a channel configuration. The velocity sponge technique is used to impose a parabolic velocity profile in Ω_{sp}. The appendage is either rigid (CFD test) or flexible (FSI test).

Table 2
Results of the CFD3 test with a fixed obstacle.

Resolution	ε	Lift				Drag				Frequency	
		Max		Min		Max		Min			
1300×288	$4 \cdot 10^{-3}$	423.66	−0.53%	−500.44	11.28%	512.62	15.18%	498.44	14.89%	4.46	1.47%
2600×576	$1 \cdot 10^{-3}$	392.47	−7.85%	−464.57	3.31%	475.70	6.88%	463.80	6.91%	4.41	0.33%
5200×1152	$2.5 \cdot 10^{-4}$	402.00	−5.62%	−452.64	0.65%	453.42	1.88%	442.14	1.92%	4.37	−0.58%
10400×2304	$6.3 \cdot 10^{-5}$	405.46	−4.80%	−446.48	−0.72%	447.29	0.50%	436.15	0.53%	4.36	−0.81%
Reference		425.92		−449.70		445.07		433.83		4.40	
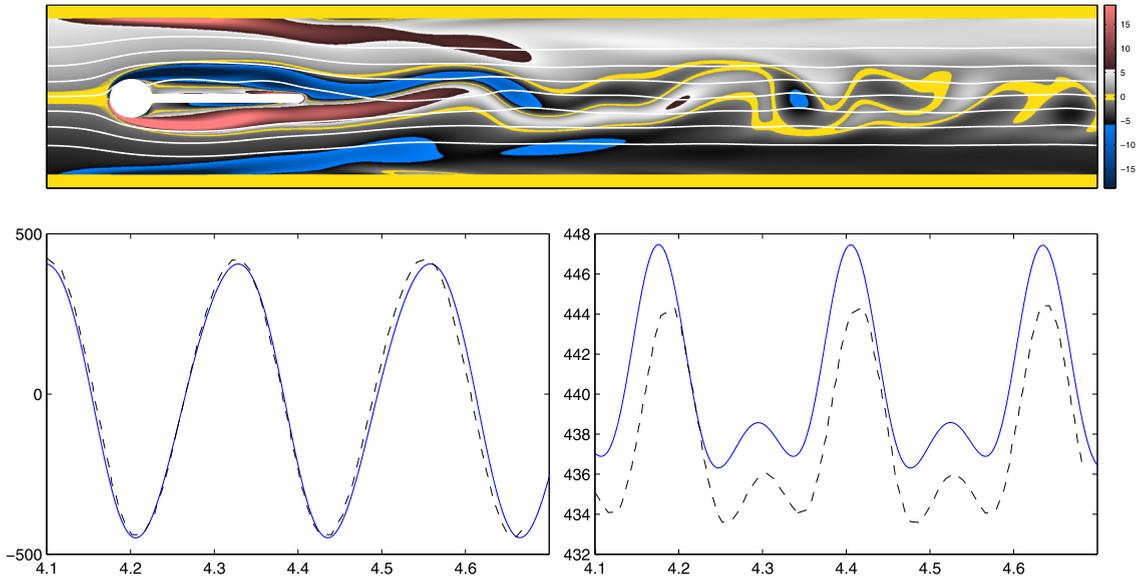
Fig. 7. Results for the CFD3 test with rigid appendage. Vorticity field with stream lines and the obstacle (top). Bottom: time series of the lift (left) and drag force (right) on the obstacle. Present results are represented by blue solid lines and the reference solution [18] by the dashed black line.

4.5.1. CFD test

First we consider the CFD3 test, where the obstacle remains fixed. The Reynolds number based on the length of the obstacle ℓ is $\mathrm{Re} = u_\infty \ell / \nu = 700$. The test is designed such that the flow reaches a periodic state which is independent of the initial condition. As the channel walls are modeled with the penalization method, they cover a finite area, which has to be taken into account when forcing the mean velocity. To force unit mean flow in the channel, u_∞ is corrected using the channel walls height h_{chan}, thus $u_\infty = 1 - 2h_{\mathrm{chan}}/L_y$.

When starting with a uniform flow as initial condition, the developing wake behind the obstacle requires a considerable amount of time to destabilize and to reach the final periodic state. For this reason, all present simulations are started with the flow field at the end of the destabilization phase. The quantities for comparison are the lift and drag force in the developed periodic regime and the fundamental frequency of the lift force, which corresponds to the lowest significant frequency present in the spectrum. The forces are represented by their minimum and maximum values during one period. Due to transient effects, the min/max values are averaged over the last periods, and simulations are run long enough for the width of the 95% confidence interval to be smaller than 0.1% of the predicted value. This results in a computational time of about 30 period times.

In Table 2, the present results for four levels of resolution are compared with the reference solution presented in [18]. The general accuracy of our method is satisfying, even though the maximum value of the lift force presents some noticeable difference with respect to the reference solution. In addition to the max/min values, the shape of the lift/drag curves are compared in Fig. 7, bottom. It can be observed that the curves are very similar. The drag curve exhibits an offset which is related to the smoothing layer in the χ-function, which can be interpreted as surface roughness. Due to the smoothing layer, our method generally overpredicts the drag force. On the other hand, the offset is only 0.5% of the mean value and thus rather small. The corresponding vorticity field during the periodic state is shown in Fig. 7, top.

4.5.2. FSI test

To conclude the quantitative validation we proceed to the FSI3 test case. Now, the appendage attached to the cylinder is flexible. As for the CFD3 test, some computational time can be saved by choosing a suitable initial condition. We use the same flow field as in the CFD3 test case, and the beam is initially at rest and undeformed. The non-dimensional

Table 3

Results of the FSI3 test. Present computations using four levels of resolution on the top, compared to various results from the literature. For orientation the average over the different publications (1)–(8), excluding the present ones, is also shown.

|	d_y [$\times 10^{-3}$]		F_{drag}		F_{lift}		f_0
	max	min	max	min	max	min	
$1300 \times 288,\ \varepsilon = 4 \cdot 10^{-3}$	36.47	−32.12	591.45	525.75	184.83	−189.29	5.89
$2600 \times 576,\ \varepsilon = 1 \cdot 10^{-3}$	36.32	−32.98	516.73	455.39	187.83	−185.73	5.56
$5200 \times 1152,\ \varepsilon = 2.5 \cdot 10^{-4}$	35.58	−32.59	487.87	437.93	190.50	−184.36	5.47
$10\,400 \times 2304,\ \varepsilon = 6.3 \cdot 10^{-5}$	35.63	−32.71	481.20	432.50	188.52	−181.30	5.44
(1) Turek [18]	36.37	−33.45	487.81	432.79	156.13	−151.31	5.47
(2) Schäfer [11,19]	36.73	−33.79	487.82	431.98	159.54	−155.86	5.82
(3) Rannacher [19]	35.89	−33.33	478.59	426.21	155.06	−150.34	5.42
(4) Turek [19]	36.46	−33.52	488.24	432.76	156.40	−151.40	5.47
(5) Breuer [19]	44.00	−41.00	507.00	428.00	204.90	−172.50	5.06
(6) Krafzyk [19]	36.58	−33.62	494.30	431.70	155.81	−152.19	5.50
(7) Wall [13,19]	30.45	−27.55	451.50	416.50	91.13	−86.07	5.30
(8) Bletzinger [19]	38.18	−35.08	503.02	446.78	169.76	−162.04	5.50
Average	36.75	−33.81	488.14	431.62	158.62	−150.78	5.46
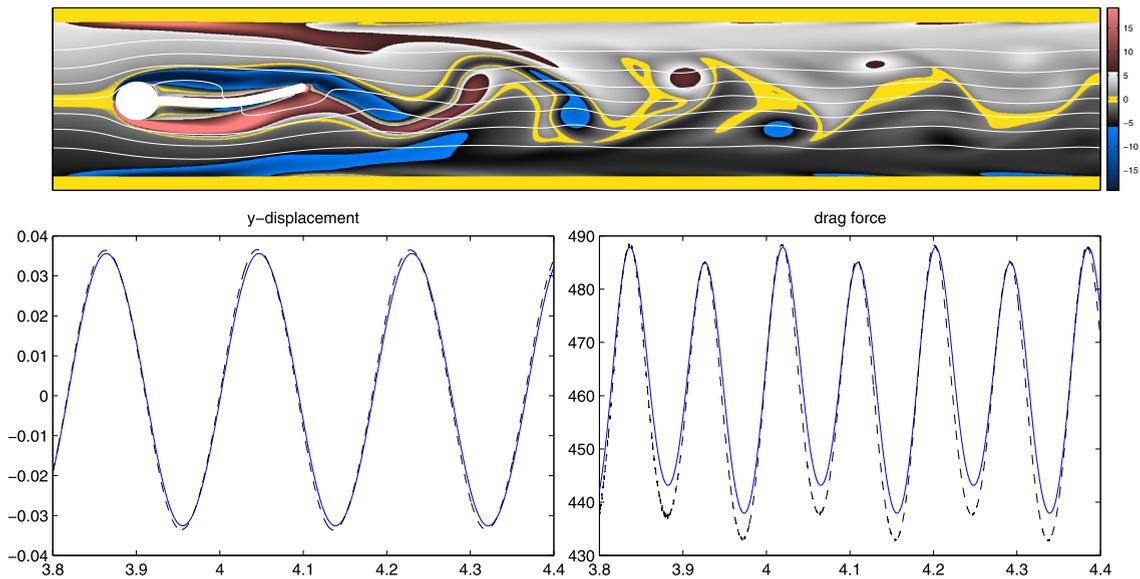
Fig. 8. Results for the FSI3 test with a flexible appendage. Top: snapshot of the vorticity field with the deformed obstacle. Bottom: time series of the y-displacement of the trailing edge and of the drag force acting on the entire obstacle.

parameters characterizing the solid are $\mu = 5.7143 \cdot 10^{-2}$ and $\eta = 2.5915 \cdot 10^{-2}$, cf. Eq. (25), and the Reynolds number is $\text{Re} = u_\infty \ell / \nu = 700$.

As for the CFD3 test, the quantitative comparison with results from the literature is done using the max/min values of the respective time series. The fundamental frequency f_0 is that of the y-displacement d_y of the trailing edge. In the present work, the max/min value are obtained as stated previously for the CFD3 test, but interestingly the FSI3 test requires a shorter computational time to settle into its final periodic state. Here, about 25 oscillations were computed. The max/min values for the displacement d_y, the drag and lift force and the frequency f_0 for this test are presented in Table 3. The different methods behind the numerical results are described in the respective papers, as listed in Table 3 and in the overview article [18].

For the y-displacement of the trailing edge, all methods except (5) and (7) agree to within 3.9%, and the present results fit well into this range of results. Unless otherwise stated, differences in this section are normalized with the averaged result. It is remarkable that the result for d_y does not significantly depend on the resolution level, as even the coarsest resolution yields an acceptable value. Compared to the initial computation from [18], present results differ by 0.6%.

For the drag force, the different methods are spread to a range over 7.5% of the averaged value. The drag computed with the present method is in excellent agreement for the two highest resolutions and again quite close to the original values from [18], with a difference of about 0.5%. The shapes of the time series of the displacement and the drag force are illustrated in Fig. 8 together with the original results presented in [18].
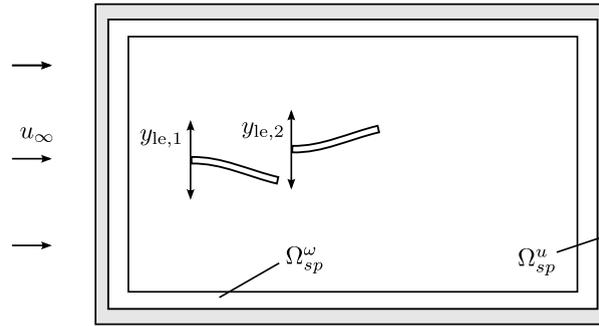
Fig. 9. Setup for the wing section configuration. The flexible foils perform a sinusoidal heaving motion, and the second one is optional. The mean fluid flow u_∞ is imposed. A mixture of a ω sponge and a u sponge is used. The ω sponge damps incoming vortices slowly, in order to minimize the upstream influence. Remaining vorticity is then removed by the u sponge.

Concerning the lift force, the range of results is broader and varies up to 40% for method (7), 30% for method (5) and 7.5% for method (8). The present method seems to overpredict the amplitude of oscillations in the lift force, and the difference in both max and min value is about 20% that of the average.

In conclusion we find good quantitative agreement with the results found in the literature, for the solid and fluid solver alone, as well as for the coupled FSI algorithm.

5. Application to insect flight

Since the present numerical method has been validated, it can now be applied to study simplified models for insect flight. These simulations are namely inspired by the experiments presented in [39]. The mechanical flapper considered therein is designed to perform heaving motion only. The wings are flexible and the resulting deflection passively imposes an angle of attack different from zero, which is why this configuration produces nonzero thrust.

In our two-dimensional approximation, this setup translates into a pure heaving motion of the wing section's leading edge. In [39], a mechanical flapper with one wing per side has been considered, and it is thus a first step to apply our method to this setup. As a second step, we add a second wing section and study the interaction between the fore- and hindwing.

The experimental setup is intrinsically three dimensional, and the present two-dimensional simulations should be viewed as a preparatory study rather than reproducing the exact physics. Our method is not intrinsically limited to two dimensions, in fact, an extension to 3D is rather straightforward, especially since the experiment aims at studying the chordwise flexibility only, which can indeed be done with our one-dimensional solid model.

For the remainder of this section, we fix the material properties of the solid to $\eta = 5.5284$ and $\mu = 2.8902$ and the Reynolds number based on the unity mean flow to $\mathrm{Re} = u_\infty \ell / \nu = 300$, where ℓ is the chord length, which also is unity. The first eigenfrequency of the beam is $f_0 = 3.516/2\pi \sqrt{\eta/\mu} = 0.77$. The mean flow is imposed to be unity, $u_\infty = 1$. The setup is illustrated in Fig. 9.

5.1. One wing section

For the single wing section, we are interested in the relation between the heaving frequency f and the thrust generated by the wing. The foil follows the imposed motion $y_{\mathrm{le},1} = y_0 + y_f \sin(2\pi f t)$. The amplitude of the heaving motion is $y_f = 0.5$ and y_0 is half the domain size, such that the mean position is on the center line of the domain. The domain size is $L_x \times L_y = 10 \times 10$ and the x-coordinate of the leading edge is $x_0 = 2$.

Special care is devoted to removing the wake at the borders of the domain. First, a vorticity sponge is applied in Ω_{sp}^ω, according to Eq. (16). The penalization parameter here is $\varepsilon_{sp}^\omega = 0.3$, which results in a relatively weak damping. The intention is to allow a traveling pair of vortices to enter the sponge collectively, since it may otherwise happen that one vortex is destroyed before the second one enters the sponge – in this case, the induced travel velocity reduces to zero and the second vortex is left orphaned in the computational domain. The vorticity sponge has a thickness of two chord lengths, which is large enough for dipoles to enter it entirely. Since the damping is relatively weak, it is possible that some remaining vorticity travels through the sponge. This is avoided by placing a velocity sponge around the vorticity sponge, see Ω_{sp}^u in Fig. 9. The penalization parameter here is $\varepsilon_{sp}^u = 10^{-3}$, and the velocity $\underline{u}_0$ in Eq. (17) is equal to the mean velocity u_∞.

The improved wake cancellation and the nonzero mean flow are the key differences to previously published preliminary material [59]. Also, the present simulations use higher resolution, smaller penalization parameters and compute more strokes, which is why present results are an improvement of the results in [59]. The resolution and corresponding penalization parameters can be found in Table 4(a).

The varying physical parameters for the study are summarized in Table 4(b). We systematically investigate the influence of the parameter f by simulating 13 different values, where each simulation spans N strokes.

Table 4

Parameters for the single foil configuration.

(a) Numerical parameters for the simulations.

		Level 1	Level 2	Level 3
Resolution	$n_x \times n_y$	768×768	1536×1536	3072×3072
Penalty parameter	ε	10^{-3}	$2.5 \cdot 10^{-4}$	$6.25 \cdot 10^{-5}$

(b) Physical parameters: the heaving frequency normalized by the first resonant and the number of strokes computed in each simulation.

f/f_0	1.29	1.19	1.08	1.00	0.97	0.92	0.86	0.80	0.76	0.65	0.54	0.43	0.32
N	25	25	25	25	25	25	25	25	25	25	20	12	9
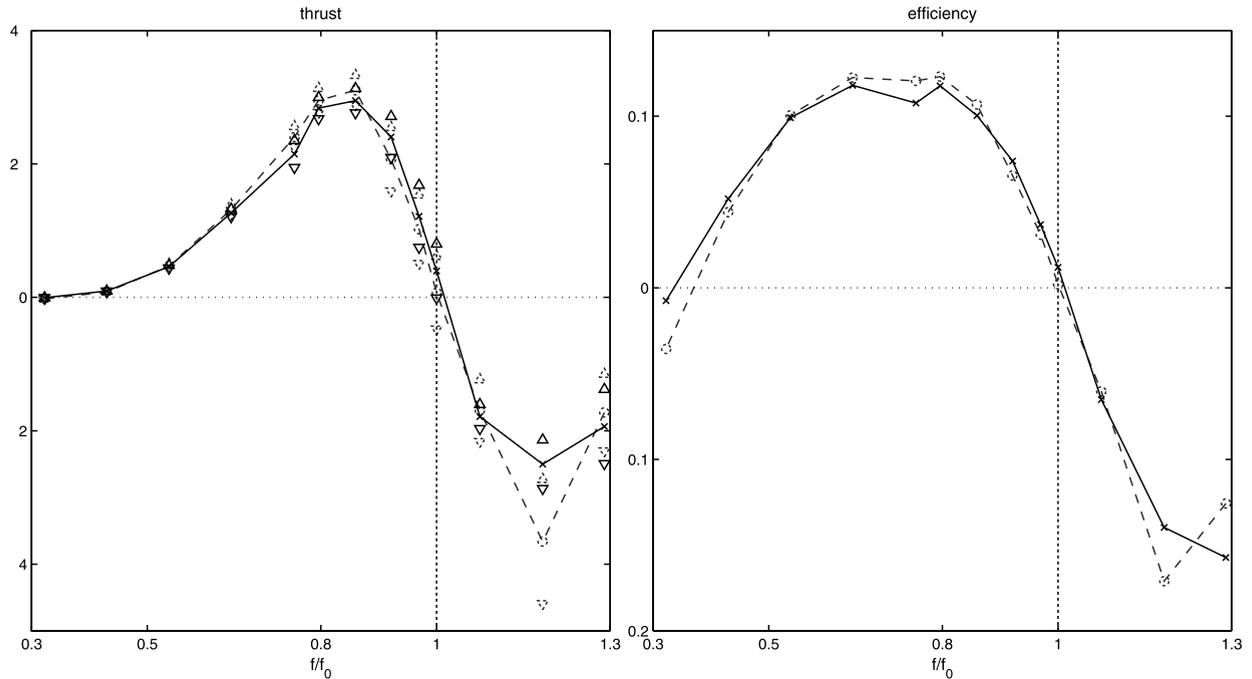
Fig. 10. Thrust (left) and propulsive efficiency (right) as a function of the reduced heaving frequency f/f_0. Curves for resolution level 2 (dashed) and 3 (solid). Triangles indicate the 95% confidence interval, vertical dashed line indicates the first resonant. The maximum thrust occurs at $f/f_0 = 0.86$, which lies also in the region of maximum efficiency of about 12.5%.

For low frequencies, the flow exhibits a strongly periodic behavior, which is also reflected in the hydrodynamic forces. Therefore, fewer strokes have been computed in these cases, the number of strokes was selected with the help of the coarsest level computation. With increasing frequency, the flow becomes more erratic and it gets increasingly difficult to estimate the mean values for the hydrodynamic forces. Long time simulations at the coarsest level indicate that no strictly periodic state is reached. The number of strokes is thus dictated by the computational cost and limited to $N = 25$ strokes for the higher resolution runs.

The main results are illustrated in Fig. 10. For the lowest frequency, we observe that the heaving foil produces virtually zero thrust, which is the same for both resolutions. The thrust increases with increasing frequency, until it reaches its maximum between 0.8 and 0.9. It then rapidly drops for higher frequencies and reverses its sign when passing the resonant frequency. The fact that maximum thrust is reached below the structural resonant is consistent with the main finding from [39]. Contrary to the thrust force, the propulsion efficiency exhibits a broader maximum between 0.6 and 0.8, which is about 12.5%. The efficiency is computed as the ratio of input power, i.e. the work of the external forces $[p]^{\pm}$ and $[\tau]^{\pm}$, and the output power $F_{\text{thrust}} \cdot u_{\infty}$.

The direct comparison of the two resolutions indicates that the main source of uncertainty is the erratic nature of the flow rather than the discretization error.

A possible explanation for the rapid decrease in thrust beyond $f/f_0 = 0.86$ is offered in Fig. 11. The trailing edge deflection angle increases with increasing frequency, and it becomes greater than 90° shortly before $f/f_0 = 0.86$. The deflection angle for higher frequencies is appreciably greater than 90°, which indicates that the trailing edge actually points forward, which also directs fluid momentum in this direction and hence decreases the thrust.
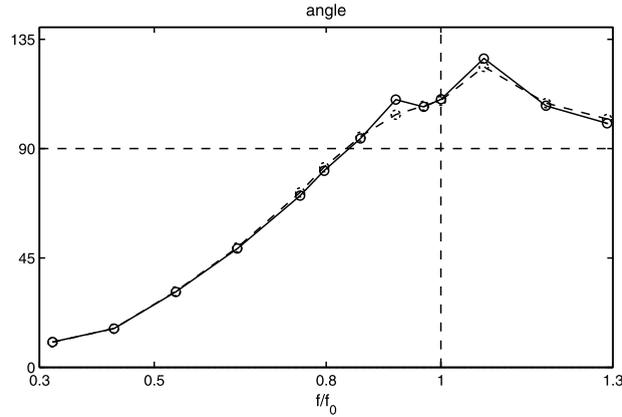
Fig. 11. Maximum trailing edge deflection angle (averaged over strokes) as a function of f/f_0 for level 2 (dashed) and level 3 resolution (solid). Horizontal and vertical dashed lines mark 90° and the resonant frequency, respectively.

5.2. Two wing sections

It is an appealing feature of the present method that it can be extended to take several obstacles into account, a task that may present some extra difficulties when applying body-fitted grids to this problem. Though a detailed investigation is beyond the scope of this paper, we present some results for two wing sections in interaction. Thus, we place a second beam behind the first one, that follows the same imposed motion, but with a phase shift δ, $y_{\mathrm{le},2} = y_0 + y_f \sin(2\pi f t + \delta)$. We fix the heaving frequency to $f/f_0 = 0.80$, which is the last datapoint before the maximum thrust in Fig. 10.

Inspired by the paper of Usherwood and Lehmann [60], we investigate the influence of the phase shift on the wake structures for two values of δ, namely $\delta = 0.75\pi$ and $\delta = 1.25\pi$. As we are interested in the wake structure, the computational domain is $L_x \times L_y = 30 \times 20$ chord lengths, and the resolution is 4608×3072 nodes. The penalization parameter is $\varepsilon = 6.25 \cdot 10^{-5}$. The same sponge technique as for the single foils is applied, see Fig. 9.

In Fig. 12 (left column), the instantaneous vorticity fields for both cases are shown. They reveal that the $\delta = 1.25\pi$ case (bottom) yields a top/bottom asymmetry in the wake, as stronger dipoles are shed downwards while only little vortices are shed on the upper part of the wake. The time averaged vorticity field on the other hand reveals a more symmetric structure, indicating that the wake settles in either direction during the simulation. The time averaged vorticity in the $\delta = 0.75\pi$ case (top, right) also shows a symmetric structure, but the vorticity is overall spread less widely.

6. Conclusion and extension to three-dimensional flows

In this paper, we presented a novel approach for the numerical simulation of fluid–structure interaction. On the fluid side, the volume penalization method has been used for modeling the flow past moving and flexible objects, and we have shown that the approach of moving a smooth mask function yields good quantitative results. Compared to [31], this approach is more efficient and allows the simulation of flexible obstacles, but requires slightly more implementation effort. We have presented a one-dimensional model for thin flexible structures which allows to impose an arbitrary motion at the leading edge. Comparison with results from the literature confirms its validity. Its main advantage is its low computational cost and ease of implementation, compared to finite elements solution of actual two-dimensional continua. The fluid–solid coupling algorithm has been shown to respect energy conservation, i.e. it does not introduce unphysical energy to the coupled system. The quantitative validation with a widely-used benchmark shows good agreement and concludes the validation of the algorithm. Our simulations are confined to two-dimensional cases here, but the method can easily be extended to three dimensions. The application to simplified models for insect flight finally shows that the present method can reproduce some features of the experimental configuration presented in [39].

The extension of the presented method for simulating fluid–structure interaction in 3D flows is straightforward. The vorticity formulation can be used, but the advantage of having a scalar valued equation is lost as the vorticity becomes a vector valued quantity. Thus the primitive variable formulation (2a)–(2c) is typically preferred. The incompressibility of the velocity field can thus be imposed by applying a Riesz projector in Fourier space to the non-linear and the penalization terms and thus pressure is eliminated, a standard technique in the spectral community [48,47]. Denoting both terms with $\mathcal{N} = -\underline{\omega} \times \underline{u} - \frac{\chi}{\varepsilon}(\underline{u} - \underline{u}_s)$, such a scheme reads

$$\partial_t \underline{u} - \frac{1}{\mathrm{Re}} \nabla^2 \underline{u} = \underline{\mathcal{N}} - \nabla \frac{\nabla \cdot \underline{\mathcal{N}}}{\nabla^2}$$

Apart from this modification the remaining parts of the described algorithm can be directly used in 3D, keeping in mind that the proposed solid model only allows 1D flexibility. First results of 3D FSI simulations of pitching flexible foils and will be presented elsewhere.
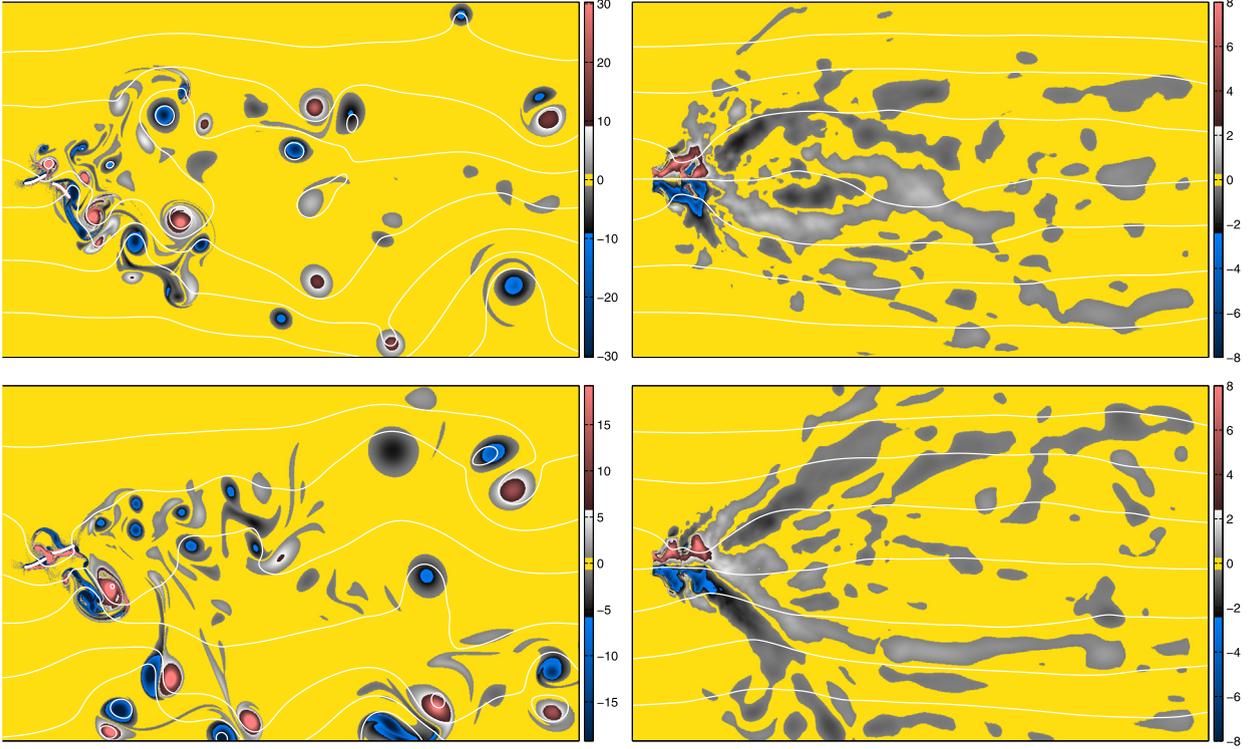
Fig. 12. Two heaving foils in tandem, instantaneous (left) and time-averaged (right) vorticity fields. For visibility, only part of the computational domain is shown. The phase shift is $\delta = 0.75\pi$ (top) and $\delta = 1.25\pi$ (bottom).

Acknowledgements

Appendix A. Construction of the χ-function for the flexible beam

The way the mask function χ is constructed is inspired by the discretization of the beam equation. A small rectangle is drawn between two discrete points on the beam, aligned with the angle between them. The χ-function depends then only on the distance normal to the center line. Gaps between two rectangles of different angle are filled with circle segments. The mask construction is illustrated in Fig. 13. The solid velocity field $\underline{u}_s$ is constant in the $\underline{e}_n$-direction and linearly interpolated in the $\underline{e}_t$-direction. To summarize, the algorithm reads

1. Given the n_s center line coordinates $\underline{x}_c^{(i)} = (x_c^{(i)}, y_c^{(i)})$ from Eq. (27a) and the local deflection angles $\Theta^{(i)}$.
2. For each beam segment $(\underline{x}_c^{(i)}, \underline{x}_c^{(i+1)})$
   (a) Determine the indices of the rectangular box B of Eulerian grid points containing the segment: $B = [\text{ix}_{min}, \text{ix}_{max}] \times [\text{iy}_{min}, \text{iy}_{max}]$ with

$$\text{ix}_{min} = \text{floor}\left(\frac{\min(x_c^{(i)}, x_c^{(i+1)}) - t^\star}{\Delta x}\right) \qquad \text{iy}_{min} = \text{floor}\left(\frac{\min(y_c^{(i)}, y_c^{(i+1)}) - t^\star}{\Delta y}\right)$$

$$\text{ix}_{max} = \text{ceil}\left(\frac{\max(x_c^{(i)}, x_c^{(i+1)}) + t^\star}{\Delta x}\right) \qquad \text{iy}_{max} = \text{ceil}\left(\frac{\max(y_c^{(i)}, y_c^{(i+1)}) + t^\star}{\Delta y}\right)$$

where $t^\star$ is the thickness of the beam plus the smoothing layer.
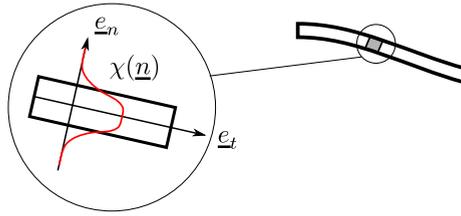
Fig. 13. Schematic illustration of mask generation. The width of the smoothing layer is exaggerated.

   (b) For each point P in B

      i. Compute the point's coordinate in the coordinate system aligned with the beam segment, $\underline{x}_p = n_p \underline{e}_n + t_p \underline{e}_t$, cf. Fig. 13

$$n_p = \cos\left(\Theta^{(i)}\right)x_p - \sin\left(\Theta^{(i)}\right)y_p$$
$$t_p = \sin\left(\Theta^{(i)}\right)x_p + \cos\left(\Theta^{(i)}\right)y_p$$

      ii. Set the value of $\chi = H(n_p)$ according to Eq. (23). If χ already has a value at this position, overwrite it if the new value is larger.

      iii. If χ at this position is set, then set the linearly interpolated velocity $\underline{u}_s = \underline{u}_s^{(i)} + \frac{t_p}{\Delta s}(\underline{u}_s^{(i+1)} - \underline{u}_s^{(i)})$ here.

  3. For each hinge point located between two segments, $\underline{x}_c^{(i)}$, fill the remaining gap with a circle segment.

In the last step, it should be noted that one cannot use full circles to fill the gaps as with increasing resolution of the beam, one circle may fill several gaps, resulting in wrong velocity vectors.

References

[1] M. Gazzola, W.M.V. Rees, P. Koumoutsakos, C-start: optimal start of larval fish, J. Fluid Mech. (2012) 1–14.

[2] M. Bergmann, A. Iollo, Modeling and simulation of fish-like swimming, J. Comput. Phys. 230 (2011) 329–348.

[3] T. Nakata, H. Liu, Aerodynamic performance of a hovering hawkmoth with flexible wings: a computational approach, Proc. – Royal Soc., Biol. Sci. 279 (1729) (2012) 722–731.

[4] T. Nakata, H. Liu, A fluid–structure interaction model of insect flight with flexible wings, J. Comput. Phys. 231 (4) (2012) 1822–1847.

[5] K. Takizawa, C. Moorman, S. Wright, T. Tezduyar, Computer modeling and analysis of the orion spacecraft parachutes, in: H.-J. Bungartz, M. Mehl, M. Schäfer (Eds.), Fluid Structure Interaction II, in: Lect. Notes Comput. Sci. Eng., vol. 73, Springer, Berlin, Heidelberg, 2010, pp. 53–81.

[6] Y. Kim, C.S. Peskin, 3-d parachute simulation by the immersed boundary method, Comput. Fluids 38 (6) (2009) 1080–1090.

[7] B. Hübner, E. Walhorn, D. Dinkler, A monolithic approach to fluid–structure interaction using space–time finite elements, Comput. Methods Appl. Mech. Eng. 193 (23–26) (2004) 2087–2104.

[8] P. Ryzhakov, R. Rossi, S. Idelsohn, E. Oñate, A monolithic Lagrangian approach for fluid–structure interaction problems, Comput. Mech. 46 (6) (2010) 883–899.

[9] C. Michler, S. Hulshoff, E. Van Brummelen, R. De Borst, A monolithic approach to fluid–structure interaction, Comput. Fluids 33 (5) (2004) 839–848.

[10] P. Causin, J. Gerbeau, F. Nobile, Added-mass effect in the design of partitioned algorithms for fluid–structure problems, Comput. Methods Appl. Mech. Eng. 194 (2005) 4506–4527.

[11] D. Sternel, M. Schäfer, M. Heck, S. Yigit, Efficiency and accuracy of fluid–structure interaction simulations using an implicit partitioned approach, Comput. Mech. 43 (2008) 103–113.

[12] M. Storti, N. Nigro, R. Paz, L. Dalcon, Strong coupling strategy for fluid–structure interaction problems in supersonic regime via fixed point iteration, J. Sound Vib. 320 (2009) 859–877.

[13] U. Küttler, W. Wall, Fixed-point fluid–structure interaction solvers with dynamic relaxation, Comput. Mech. 43 (2008) 61–72.

[14] C. Foerster, W. Wall, E. Ramm, Artificial added mass instabilities in sequential staggered coupling of nonlinear structures and incompressible viscous flows, Comput. Methods Appl. Mech. Eng. 196 (2007) 1278–1293.

[15] J.P. Gomes, S. Yigit, H. Lienhart, M. Schäfer, Experimental and numerical study on a laminar fluid–structure interaction reference test case, J. Fluids Struct. 27 (1) (2011) 43–61.

[16] T. Wick, Fluid–structure interactions using different mesh motion techniques, Comput. Struct. 89 (13–14) (2011) 1456–1467.

[17] K. Bathe, H. Zhang, A mesh adaptivity procedure for CFD and fluid–structure interactions, Comput. Struct. 87 (2009) 604–617.

[18] S. Turek, J. Hron, Proposal for numerical benchmarking of fluid–structure interaction between an elastic object and laminar incompressible flow, in: H. Bungartz, M. Schäfer (Eds.), Fluid–Structure Interaction: Modelling, Simulation, Optimisation, in: Lect. Notes Comput. Sci. Eng., vol. 53, Springer, Berlin, Heidelberg, 2006, pp. 371–385.

[19] S. Turek, J. Hron, M. Razzaq, H. Wobker, M. Schäfer, Numerical benchmarking of fluid–structure interaction: a comparison of different discretization and solution approaches, in: H.-J. Bungartz, M. Mehl, M. Schäfer (Eds.), Fluid Structure Interaction II, in: Lect. Notes Comput. Sci. Eng., vol. 73, Springer, Berlin, Heidelberg, 2010, pp. 413–424.

[20] E.H. Dowell, K.C. Hall, Modeling of fluid–structure interaction, Annu. Rev. Fluid Mech. 33 (2001) 445–490.

[21] C.S. Peskin, Numerical analysis of blood flow in the heart, J. Comput. Phys. 25 (1977) 220–252.

[22] C.S. Peskin, The immersed boundary method, Acta Numer. 11 (2002) 479–517.

[23] R. Mittal, G. Iaccarino, Immersed boundary methods, Annu. Rev. Fluid Mech. 37 (2005) 239–261.

[24] X. Zhu, G. He, X. Zhang, Numerical study on hydrodynamic effect of flexibility in a self-propelled plunging foil, Comput. Fluids 97 (2014) 1–20.

[25] L. Zhu, C.S. Peskin, Simulation of a flapping flexible filament in a flowing soap film by the immersed boundary method, J. Comput. Phys. 179 (2002) 452–468.

[26] P.D. Yeh, A. Alexeev, Free swimming of an elastic plate plunging at low Reynolds number, Phys. Fluids 26 (5) (2014) (1994-present).

[27] G. Hou, J. Wang, A. Layton, Numerical methods for fluid–structure interaction – a review, Commun. Comput. Phys. 12 (2012) 337–377.

[28] P. Angot, C. Bruneau, P. Fabrie, A penalization method to take into account obstacles in incompressible viscous flows, Numer. Math. 81 (1999) 497–520.
[29] G. Carbou, P. Fabrie, Boundary layer for a penalization method for viscous incompressible flow, Adv. Differ. Equ. 8 (2003) 1453–2480.
[30] R. Nguyen van yen, D. Kolomenskiy, K. Schneider, Approximation of the Laplace and Stokes operators with Dirichlet boundary conditions through volume penalization: a spectral viewpoint, Numer. Math. 128 (2) (2014) 301–338, http://dx.doi.org/10.1007/s00211-014-0610-8.
[31] D. Kolomenskiy, K. Schneider, A Fourier spectral method for the Navier–Stokes equations with volume penalization for moving solid obstacles, J. Comput. Phys. 228 (2009) 5687–5709.
[32] D. Kolomenskiy, H.K. Moffatt, M. Farge, K. Schneider, Two- and three-dimensional numerical simulations of the clap–fling–sweep of hovering insects, J. Fluids Struct. 27 (5–6) (2011) 784–791.
[33] Y. Elimelech, D. Kolomenskiy, S.B. Dalziel, H.K. Moffatt, Evolution of the leading edge vortex over an accelerating rotating wing, Proced. IUTAM 7 (2013) 233–242.
[34] G. Bimbard, D. Kolomenskiy, O. Bouteleux, J. Casas, R. Godoy-Diana, Force balance in the take-off of a pierid butterfly: relative importance and timing of leg impulsion and aerodynamic forces, J. Exp. Biol. 216 (2013) 3551–3563.
[35] W. Shyy, H. Aono, S.K. Chimakurthi, P. Trizila, C.-K. Kang, C.E.S. Cesnik, H. Liu, Recent progress in flapping wing aerodynamics and aeroelasticity, Prog. Aerosp. Sci. 46 (7) (2010) 284–327.
[36] G. Du, M. Sun, Effects of wing deformation on aerodynamic forces in hovering hoverflies, J. Exp. Biol. 213 (13) (2010) 2273–2283.
[37] L. Zheng, T.L. Hedrick, R. Mittal, Time-varying wing-twist improves aerodynamic efficiency of forward flight in butterflies, PLoS ONE 8 (1) (2013) e53060.
[38] D. Lentink, S.R. Jongerius, N.L. Bradshaw, The scalable design of flapping micro-air vehicles inspired by insect flight, in: D. Floreano, J.-C. Zufferey, M.V. Srinivasan, C. Ellington (Eds.), Flying Insects and Robots, Springer, Berlin, Heidelberg, 2010, pp. 185–205.
[39] S. Ramananarivo, R. Godoy-Diana, B. Thiria, Rather than resonance, flapping wing flyers may play on aerodynamics to improve performance, Proc. Natl. Acad. Sci. USA 108 (2011) 5964–5969.
[40] S. Michelin, S.G. Llewellyn Smith, Resonance and propulsion performance of a heaving flexible wing, Phys. Fluids 21 (2009) 071902-1–071902-15.
[41] S. Alben, Simulating the dynamics of flexible bodies and vortex sheets, J. Comput. Phys. 228 (2009) 2587–2603.
[42] C.-K. Kang, H. Aono, C.E.S. Cesnik, W. Shyy, Effects of flexibility on the aerodynamic performance of flapping wings, J. Fluid Mech. 689 (2011) 32–74.
[43] K.-B. Lee, J.-H. Kim, C. Kim, Aerodynamic effects of structural flexibility in two-dimensional insect flapping flight, J. Aircr. 48 (3) (2011) 894–909.
[44] J. Pederzani, H. Haj-Hariri, A numerical method for the analysis of flexible bodies in unsteady viscous flows, Int. J. Numer. Methods Eng. 68 (10) (2006) 1096–1112.
[45] M. Vanella, T. Fitzgerald, S. Preidikman, E. Balaras, B. Balachandran, Influence of flexibility on the aerodynamic performance of a hovering wing, J. Exp. Biol. 212 (2009) 95–105.
[46] L.A. Miller, C.S. Peskin, Flexible clap and fling in tiny insect flight, J. Exp. Biol. 212 (2009) 3076–3090.
[47] C. Canuto, M.Y. Hussaini, A. Quarteroni, T. Zang, Spectral Methods in Fluid Dynamics, Springer Verlag, 1986.
[48] C. Canuto, M.Y. Hussaini, A. Quarteroni, T.A. Zang, Spectral Methods – Evolution to Complex Geometries and Applications to Fluid Dynamics, Springer, 2007.
[49] K. Schneider, D. Kolomenskiy, E. Deriaz, Is the CFL condition sufficient? Some remarks, in: C.A. de Moura, C.S. Kubrusly (Eds.), The Courant–Friedrichs–Lewy (CFL) Condition. 80 Years After Its Discovery, 2013, pp. 139–146.
[50] K. Schneider, Numerical simulation of the transient flow behaviour in chemical reactors using a penalisation method, Comput. Fluids 34 (2005) 1223–1238.
[51] M. Uhlmann, An immersed boundary method with direct forcing for the simulation of particulate flows, J. Comput. Phys. 209 (2005) 448–476.
[52] C. Jause-Labert, Simulation numerique d'ecoulements turbulents en rotation, confinement et forcage a l'aide d'une methode de penalisation, Ph.D. thesis, Ecole centrale de Lyon, 2012.
[53] A. Monin, A. Yaglom, Statistical Fluid Mechanics: Mechanics of Turbulence, MIT Press, Cambridge, MA, 1971.
[54] D. Wan, S. Turek, L. Rivkind, An efficient multigrid FEM solution technique for incompressible flow with moving rigid bodies, in: M. Feistauer, V. Dolejsi, P. Knobloch, K. Najzar (Eds.), Numerical Mathematics and Advanced Applications, ENUMATH 2003, Prague, Springer, ISBN 3-540-21460-7, 2003, pp. 844–853.
[55] S. Michelin, S.L. Smith, B. Glover, Vortex shedding model of a flapping flag, J. Fluid Mech. 617 (2008) 1–10.
[56] T. Engels, D. Kolomenskiy, K. Schneider, J. Sesterhenn, Two-dimensional simulation of the fluttering instability using a pseudospectral method with volume penalization, Comput. Struct. 122 (2012) 101–112.
[57] J. Becker, A second order backward difference method with variable steps for a parabolic problem, BIT Numer. Math. 38 (1998) 644–662.
[58] X. Yang, X. Zhang, Z. Li, G.-W. He, A smoothing technique for discrete delta functions with application to immersed boundary method in moving boundary simulations, J. Comput. Phys. 228 (2009) 7821–7836.
[59] D. Kolomenskiy, T. Engels, K. Schneider, Numerical modelling of flexible heaving foils, J. Aero Aqua Bio-mech. 3 (2013) 22–28.
[60] J.R. Usherwood, F.-O. Lehmann, Phasing of dragonfly wings can improve aerodynamic efficiency by removing swirl, J. R. Soc. Interface 5 (2008) 1303–1307.