High Performance Computing with Spectral Element Methods

R. Bouffanais N. Fiétier J. Latt M. Deville*
School of Engineering
Laboratory of Computational Engineering
École Polytechnique Fédérale de Lausanne
Lausanne, Switzerland

Abstract

The two lectures are devoted to the spectral element method (SEM) applied to the Navier–Stokes equations for viscous incompressible Newtonian fluids. The first presentation describes the state-of-the-art algorithms for direct numerical simulations. The space discretization relies on the $\mathbb{P}_N - \mathbb{P}_{N-2}$ approximation while the time integrators use an implicit/explicit approach. Solvers and preconditioners are based on the conjugate gradient method. The pressure computation is handled by the Uzawa algorithm which may be preconditioned in several ways. The case of complex geometries leads to the use of domain decomposition methods and various preconditioning techniques. The second lecture will focus on recent developments, namely the arbitrary Lagrangian-Eulerian formulation and the method of moving grid. Two examples are considered: free surface flows and the interaction between a viscous fluid and a Hookean elastic solid. As a final application, large-eddy simulation is analyzed and new models are proposed. The lecture ends with practical remarks on the numerical implmenetation and parallelization of spectral element codes.

Keywords: spectral element, ALE formulation, LES, ADM

AMS Subject Classification: 65M70

1 SEM State-of-the-Art

1.1 The continuous Navier–Stokes equations

The governing equations describing the flow of a viscous Newtonian incompressible fluid of density ρ and dynamic viscosity μ are given by the Navier–Stokes equations

$$\rho \frac{D\boldsymbol{v}}{Dt} = -\boldsymbol{\nabla}\, p + \mu\, \Delta \boldsymbol{v} + \rho \boldsymbol{f} \tag{1}$$

with the incompressibility constraint coming from the mass conservation

$$\operatorname{div} \boldsymbol{v} = 0 \tag{2}$$

In Eq. (1), the symbol D/Dt denotes the material time derivative

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + \boldsymbol{v} \cdot \nabla, \tag{3}$$

where $\boldsymbol{v}$ is the velocity field in the Eulerian representation and p the pressure. The body force is denoted by $\boldsymbol{f}$.

1.1.1 Boundary Conditions

The continuum mechanics hypothesis implies that the viscous fluid sticks to the wall. Therefore, one imposes the no-slip wall condition

$$\boldsymbol{v} = \boldsymbol{v}_w, \tag{4}$$

$\boldsymbol{v}_w$ being the prescribed wall velocity.

An interesting class of fluid flow problems is concerned with free surface flows, where a viscous fluid is directly in contact with an essentially inviscid fluid such as air. Some industrial examples are coating flows, extrusion, and crystal growth. These problems are highly nonlinear because the shape of the free surface, where special conditions are to be applied, is also part of the solution itself. Let us introduce the Cauchy principle where the density of contact forces represented by the stress vector $\boldsymbol{t}$ depends on the unit normal $\boldsymbol{n}$ to the surface $\partial\Omega(t)$ at position $\boldsymbol{x}$. More precisely, there exists a tensor field such that

$$\boldsymbol{t}(\boldsymbol{x}, \boldsymbol{n}) = \boldsymbol{\sigma}(\boldsymbol{x})\boldsymbol{n}, \tag{5}$$

where $\boldsymbol{\sigma}$ denotes the symmetric stress tensor. Since the free surface is in mechanical equilibrium, we have, from Eq. (5),

$$\boldsymbol{t}_{\text{fluid}} + \boldsymbol{t}_{\text{gas}} = \boldsymbol{0}. \tag{6}$$

The projection of this relation onto the normal ($\boldsymbol{n}$) with respect to the viscous fluid and tangent ($\boldsymbol{\tau}$) unit vectors of the surface produces the free surface conditions ($\boldsymbol{n}_{gas} = -\boldsymbol{n}$)

$$\boldsymbol{t}_{\text{fluid}} \cdot \boldsymbol{n} = \boldsymbol{t}_{\text{gas}} \cdot \boldsymbol{n} = -p_{\text{gas}}, \tag{7}$$

$$\boldsymbol{t}_{\text{fluid}} \cdot \boldsymbol{\tau} = 0. \tag{8}$$

Here we have discarded the surface tension effects.

1.1.2 Initial Conditions

The initial condition requires that the velocity field satisfies

$$\boldsymbol{v}(\boldsymbol{x}, t = 0) = \boldsymbol{v}^0(\boldsymbol{x}), \tag{9}$$

with

$$\operatorname{div} \boldsymbol{v}^0 = 0, \tag{10}$$

and the initial boundary conditions. The superscript indicates the time level.

1.2 Weak Formulation: the Galerkin Method

The weak formulation of the Navier–Stokes equations is given in the functional context of Sobolev spaces and the spatial approximation is considered in the framework of the Galerkin method. The weak formulation is built up using appropriate test functions. Integration by parts with the help of a Green identity reduces the order of the partial differential operators and enlarges the space of admissible solutions. In the framework of spectral element methods (SEM) the computational domain is broken up into E elements and the space approximation is performed with high-order Lagrange-Legendre polynomials of degree N that are constructed as local tensor-product nodal bases for multidimensional problems. The rate of convergence may be analyzed for smooth

problems and leads to the so-called spectral (exponential) accuracy. Let us notice that the tensor product property of the bases constitutes the basic ingredient for the algorithmic complexity of $O(N^4)$ for the number of operations involved in a matrix vector product in three-dimensional computations.

The weak formulation leads to integrals which are approximated by Gauss-type quadrature rules. In SEM the quadrature rules use the same nodes as the approximation polynomials, therefore inducing a direct correspondence between interpolation and integration. The spectral element discretization for the Navier–Stokes primitive variables $\boldsymbol{v}$ and p is based on the $\mathbb{P}_N - \mathbb{P}_{N-2}$ element which is free of spurious pressure modes. One can show that this element is optimal. The underlying quadrature rules involve the Gauss–Lobatto–Legendre nodes for the velocity field and the Gauss–Legendre grid for the pressure. Figure 1 displays staggered spectral element for the polynomial degree $N = 6$ (left) and $N = 7$ (right).

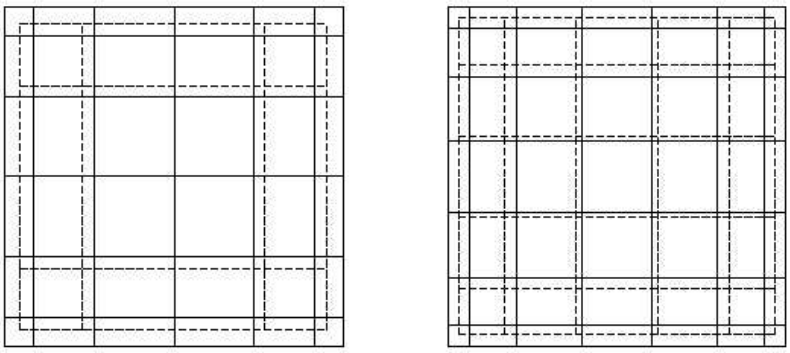


Figure 1: 2D staggered spectral elements for $N = 6$ (left) and $N = 7$ (right).

With the notations borrowed from the monograph [7], the semi-discrete problem reads

$$\boldsymbol{M}\frac{d\underline{\boldsymbol{v}}}{dt} + \boldsymbol{K}\underline{\boldsymbol{v}} - \boldsymbol{D}^T\underline{p} = -Re\,\boldsymbol{C}\underline{\boldsymbol{v}} + \underline{\mathbf{F}}, \tag{11}$$

$$-\boldsymbol{D}\underline{\boldsymbol{v}} = 0 \tag{12}$$

where $\boldsymbol{M}$ is the diagonal mass matrix, $\boldsymbol{K}$ the stiffness matrix, $\boldsymbol{D}^T$ the discrete gradient, $\boldsymbol{C}$ the nonlinear convection operator and $\boldsymbol{D}$ the discrete divergence (transpose of $\boldsymbol{D}^T$). The symbol Re denotes the Reynolds number. The time integration algorithms rest upon an implicit treatment of the transient Stokes operator in order to avoid the stringent stability restriction by the viscous term. This is done by a backward differentiation (Euler) scheme of order two (BDF2). The nonlinear term is treated explicitly by extrapolation in time (EXT2). This method BDF2/EXT2 has no splitting error(see [21]) and yields minimal numerical dissipation and dispersion when applied, for example, to the severe test of the rotating cone problem. However the explicit treatment of the advection term imposes a CFL condition on the time step Δt:

$$u_{\max}\Delta t \leq C/N^2. \tag{13}$$

The solvers for the elliptic symmetric operators are preconditioned conjugate gradient methods by the diagonal matrix extracted from the system matrix. Other preconditioners may be chosen among finite differences, finite elements, the overlapping Schwarz method and more recently, the finite volume method.

The velocity and pressure computations are decoupled as the 3D matrices are too large to be handled by efficient methods. The velocity-pressure matrix is given by

$$\begin{pmatrix} \boldsymbol{H} & -\boldsymbol{D}^T \\ -\boldsymbol{D} & 0 \end{pmatrix} \begin{pmatrix} \underline{\mathbf{v}}^{n+1} \\ \underline{p}^{n+1} \end{pmatrix} = \begin{pmatrix} \boldsymbol{M}\underline{\mathbf{f}}^{n+1} \\ 0 \end{pmatrix}$$

with $\boldsymbol{H}$ a block-diagonal matrix made of d (d is the space dimension) blocks $H = \frac{\beta_k}{\Delta t} M + K$. Solving for the pressure $\underline{p}^{n+1}$ at the new time level, one gets the Uzawa algorithm

$$\mathbf{S}\underline{p}^{n+1} = -\mathbf{D}\mathbf{H}^{-1}\mathbf{M}\underline{\mathbf{f}}^{n+1}$$

$$\mathbf{S} = \mathbf{D}\mathbf{H}^{-1}\mathbf{D}^T$$

. The pressure solvers may also be obtained through generalized block *LU* decompositions and several time splitting schemes are deduced.

As in finite elements, complex geometries are tackled through the use of isoparametric elements. Nonetheless, mortar elements and nodal triangles have become useful tools to deal with complicated geometrical computational domains. In [13], an example of the flow under a blade obstructing severely a plane channel is given demonstrating the capabilities of mortar elements. There the mortars are geometrically conforming but different polynomial degrees are used around the sharp corners of the blade. That procedure allows the user to obtain a smooth pressure field where the geometry might otherwise induce oscillations in the computation.

As an example of application let us consider the two-dimensional vortex dipole colliding on a wall. This problem has been proposed recently as a benchmark problem by Clercx and Bruneau [6]. Numerical results are obtained for $Re = 100, E_x = E_y = 20, N_x = N_y = 10$. The initial conditions are two vortices of same strength and of opposite signs, located at the center of the computational domain with vorticity $\omega = \omega_0(1 - (r/r_0)^2)\exp(-(r/r_0)^2)$, $\omega_0 = 300$ and $r_0 = 0.1$. The results yield very good accuracy as the SEM is able to resolve the small scale structures when the two vortices hit the boundary as can be seen in Fig. 2 at a given instant.

2 Moving-boundary problems and the ALE formulation

2.1 Mathematical model

A moving boundary-fitted grid technique is chosen to simulate the unsteady part of the boundary. In these lecture notes, we focus on cases where the unsteady part of the boundary can be either the free surface in case of free-surface flows, or for fluid-structure interaction problems, the interface between the fluid and the structure. This choice of a surface-tracking technique is primarily based on accuracy requirements. With this group of techniques, the grid is configured to conform to the shape of the interface, and thus adapts continuously—at each time step—to it and therefore provides an accurate description of the moving boundary to express the related kinematic and/or dynamic boundary conditions.

2.1.1 The ALE kinematic framework

This section will introduce the notations used in these lecture notes to define the variables and frames of reference related to the ALE formulation. The notations adopted hereafter are borrowed from [14, 15]. We denote by Ω_t the fluid domain subset of $\mathbb{R}^d$ with $d = 2, 3$ the space dimension, the subscript t referring to the time t as the fluid domain is changing when its boundaries are moving. Let us denote by Ω_0 a reference configuration—for instance the domain configuration at initial time $t = t_0$.

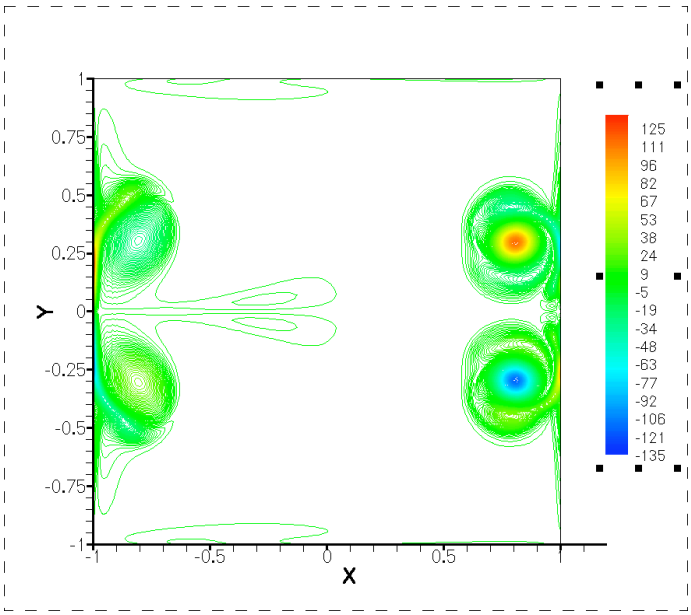


Figure 2: Dipoles of vortices: contours of the z-component of the vorticity field at a given instant.

The system evolution is studied in the time interval $I = [t_0, T]$. The position of a point in the current fluid domain Ω_t is denoted by $\mathbf{x}$—Eulerian coordinate—and in the reference frame Ω_0 by $\mathbf{Y}$—ALE coordinate. Let $\mathcal{A}_t$ be a family of mappings, which at each $t \in I$ associates a point $\mathbf{Y} \in \Omega_0$ to a point $\mathbf{x} \in \Omega_t$:

$$\mathcal{A}_t \ : \Omega_0 \subset \mathbb{R}^d \to \Omega_t \subset \mathbb{R}^d, \qquad \mathbf{x}(\mathbf{Y}, t) = \mathcal{A}_t(\mathbf{Y}). \tag{14}$$

$\mathcal{A}_t$ is assumed to be continuous and invertible on $\overline{\Omega}_0$ and differentiable almost everywhere in I. The inverse of the mapping $\mathcal{A}_t$ is also continuous on $\overline{\Omega}_0$.

The Jacobian matrix of the ALE mapping $\mathcal{A}_t$ is given by

$$\mathbf{J}_{\mathcal{A}_t} = \frac{\partial \mathbf{x}}{\partial \mathbf{Y}}, \tag{15}$$

and its determinant $J_{\mathcal{A}_t} = \det \mathbf{J}_{\mathcal{A}_t}$ is the Jacobian of the mapping characterizing the metrics of Ω_t generated from the one of Ω_0. The Euler expansion formula gives the relationship between the Jacobian of the mapping $\mathcal{A}_t$ and the divergence of the ALE velocity $\mathbf{w}$:

$$\left.\frac{\partial J_{\mathcal{A}_t}}{\partial t}\right|_{\mathbf{Y}} = J_{\mathcal{A}_t} \boldsymbol{\nabla}_{\mathbf{x}} \cdot \mathbf{w}, \qquad \forall (\mathbf{Y}, t) \in \Omega_0 \times I, \tag{16}$$

supplemented by the initial condition $J_{\mathcal{A}_t} = 1$ for $t = t_0$. In real computations, $\mathbf{w}$ will be associated to the mesh velocity. The hyperbolic partial differential equation (16) highlights the important role played by the divergence of the mesh velocity in the time evolution of the mapping $\mathcal{A}_t$. This particular point is emphasized in Section 2.3, where one of the mesh-update techniques used in our simulations, enforces a divergence-free condition for $\mathbf{w}$ resulting in a constant in time Jacobian $J_{\mathcal{A}_t}$ [16]. Furthermore Eq. (16) constitutes the evolution law for $J_{\mathcal{A}_t}$ as in our formulation the ALE mesh velocity is calculated based on the essential boundary conditions of our problem in Ω_t, thereby defining the location of the grid nodes and the value of $\mathcal{A}_t$.

Considering a time-dependent scalar field f defined on $\Omega_t \times I$, the notation $\partial f/\partial t|_{\mathbf{Y}}$ refers to the time derivative in the ALE frame, or in short the ALE time derivative expressed in Eulerian coordinates as opposed to the regular time derivative in Eulerian coordinates and denoted by $\partial f/\partial t|_{\mathbf{x}}$. Finally, the ALE mesh velocity $\mathbf{w}$ is defined as

$$\mathbf{w}(\mathbf{x}, t) = \left.\frac{\partial \mathcal{A}_t}{\partial t}\right|_{\mathbf{Y}}. \tag{17}$$

It is worth noting that a standard application of the chain rule to the time derivative gives

$$\left.\frac{\partial f}{\partial t}\right|_{\mathbf{Y}} = \left.\frac{\partial f}{\partial t}\right|_{\mathbf{x}} + \mathbf{w} \cdot \boldsymbol{\nabla}_{\mathbf{x}} f. \tag{18}$$

The symbol $\boldsymbol{\nabla}_{\mathbf{x}}$ indicates the gradient operation applied to the scalar field f with respect to the Eulerian coordinate $\mathbf{x}$. If $\mathbf{w} = \mathbf{0}$, the mesh is fixed, and we recover the Eulerian description where $\partial/\partial t|_{\mathbf{Y}}$ is the classical time derivative $\partial/\partial t|_{\mathbf{x}}$. If $\mathbf{w} = \mathbf{u}$ where $\mathbf{u}$ is the fluid velocity field, we obtain the Lagrangian description and $\partial/\partial t|_{\mathbf{Y}}$ is the particle derivative. Eq. (18) allows to generalize the Reynolds transport theorem for a time-dependent volume integral of a scalar field

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\int_{\Omega_t} f \,\mathrm{d}\Omega\right) = \int_{\Omega_t} \left(\left.\frac{\partial f}{\partial t}\right|_{\mathbf{Y}} + f \boldsymbol{\nabla}_{\mathbf{x}} \cdot \mathbf{w}\right) \mathrm{d}\Omega. \tag{19}$$

2.1.2 The strong ALE formulation for the Navier–Stokes equations

The governing equations of our moving-boundary problem in the ALE kinematic description, for an incompressible Newtonian fluid flow occupying a fluid domain Ω_t whose boundary $\partial\Omega_t$ is evolving with time, are the Navier–Stokes equations which in strong form and in the Eulerian kinematic description read

$$\left.\frac{\partial \mathbf{u}}{\partial t}\right|_{\mathbf{x}} + \mathbf{u} \cdot \boldsymbol{\nabla}_{\mathbf{x}} \mathbf{u} = -\boldsymbol{\nabla}_{\mathbf{x}} p + 2\nu \boldsymbol{\nabla}_{\mathbf{x}} \cdot \mathbf{D}_{\mathbf{x}}(\mathbf{u}) + \mathbf{f}, \qquad \forall (\mathbf{x}, t) \in \Omega_t \times I, \tag{20}$$

$$\boldsymbol{\nabla}_{\mathbf{x}} \cdot \mathbf{u} = 0, \qquad \forall (\mathbf{x}, t) \in \Omega_t \times I, \tag{21}$$

where $\mathbf{u}$ is the velocity field, p the pressure field (normalized by the constant fluid density ρ and relative to zero ambient), ν the kinematic viscosity of the fluid, $\mathbf{D}_{\mathbf{x}}(\mathbf{u}) = \frac{1}{2}(\boldsymbol{\nabla}_{\mathbf{x}} \mathbf{u} + \boldsymbol{\nabla}_{\mathbf{x}} \mathbf{u}^T)$ the rate-of-deformation tensor and $\mathbf{f}$ the body force per unit mass, with the superscript T indicating the transpose. Eq. (20) expresses the conservation of momentum and the divergence-free condition (21) is the continuity equation in its simplified form for an incompressible flow. Equations (20)–(21) are valid in the internal fluid domain Ω_t, and have to be supplied with boundary conditions on the boundary $\partial\Omega_t$ and the problem being unsteady, an initial condition is also required. The initial velocity field is chosen as

$$\mathbf{u}(\mathbf{x}, t = t_0) = \mathbf{u}^0(\mathbf{x}), \qquad \forall \mathbf{x} \in \Omega_t, \quad \text{with } \Omega_t = \Omega_{t_0} = \Omega_0, \tag{22}$$

such that $\boldsymbol{\nabla}_{\mathbf{x}} \cdot \mathbf{u}^0 = 0$. Let the boundary $\partial\Omega_t$ be split into two non-overlapping parts $\partial\Omega_t = \partial\Omega_t^{\mathcal{D}} \cup \partial\Omega_t^{\boldsymbol{\sigma}}$. In the sequel we will consider the two following types of boundary conditions

$$\mathbf{u} = \mathbf{g}(t), \qquad \text{on } \partial\Omega_t^{\mathcal{D}} \text{ and } \forall t \in I, \tag{23}$$

$$\boldsymbol{\sigma} \cdot \hat{\mathbf{n}} = -p\mathbf{I} \cdot \hat{\mathbf{n}} + 2\nu \mathbf{D}_{\mathbf{x}}(\mathbf{u}) \cdot \hat{\mathbf{n}} = \mathbf{h}(t), \qquad \text{on } \partial\Omega_t^{\boldsymbol{\sigma}} \text{ and } \forall t \in I, \tag{24}$$

where $\boldsymbol{\sigma}$ is the stress tensor, $\mathbf{I}$ the identity tensor and $\hat{\mathbf{n}}$ the local unit outward normal vector to $\partial\Omega_t$. Eq. (23) is an essential boundary condition on $\partial\Omega_t^{\mathcal{D}}$ of Dirichlet type. In the cases of free-surface flows and fluid-structure interaction problems, $\partial\Omega_t^{\mathcal{D}}$ corresponds to fixed or prescribed moving solid walls where a no-slip condition has

to be satisfied. Eq. (24) is a natural boundary condition on $\partial\Omega_t^{\boldsymbol{\sigma}}$. For free-surface flows and fluid-structure interactions, $\partial\Omega_t^{\boldsymbol{\sigma}}$ represents prescribed inflow and/or outflow depending on the situations considered, but primarily the free surface itself or the interface between the fluid and the structure, where a mechanical equilibrium has to be enforced. Therefore Eq. (24) is a dynamic boundary condition (DBC) expressing the continuity of the normal stress at the moving boundary. If free-surface flows are envisaged and if the surface tension is neglected, $\mathbf{h} = -p_0\hat{\mathbf{n}}$ where p_0 is the pressure of the surrounding fluid, taken as zero in the sequel.

Using Eq. (18), we can recast the strong form of the conservation of the momentum of the Navier–Stokes equations in the ALE frame

$$\left.\frac{\partial \mathbf{u}}{\partial t}\right|_{\mathbf{Y}} + (\mathbf{u}-\mathbf{w})\cdot\boldsymbol{\nabla}_{\mathbf{x}}\mathbf{u} = -\boldsymbol{\nabla}_{\mathbf{x}}p + 2\nu\boldsymbol{\nabla}_{\mathbf{x}}\cdot\mathbf{D}_{\mathbf{x}}(\mathbf{u}) + \mathbf{f}, \qquad \forall(\mathbf{x},t)\in\Omega_t\times I, \quad (25)$$

the divergence-free condition (21), the initial and boundary conditions (22)–(24) remaining unchanged in the ALE kinematic description. Indeed, boundary conditions are related to the problem and not to the kinematic description employed, be it Eulerian, Lagrangian or arbitrary Lagrangian-Eulerian. Nevertheless the ALE mesh velocity $\mathbf{w}$ is to a certain extent part of the unknown fields of the problem even though some freedom in moving the mesh makes the ALE technique so attractive. The details related to the treatment and the computation of the mesh velocity are presented in Section 2.3.

2.1.3 The weak ALE formulation for the moving-boundary problem governed by the Navier–Stokes equations

Based on the strong formulation of the moving-boundary problem described in Section 2.1.2, one can derive the more appropriate weak transient ALE formulation. In a standard approach, first are introduced the spaces of test and trial functions used to express the initial problem in its weak form based on the reference configuration Ω_0. We may note that the spaces of test and trial functions for the pressure are identical as no essential Dirichlet condition is being imposed on this field. This space is the space of functions that are square Lebesgue-integrable on the domain Ω_t and is denoted by $L^2(\Omega_t)$. In general the velocity does not necessarily vanish on the domain boundary; in our particular case the existence of a non-homogeneous essential Dirichlet boundary condition on $\partial\Omega_t^{\mathcal{D}}$ leads us to consider different spaces for the test and trial functions for the velocity field. The solution for the velocity $\mathbf{u}$, of the problem (20)–(24) will be searched for directly in the Sobolev space of trial functions $H^1_{\mathcal{D}}(\Omega_t)^d$ defined as follows

$$H^1_{\mathcal{D}}(\Omega_t)^d = \{\mathbf{u}\in L^2(\Omega_t)^d, \quad \boldsymbol{\nabla}_{\mathbf{x}}u_i \in L^2(\Omega_t)^d \text{ with } i=1,\dots,d, \quad \mathbf{u}_{|_{\partial\Omega_t^{\mathcal{D}}}} = \mathbf{g}\}, \quad (26)$$

and corresponding to the current configuration Ω_t. The reference configuration Ω_0 will be used to build the velocity test functions $\hat{\mathbf{v}}$, which will be taken in the space $H^1_{0,\mathcal{D}}(\Omega_0)^d$ with

$$H^1_{0,\mathcal{D}}(\Omega_0) = \{\hat{v}\in L^2(\Omega_0), \quad \boldsymbol{\nabla}_{\mathbf{x}}\hat{v}\in L^2(\Omega_0)^d, \quad \hat{v}_{|_{\partial\Omega_0^{\mathcal{D}}}} = 0\}, \quad (27)$$

to satisfy a homogenous Dirichlet condition on $\partial\Omega_0^{\mathcal{D}}$. Subsequently, the dynamics of the test functions on the configuration Ω_t is obtained by means of the existing inverse of the mapping $\mathcal{A}_t$. Therefore the velocity test functions on the configuration at time t are the set of functions $(\hat{\mathbf{v}}\circ\mathcal{A}_t^{-1})$ with $\hat{\mathbf{v}}$ belonging to $H^1_{0,\mathcal{D}}(\Omega_0)^d$. In the sequel, the notation $(\hat{\mathbf{v}}\circ\mathcal{A}_t^{-1})$ is kept in order to emphasize two key points. First, the treatment of the weak form of the time derivative $\partial\mathbf{u}/\partial t|_{\mathbf{Y}}$ in Eq. (25) is based on the essential property that $\hat{\mathbf{v}}$ is not time dependent and consequently $\partial\hat{\mathbf{v}}/\partial t|_{\mathbf{Y}} = \mathbf{0}$. Second, such formulation highlights the path to follow when practically implementing the weak ALE

formulation. Indeed, the time dependency is fully incorporated in the inverse mapping $\mathcal{A}_t^{-1}$ and the functions $\hat{\mathbf{v}}$ remains the same as the ones used in fixed-grid problems. Moreover, in a more general framework where $\mathcal{A}_t$ is still invertible but less regular, this formulation still holds and one only needs to care for the regularity of the functions $\hat{\mathbf{v}}$ and not of the functions $(\hat{\mathbf{v}} \circ \mathcal{A}_t^{-1})$. With the notations and spaces introduced, the weak transient ALE formulation reads:
Find $(\mathbf{u}(t), p(t)) \in H^1_{\mathcal{D}}(\Omega_t)^d \times L^2(\Omega_t)$ *such that for almost every* $t \geq t_0$

$$\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}t}\int_{\Omega_t} (\hat{\mathbf{v}} \circ \mathcal{A}_t^{-1}) \cdot \mathbf{u}\,\mathrm{d}\Omega + \int_{\Omega_t} (\hat{\mathbf{v}} \circ \mathcal{A}_t^{-1}) \cdot \boldsymbol{\nabla}_{\mathbf{x}} \cdot [\mathbf{u}\mathbf{u} - \mathbf{u}\mathbf{w}]\,\mathrm{d}\Omega = \\
&\qquad \int_{\Omega_t} (p\boldsymbol{\nabla}_{\mathbf{x}} \cdot (\hat{\mathbf{v}} \circ \mathcal{A}_t^{-1}) - 2\nu \mathbf{D}_{\mathbf{x}}(\mathbf{u}) : \boldsymbol{\nabla}_{\mathbf{x}}(\hat{\mathbf{v}} \circ \mathcal{A}_t^{-1}))\,\mathrm{d}\Omega \qquad (28)\\
&\quad + \int_{\Omega_t} \mathbf{f} \cdot (\hat{\mathbf{v}} \circ \mathcal{A}_t^{-1})\,\mathrm{d}\Omega + \int_{\partial\Omega_t^{\boldsymbol{\sigma}}} \mathbf{h} \cdot (\hat{\mathbf{v}} \circ \mathcal{A}_t^{-1})\,\mathrm{d}\partial\Omega, \qquad \forall \hat{\mathbf{v}} \in H^1_{0,\mathcal{D}}(\Omega_0)^d,
\end{aligned}$$

and

$$-\int_{\Omega_t} q\boldsymbol{\nabla}_{\mathbf{x}} \cdot \mathbf{u}\,\mathrm{d}\Omega = 0, \qquad \forall q \in L^2(\Omega_t). \qquad (29)$$

The above set of equations has to be intended in the sense of distribution in the interval $t > t_0$, therefore justifying the qualifier "for almost every $t \geq t_0$", see [17] for greater details. As expected the DBC (24) appears 'naturally' in the weak formulation above, corresponding to the last term on the right-hand side of (28) and being the only 'surface term' as the spatial integration is limited to $\partial\Omega_t^{\boldsymbol{\sigma}}$. In addition, the DBC (24) defines the reference pressure level and therefore no additional condition on the mean value of the pressure is required. Finally, it is well known (see [17] for instance) that a non-homogeneous Dirichlet boundary condition engenders a compatibility condition that the field $\mathbf{u}$ has to satisfy. The origin of this condition is that, in order to be compatible with (21) the boundary condition (23) imposes

$$\int_{\partial\Omega_t} \mathbf{u} \cdot \hat{\mathbf{n}}\,\mathrm{d}\partial\Omega = \int_{\partial\Omega_t^{\mathcal{D}}} \mathbf{g}(t) \cdot \hat{\mathbf{n}}\,\mathrm{d}\partial\Omega + \int_{\partial\Omega_t^{\boldsymbol{\sigma}}} \mathbf{u}(t) \cdot \hat{\mathbf{n}}\,\mathrm{d}\partial\Omega = 0, \qquad \forall t \in I. \qquad (30)$$

Eq. (30) is a consequence of (29) with $q = 1$.

In order to ease the discretization of the continuous weak equations (28)–(29), we introduce the following notations and bilinear forms, such as a scalar product defined by

$$(\mathbf{u}, \hat{\mathbf{v}}) := \int_{\Omega_t} (\hat{\mathbf{v}} \circ \mathcal{A}_t^{-1}) \cdot \mathbf{u}\,\mathrm{d}\Omega, \qquad \forall \hat{\mathbf{v}} \in H^1_{0,\mathcal{D}}(\Omega_0)^d, \qquad (31)$$

a so-called 'energy bilinear form'

$$\mathcal{A}(\mathbf{u}, \hat{\mathbf{v}}) := 2\nu \int_{\Omega_t} \mathbf{D}_{\mathbf{x}}(\mathbf{u}) : \boldsymbol{\nabla}_{\mathbf{x}}(\hat{\mathbf{v}} \circ \mathcal{A}_t^{-1})\,\mathrm{d}\Omega, \qquad \forall \hat{\mathbf{v}} \in H^1_{0,\mathcal{D}}(\Omega_0)^d, \qquad (32)$$

a bilinear form related to the weak incompressibility constraint

$$\mathcal{B}(\hat{\mathbf{v}}, p) := -\int_{\Omega_t} p\boldsymbol{\nabla}_{\mathbf{x}} \cdot (\hat{\mathbf{v}} \circ \mathcal{A}_t^{-1})\,\mathrm{d}\Omega, \qquad \forall \hat{\mathbf{v}} \in H^1_{0,\mathcal{D}}(\Omega_0)^d, \qquad (33)$$

a trilinear form corresponding to the nonlinear convective term

$$\mathcal{C}(\hat{\mathbf{v}}; \mathbf{u}, \mathbf{w}) := \int_{\Omega_t} (\hat{\mathbf{v}} \circ \mathcal{A}_t^{-1}) \cdot \boldsymbol{\nabla}_{\mathbf{x}} \cdot [\mathbf{u}\mathbf{u} - \mathbf{u}\mathbf{w}]\,\mathrm{d}\Omega, \qquad \forall \hat{\mathbf{v}} \in H^1_{0,\mathcal{D}}(\Omega_0)^d, \qquad (34)$$

and finally two linear forms, the first one being related to the source term $\mathbf{f}$

$$\mathcal{F}(\hat{\mathbf{v}}) := \int_{\Omega_t} \mathbf{f} \cdot (\hat{\mathbf{v}} \circ \mathcal{A}_t^{-1}) \, \mathrm{d}\Omega, \qquad \forall \hat{\mathbf{v}} \in H^1_{0,\mathcal{D}}(\Omega_0)^d, \tag{35}$$

and the second one to the non-homogeneous natural boundary condition (24) on $\partial\Omega_t^{\boldsymbol{\sigma}}$

$$\mathcal{H}^{\boldsymbol{\sigma}}(\hat{\mathbf{v}}) := \int_{\partial\Omega_t^{\boldsymbol{\sigma}}} \mathbf{h} \cdot (\hat{\mathbf{v}} \circ \mathcal{A}_t^{-1}) \, \mathrm{d}\partial\Omega, \qquad \forall \hat{\mathbf{v}} \in H^1_{0,\mathcal{D}}(\Omega_0)^d. \tag{36}$$

With these notations, the continuous weak ALE form of our moving-boundary Navier–Stokes problem can be recast as
Find $(\mathbf{u}(t), p(t)) \in H^1_{\mathcal{D}}(\Omega_t)^d \times L^2(\Omega_t)$ *such that for almost every* $t \geq t_0$

$$\frac{\mathrm{d}}{\mathrm{d}t}(\mathbf{u}, \hat{\mathbf{v}}) + \mathcal{A}(\mathbf{u}, \hat{\mathbf{v}}) + \mathcal{B}(\hat{\mathbf{v}}, p) + \mathcal{C}(\hat{\mathbf{v}}; \mathbf{u}, \mathbf{w}) = \mathcal{F}(\hat{\mathbf{v}}) + \mathcal{H}^{\boldsymbol{\sigma}}(\hat{\mathbf{v}}), \quad \forall \hat{\mathbf{v}} \in H^1_{0,\mathcal{D}}(\Omega_0)^d, \tag{37}$$

$$\mathcal{B}(\mathbf{u}, q) = 0, \qquad \forall q \in L^2(\Omega_t). \tag{38}$$

2.2 Numerical technique and discretization

Moving-boundary problems, either free-surface or fluid-structure interaction, represent a real challenge not only for the mathematician but also for the numericists. As presented in Section 2.1, the weak formulation of the problem (37)–(38) is an evidence of its complexity. In this section, particular emphasis will be put on the numerical space discretization of this arduous continuous problem. The general case with non-homogeneous natural and essential Dirichlet boundary conditions is dealt with, justifying the present choice of a very detailed presentation. The particular case of steady problems with non-homogeneous Neumann conditions and homogeneous Dirichlet boundary conditions was first formulated by Ho and Patera in [18] in their study of free-surface flows dominated by inhomogeneous surface-tension effects. Furthermore, Rønquist extended the previous work of Ho and Patera to the more general case of steady free-surface flow problems with non-homogeneous Neumann and Dirichlet boundary conditions [19]. The specificities related to the treatment of transient problems is highlighted in the present lecture notes, which to our knowledge is not available in the current literature.

2.2.1 Spectral element discretization

The first step in the spectral element method (SEM) discretization consists in subdividing the fluid domain $\overline{\Omega}_t = \Omega_t \cup \partial\Omega_t$ into E non-overlapping elements $\{\Omega_t^e\}_{e=1}^E$. In the sequel we will assume that this elemental subdivision is maintained for all values of t in the interval I, therefore meaning that no re-meshing procedure is applied and leading to

$$\Omega_t^e = \mathcal{A}_t(\Omega_0^e), \qquad \text{for } e = 1, \cdots, E, \qquad \forall t \in I. \tag{39}$$

A re-meshing procedure for problem using SEM is presented in [16] and can be used if needed. Each element Ω_t^e involves a mesh constructed as a tensor product of one-dimensional grids. Although each space direction may be discretized independently of the others, without loss of generality we will consider only meshes obtained with the same number of nodes in each direction, denoted by $N+1$, corresponding to the dimension of the space of Nth-order polynomials. To describe the discretization process accurately, we adopt the same procedure as in [7] and define the following spaces

$$X := H^1_{0,\mathcal{D}}(\Omega_0)^d, \qquad Y := H^1_{\mathcal{D}}(\Omega_t)^d, \qquad Z := L^2(\Omega_t). \tag{40}$$

2.2.2 Galerkin approximation

We apply the Galerkin approximation to our Navier–Stokes problem in the ALE formulation in its weak form (37)–(38), and therefore select finite dimensional polynomial subspaces X_N, Y_N and Z_N to represent X, Y and Z respectively. A staggered-grid approach with element based on $\mathbb{P}_N - \mathbb{P}_{N-2}$ spaces for the velocity and pressure field respectively, allows to eliminate completely the spurious pressure modes [20]. In this context, the finite dimensional functional spaces are defined as

$$X_N := X \cap \mathbb{P}^d_{N,E}, \qquad Y_N := Y \cap \mathbb{P}^d_{N,E}, \qquad Z_N := Z \cap \mathbb{P}_{N-2,E}, \tag{41}$$

with

$$\mathbb{P}_{M,E} = \{\phi | \phi \in L^2(\Omega_t); \phi|_{\Omega_t^e} \text{ is a polynomial of degree } \leq M, \forall e = 1, \cdots, E\}, \tag{42}$$

where the superscript d in (41) reflects the fact that test and trial velocity functions are d-dimensional. With these notations the Galerkin approximation of (37)–(38) reads
Find $(\mathbf{u}_N(t), p_N(t)) \in Y_N \times Z_N$ *such that for almost every* $t \geq t_0$

$$\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}t}(\mathbf{u}_N, \hat{\mathbf{v}}_N) + \mathcal{A}(\mathbf{u}_N, \hat{\mathbf{v}}_N) + \mathcal{B}(\hat{\mathbf{v}}_N, p_N) + \\
&\qquad \mathcal{C}(\hat{\mathbf{v}}_N; \mathbf{u}_N, \mathbf{w}_N) = \mathcal{F}_N(\hat{\mathbf{v}}_N) + \mathcal{H}^{\boldsymbol{\sigma}}_N(\hat{\mathbf{v}}_N), \qquad \forall \hat{\mathbf{v}}_N \in X_N, \qquad (43)\\
&\qquad \mathcal{B}(\mathbf{u}_N, q_N) = 0, \qquad \forall q_N \in Z_N, \qquad (44)
\end{aligned}$$

with

$$(\mathbf{u}_N, \hat{\mathbf{v}}_N) = \sum_{e=1}^{E} \int_{\Omega_t^e} \mathbf{u}_N \cdot (\hat{\mathbf{v}}_N \circ \mathcal{A}_t^{-1}) \, \mathrm{d}\Omega, \qquad \forall \hat{\mathbf{v}}_N \in X_N \tag{45}$$

$$\mathcal{F}_N(\hat{\mathbf{v}}_N) = \sum_{e=1}^{E} \int_{\Omega_t^e} \mathbf{f}_N \cdot (\hat{\mathbf{v}}_N \circ \mathcal{A}_t^{-1}) \, \mathrm{d}\Omega, \qquad \forall \hat{\mathbf{v}}_N \in X_N \tag{46}$$

$$\mathcal{H}^{\boldsymbol{\sigma}}_N(\hat{\mathbf{v}}_N) = \sum_{e=1}^{E} \int_{\partial\Omega_t^{e,\boldsymbol{\sigma}}} \mathbf{h}_N \cdot (\hat{\mathbf{v}}_N \circ \mathcal{A}_t^{-1}) \, \mathrm{d}\partial\Omega, \qquad \forall \hat{\mathbf{v}}_N \in X_N \tag{47}$$

$\mathbf{f}_N$ and $\mathbf{w}_N$ being the projection of the fields $\mathbf{f}$ and $\mathbf{w}$ onto the finite dimensional space $\mathbb{P}^d_{N,E}$.

The integrals within each of the spectral elements $\{\Omega_t^e\}_{e=1}^E$ and on the boundaries $\{\partial\Omega_t^{e,\boldsymbol{\sigma}}\}_{e=1}^E$ are performed in a discrete manner using Gaussian quadrature rules. More specifically, all the terms in (43)–(44) are integrated using a Gauss-Lobatto-Legendre (GLL) quadrature rule except for the two terms involving the bilinear form $\mathcal{B}$ where functions discretized in $\mathbb{P}_{N-2,E}$ appear. For these two terms, namely the pressure term and the divergence-free condition, a Gauss-Legendre (GL) quadrature rule is chosen. Therefore, the $\mathbb{P}_N - \mathbb{P}_{N-2}$ Navier–Stokes problem in the ALE form is finally stated as
Find $(\mathbf{u}_N(t), p_N(t)) \in Y_N \times Z_N$ *such that for almost every* $t \geq t_0$

$$\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}t}(\mathbf{u}_N, \hat{\mathbf{v}}_N)_{N,GLL} + \mathcal{A}_{N,GLL}(\mathbf{u}_N, \hat{\mathbf{v}}_N) + \mathcal{B}_{N,GL}(\hat{\mathbf{v}}_N, p_N) + \\
&\mathcal{C}_{N,GLL}(\hat{\mathbf{v}}_N; \mathbf{u}_N, \mathbf{w}_N) = \mathcal{F}_{N,GLL}(\hat{\mathbf{v}}_N) + \mathcal{H}^{\boldsymbol{\sigma}}_{N,GLL}(\hat{\mathbf{v}}_N), \qquad \forall \hat{\mathbf{v}}_N \in X_N, \qquad (48)\\
&\qquad \mathcal{B}_{N,GL}(\mathbf{u}_N, q_N) = 0, \qquad \forall q_N \in Z_N. \qquad (49)
\end{aligned}$$

To simplify the notations in the sequel, we will drop the subscript GLL and unless being explicitly specified, whenever an integration rule is required, the GLL one is implicitly being used.

2.2.3 Semi-discrete Navier–Stokes moving-boundary problem in the ALE form

In order to formulate the semi-discrete version of our moving-boundary problem governed by the Navier–Stokes equations in the ALE form, we introduce the two tensor-product bases on the reference element $\hat{\Omega} := [-1,1]^d$ and for the sake of simplicity we will choose the same discretization order in each space direction N:

- the Gauss-Lobatto-Legendre Lagrangian interpolation basis of degree N

$$\{\pi_{i,j,k}(\boldsymbol{\xi})\}_{i,j,k=0}^{N} := \{\pi_i(\xi_i)\}_{i=0}^{N} \otimes \{\pi_j(\xi_j)\}_{j=0}^{N} \otimes \{\pi_k(\xi_k)\}_{k=0}^{N}, \qquad \forall \boldsymbol{\xi} \in \hat{\Omega}, \tag{50}$$

  to expand the velocity field discretized on the Nth-order GLL grid;

- the Gauss-Legendre Lagrangian interpolation basis of degree $N-2$

$$\{\varpi_{i,j,k}(\boldsymbol{\zeta})\}_{i,j,k=1}^{N-1} := \{\varpi_i(\zeta_i)\}_{i=1}^{N-1} \otimes \{\varpi_j(\zeta_j)\}_{j=1}^{N-1} \otimes \{\varpi_k(\zeta_k)\}_{k=1}^{N-1}, \qquad \forall \boldsymbol{\zeta} \in \hat{\Omega}, \tag{51}$$

  to expand the pressure field discretized on the GL grid of order $N-2$.

The expressions of the one-dimensional GLL and GL Lagrangian interpolant polynomials $\pi(\boldsymbol{\xi})$ and $\varpi(\boldsymbol{\zeta})$ appearing in (50)–(51) can be found in [7]. The polynomials $\{\pi_{i,j,k}(\boldsymbol{\xi})\}_{i,j,k=0}^{N}$ and $\{\varpi_{i,j,k}(\boldsymbol{\zeta})\}_{i,j,k=1}^{N-1}$ will serve as bases for the functions in the spaces X_N, Y_N and Z_N

$$\mathbf{u}_N(\mathbf{x}(\boldsymbol{\xi}),t) = \sum_{i,j,k=0}^{N} \mathbf{u}_{ijk}(t)\ \pi_{i,j,k}(\boldsymbol{\xi}), \qquad \forall (\boldsymbol{\xi},t) \in \hat{\Omega} \times I, \tag{52}$$

$$p_N(\mathbf{x}(\boldsymbol{\zeta}),t) = \sum_{i,j,k=1}^{N-1} p_{ijk}(t)\ \varpi_{i,j,k}(\boldsymbol{\zeta}), \qquad \forall (\boldsymbol{\zeta},t) \in \hat{\Omega} \times I, \tag{53}$$

where $\mathbf{x} = \mathbf{x}^e$ is the location of the point considered in the spectral element Ω_t^e, $\{\mathbf{u}_{ijk}(t)\}_{i,j,k=0}^{N}$ the set of nodal values of the velocity field on the GLL grid of Ω_t^e and $\{p_{ijk}(t)\}_{i,j,k=1}^{N-1}$ the set of nodal values of the pressure field on the GL grid of Ω_t^e. It is important to note that the time-dependency of the discretized velocity $\mathbf{u}_N$ and pressure p_N is not solely accounted by the time-dependent nodal values of these two fields. Indeed, due to the motion of the grid, the mapping between the position in the reference element $\hat{\Omega}$ and the spectral element Ω_t^e at time t is also time-dependent via the ALE mapping $\mathcal{A}_t$. If we note $\mathcal{M}_t^e$ (resp. $\mathcal{M}_0^e$) the mapping from the reference element $\hat{\Omega}$ onto Ω_t^e (resp. Ω_0^e), we can write

$$\mathbf{x}^e = \mathcal{M}_t^e(\boldsymbol{\xi}), \qquad \forall (\boldsymbol{\xi},t) \in \hat{\Omega} \times I, \tag{54}$$

$$\mathbf{Y}^e = \mathcal{M}_0^e(\boldsymbol{\xi}), \qquad \forall (\boldsymbol{\xi},t) \in \hat{\Omega} \times I, \tag{55}$$

$$\mathbf{x}^e = \mathcal{A}_t(\mathbf{Y}^e), \qquad \forall (\mathbf{Y}^e,t) \in \Omega_0^e \times I, \tag{56}$$

leading to following relationship between the different mappings

$$\mathcal{M}_t^e = \mathcal{A}_t \circ \mathcal{M}_0^e, \qquad \forall t \in I. \tag{57}$$

Eq. (57) shows that the second origin of the time-dependency of (52) and (53), after the one due to the set of GLL and GL nodal values, is the moving-grid technique via the time-dependency of the ALE mapping $\mathcal{A}_t$.

Before embarking on the final process of semi-descritizing the equations for the moving-boundary problem, a last issue needs to be addressed: the treatment of the non-homogeneous Dirichlet boundary condition (23) on $\partial\Omega_t^{\mathcal{D}}$. First of all and as mentioned earlier, the non-homogeneity of (23) leads to different spaces for the trial and

test functions for the velocity field, X_N and Y_N respectively. Therefore the basis (52) developed for X_N is not suitable for the solution $\mathbf{u}_N(t)$ of the problem (43)–(44) sought in Y_N. As presented earlier, the non-homogeneous Dirichlet boundary condition imposes to satisfy the compatibility condition (30) whose discrete version reads

$$\sum_{e=1}^{E} \int_{\partial\Omega_t^e} \mathbf{u}_N(t) \cdot \hat{\mathbf{n}} \, \mathrm{d}\partial\Omega = 0, \qquad \forall t \in I. \tag{58}$$

Let $\mathbf{u}_{b,N}$ be a (piecewise) polynomial defined on the discrete boundary $\partial\Omega_t^e$ ($e = 1, \ldots, E$) and such that its nodal boundary values are equal to the boundary data $\mathbf{g}(t)$. In practice, the GLL Lagrangian interpolation bases defined on the element boundaries are used to expand $\mathbf{u}_{b,N}$; however, in the rest of the inner domain these functions are zero. By construction, $\mathbf{u}_{b,N}$ satisfies (58). Setting $\mathbf{u}_N = \mathbf{u}_{0,N} + \mathbf{u}_{b,N}$, the problem reduces to finding $\mathbf{u}_{0,N}$ in the space $Y_{0,N} := H^1_{0,\mathcal{D}}(\Omega_t)^d \cap \mathbb{P}^d_{N,E}$. Therefore the problem (43)–(44) can be reformulated as follows
satisfying (58), and set $\mathbf{u}_{0,N} = \mathbf{u}_N - \mathbf{u}_{b,N}$, then the problem (43)–(44) can be reformulated as follows
Find $(\mathbf{u}_{0,N}(t), p_N(t)) \in Y_{0,N} \times Z_N$ *such that for almost every* $t \geq t_0$

$$\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}t} \left(\mathbf{u}_{0,N}, \hat{\mathbf{v}}_N\right)_N + \mathcal{A}_N(\mathbf{u}_{0,N}, \hat{\mathbf{v}}_N) + \mathcal{B}_{N,GL}(\hat{\mathbf{v}}_N, p_N) + \\
&\mathcal{C}_N(\hat{\mathbf{v}}_N; \mathbf{u}_{0,N}, \mathbf{w}_N) = \mathcal{F}_N(\hat{\mathbf{v}}_N) + \mathcal{H}^{\sigma}_N(\hat{\mathbf{v}}_N) + \mathcal{D}_{1,N}(\hat{\mathbf{v}}_N, t), \qquad \forall \hat{\mathbf{v}}_N \in X_N, \qquad (59)\\
&\mathcal{B}_{N,GL}(\mathbf{u}_{0,N}, q_N) = \mathcal{D}_{2,N,GL}(q_N, t), \qquad \forall q_N \in Z_N, \qquad (60)
\end{aligned}$$

with

$$\begin{aligned}
\mathcal{D}_{1,N}(\hat{\mathbf{v}}_N, t) = &-\frac{\mathrm{d}}{\mathrm{d}t} \left(\mathbf{u}_{b,N}(t), \hat{\mathbf{v}}_N\right)_N \\
&- \mathcal{A}_N(\mathbf{u}_{b,N}(t), \hat{\mathbf{v}}_N) - \mathcal{C}_N(\hat{\mathbf{v}}_N; \mathbf{u}_{b,N}(t), \mathbf{w}_N), \qquad \forall \hat{\mathbf{v}}_N \in X_N, \qquad (61)
\end{aligned}$$

and

$$\mathcal{D}_{2,N,GL}(q_N, t) = -\mathcal{B}_{N,GL}(\mathbf{u}_{b,N}(t), q_N), \qquad \forall q_N \in Z_N. \tag{62}$$

The two time-dependent terms $\mathcal{D}_{1,N}$ et $\mathcal{D}_{2,N,GL}$ appearing in (59) and (60) are due to the non-homogeneity of the Dirichlet boundary condition. These values are related to the values of the discrete field $\mathbf{u}_{b,N}(t)$, which as mentioned earlier, are obtained from the values of the field $\mathbf{g}(t)$ from (23).

We can now expand the trial velocity $\mathbf{u}_{0,N}$ and the trial pressure p_N onto the GLL–GL bases like in (52) and (53) respectively. In the sequel we will drop the subscript 0 in $\mathbf{u}_{0,N}$, no confusion being possible as the non-homogeneous Dirichlet boundary conditions is already accounted for in (59)–(60). The semi-discrete equations derived from (59)–(60) are

$$\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}t}(\mathbf{M}(t)\underline{\mathbf{u}}(t)) = -\mathbf{K}(t)\underline{\mathbf{u}}(t) - \mathbf{C}(\underline{\mathbf{u}}(t), \underline{\mathbf{w}}(t), t)\underline{\mathbf{u}}(t) + \mathbf{D}^T(t)\underline{p}(t) + \underline{\mathbf{F}}_1(t), \qquad (63)\\
&-\mathbf{D}(t)\underline{\mathbf{u}}(t) = \underline{F}_2(t). \qquad (64)
\end{aligned}$$

The matrices appearing in (63)–(64) are all time-dependent: $\mathbf{M}$ is the mass matrix, $\mathbf{K}$ the stiffness matrix, $\mathbf{C}$ the discrete convective operator involving the velocity field $\underline{\mathbf{u}}$ and the ALE mesh velocity $\underline{\mathbf{w}}$, $\mathbf{D}^T$ the discrete gradient operator and $\mathbf{D}$ the discrete divergence. $\underline{\mathbf{F}}_1$ and $\underline{F}_2$ are two vectors accounting for the presence of the body force $\mathbf{f}$ and the time-dependent essential Dirichlet and natural non-homogeneous boundary conditions.

2.2.4 Time discretization

The set of semi-discrete equations (63)–(64) is discretized in time using finite-difference schemes in a decoupled approach. The computation of the linear Helmholtz problem—corresponding to the energy bilinear form $\mathcal{A}$ and the stiffness matrix $\mathbf{K}$—is integrated based on an implicit backward differentiation formula of order 2, the nonlinear convective term—corresponding to the trilinear form $\mathcal{C}$ and the matrix $\mathbf{C}$—is integrated based on a relatively simple extrapolation method of order 2, introduced by Karniadakis et al. [21].

The moving-grid treatment requires the semi-discrete equations (63)–(64) to be supplemented with an equation computing the mesh nodes update

$$\frac{\mathrm{d}\underline{\mathbf{x}}}{\mathrm{d}t} = \underline{\mathbf{w}}, \tag{65}$$

with $\underline{\mathbf{x}}$ being a vector containing the d-dimensional mesh nodes position at time t. The integration of Eq. (65) necessitates the knowledge of the values of the mesh velocity provided by the moving-grid technique employed. Two particular techniques are presented in detail in Section 2.3. The time-integration of Eq. (65) is based on an explicit and conditionally stable Adams–Bashforth of order 3.

Lastly the treatment of the pressure relies on a generalized block LU decomposition with pressure correction [22, 23].

The temporal order of the overall splitting scheme has proved to be equal to two for fixed-grid problems. The grid motion induces a limited reduction of the global temporal order, which is found to fall between 1.5 and 2 for the simulations presented in [3]. The reasons for this reduction in the global order of the method is still not clearly understood.

2.2.5 Specificities pertaining to free-surface flows and fluid-structure interaction

Up to this point, the treatment of our moving-boundary problem was kept to a level general enough to encompass both the free-surface flow and fluid-structure interaction problems. At this stage, it appears natural to provide the specificities pertaining to each of these two sub-problems.

These specificities lie primarily in the natural and essential Dirichlet boundary conditions imposed to the system. For free-surface flows with no surface-tension effects—either of normal or tangential type—and with no inflow nor outflow—closed system, both the natural and essential Dirichlet boundary conditions become homogeneous—$\mathbf{g} = \mathbf{0}$ on $\partial\Omega_t^{\mathcal{D}}$ and $\mathbf{h} = \mathbf{0}$ on $\partial\Omega_t^{\boldsymbol{\sigma}}$, leading to a drastic simplification of the problem. More precisely, both vectors $\underline{\mathbf{F}}_1$ and $\underline{F}_2$ vanish in the semi-discrete formulation (63)–(64) of the problem. For fluid-structure interaction problems, the natural boundary condition on the interface between the flow and the structure is provided by the dynamics of the structure, that can be evaluated by the SEM and the Newmark method [24]. In the sequel, we will focus our attention on the flow problem for both of these two cases of interest.

2.3 Moving-grid techniques

When considering moving-boundary problems tackled in an interface-tracking and ALE frame, the moving boundary $\partial\Omega_t^{\boldsymbol{\sigma}}$ is treated in a Lagrangian way whereas the boundary $\partial\Omega_t^{\mathcal{D}}$ which is fixed or subject to a prescribed motion $\mathbf{g}(t)$, is studied in an Eulerian frame. Accordingly, such method allows large-amplitude motions of the moving boundary, by generating a shape-conformed grid. Hence, an accurate and simple application of the boundary conditions on $\partial\Omega_t$ is easily accessible.

As a consequence of the ALE kinematics, the boundary conditions imposed on the mesh velocity $\mathbf{w}$ read

$$\mathbf{w} \cdot \hat{\mathbf{n}} = \mathbf{u} \cdot \hat{\mathbf{n}}, \qquad \text{on } \partial\Omega_t^{\boldsymbol{\sigma}} \text{ and } \forall t \in I, \tag{66}$$

$$\mathbf{w} = \mathbf{g}(t), \qquad \text{on } \partial\Omega_t^{\mathcal{D}} \text{ and } \forall t \in I. \tag{67}$$

Eq. (66) is a kinematic boundary condition (KBC) on the moving boundary traducing that $\partial\Omega_t^{\boldsymbol{\sigma}}$ is a material surface with no transfer of fluid across it in the Lagrangian perspective. Eq. (67) expresses a kinematic boundary condition of no-slip type on the boundary of the domain which is not free to move. Given (66)–(67), it appears that the freedom left for the choice of $\mathbf{w}$ lies in the values of this field in the internal fluid domain Ω_t and also on the tangential values of $\mathbf{w}$ on the moving boundary $\partial\Omega_t^{\boldsymbol{\sigma}}$.

The computation of the mesh velocity $\mathbf{w}$ in the internal fluid domain Ω_t is the corner-stone of the moving-grid technique developed in the framework of the ALE formulation. The values of the mesh velocity being prescribed on the boundary $\partial\Omega_t$ as expressed by equations (66)–(67), the evaluation of $\mathbf{w}$ in Ω_t can be obtained as the solution of an elliptic equation:

$$\mathcal{E}_{\mathbf{x}}\mathbf{w} = \mathbf{0}, \qquad \forall(\mathbf{x}, t) \in \Omega_t \times I, \tag{68}$$

where $\mathcal{E}_{\mathbf{x}}$ represents any elliptic operator based on the Eulerian coordinates $\mathbf{x}$. Such elliptic equation constitutes a classical choice for calculating the mesh velocity [25]. Two types of elliptic equations are envisaged. The first elliptic operator used is a Laplacian operator $\boldsymbol{\Delta}_{\mathbf{x}}$, and will be used extensively in the fluid-structure interaction simulations presented in [3]. More details about the use of this specific operator for the computation of the mesh velocity can be found in [24]. The second strategy relies on the assumption that the motion of the mesh nodes is equivalent to a steady Stokes flow, corresponding physically to an incompressible and elastic motion of the mesh. Therefore the problem for the mesh velocity becomes a boundary-value steady Stokes problem with the mesh velocity $\mathbf{w}$ satisfying a divergence-free condition

$$\boldsymbol{\nabla}_{\mathbf{x}} \cdot \mathbf{w} = 0 \qquad \forall(\mathbf{x}, t) \in \Omega_t \times I. \tag{69}$$

The justifications of this additional constraint imposed to the mesh velocity problem is presented in detail in [16]. The free-surface flow simulations of a sloshing in a three-dimensional tank due to the gravity presented in [3], were carried out using this second strategy for $\mathbf{w}$.

2.4 Applications to free-surface cylindrical swirling flow

Unsteady incompressible viscous flows of a fluid partly enclosed in a cylindrical container, of height H, with an open top surface have been computed using the present moving-grid framework. These moving free-surface flows are generated by the steady rotation of the solid bottom end-wall. Such types of flows belong to a group of recirculating lid-driven cavity flows with geometrical axisymmetry. The top surface of the cylindrical cavity is left open so that the free surface can be freely deformed. The Reynolds regime corresponds to unsteady transitional flows with some incursions in the fully laminar regime. The approach taken here revealed new nonaxisymmetric flow states that are investigated based on a fully three-dimensional solution of the Navier–Stokes equations for the free-surface cylindrical swirling flow, without resorting to any symmetry property unlike all other results available in the literature. The results are compared with those of Bouffanais & Lo Jacono [26] corresponding to the exact same parameters but with a flat-and-fixed top free surface obtained using SEM. The complete set of results are reported in [27].

Figure 3 (left) displays the relative surface elevation η/H in any half meridional plane for both steady cases (a): Re = 900, and (b): Re = 1 500. Three-dimensional

views of these two cases are available in Fig. 4. One clearly distinguishes two regions corresponding to surface elevations of opposite signs. In the inner-core region of the flow, say $r/R \leq 0.4$, the algebraic surface elevation is the largest in absolute value and a large trough forms. This internal trough is surrounded by an outer region of strictly positive surface elevation all the way to the circular contact line along the tubular side-wall. The radial position of this ring crest is located around $r/R = 0.9$ and its amplitude is much smaller than the one of the trough measured on the cylinder axis. The outer ring of positive η is generated by the impingement of the tubular wall-jet, consequence of the pumping effect of the Ekman layer above the rotating disk. The inner trough is itself another consequence of this Ekman layer, but through its sucking effect which completes the large meridional circulation generated by the motion of the disk. It also corresponds to the region of the flow in a state of quasi-solid-body rotation. The sharp drop to $\eta/H = 0$ observed when approaching the tubular side wall is justified by the absence of dynamics of the contact line between the three phases: fluid, gas and solid. Indeed, surface tension is neglected and a no-slip condition is imposed to the fluid on the tubular side wall, while the top surface is free to move.

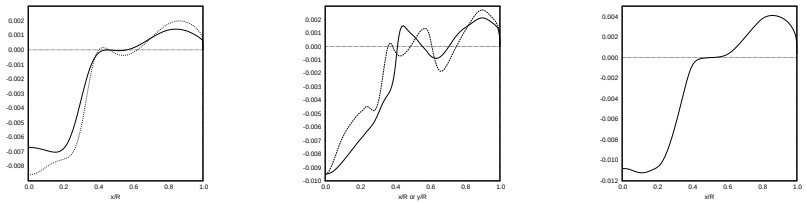


Figure 3: Relative free-surface elevation η/H measured from the position $z = H$ at rest, in a meridional plane. Left: solid line: Re = 900; Dotted line: Re = 1 500. Center: Re = 6 000 (instantaneous flow sample); Solid line: meridional plane $y/R = 0$; Dashed line: meridional plane $x/R = 0$. Right: Re = 6 000 interpolated average.

When increasing the Reynolds number up to Re = 6 000, the numerical simulation yields a non-axisymmetric free-surface shape as can be seen in two orthogonal meridional planes in Fig. 3 (center) and in a three-dimensional view in Fig. 4. The interpolated average of the relative free-surface elevation is shown in Fig. 3 (right). The comments associated to the comparison of cases (a) and (b) remain valid for this higher-Reynolds-number case. One may however note in Fig. 3 the more complex radial variations of η in the larger transition region between the inner trough an the outer crest ring. In the flat-free-surface case in Bouffanais & Lo Jacono [26], the unstable modes were found to be active in this region comprising the interfacial zone. The 3D view in the bottom row of Fig. 4 reveals a free-surface shape of "bathtub sink-vortex" type. Compared to the laminar cases (a), (b), the free-surface elevation for the unsteady transitional case at Re = 6 000 displays extremely rapid variations in space and hence a less "soft" aspect. In practice, the free surface is never perfectly clean of surfactants, which would inevitably act onto these abrupt spatial variations and lead to a "softer" free surface shape.

2.5 Fluid-structure interaction

N. Bodard [1] elaborates a SEM approach for a viscous fluid flowing over an obstacle modeled as an elastic solid. The dynamics equations for the solid are the Navier equations written in weak form. The displacement vector which is the basic variable is approximated in the $\mathbb{P}_N$ space while the time integrator is the Newmark method. Fig. 5 shows the sketch of the geometry when a plane Poiseuille flow hits a slab-type obstacle that moves around its center point with a constant angular rotation speed

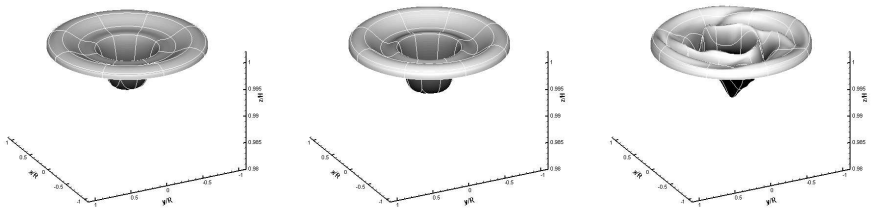


Figure 4: Three-dimensional views of the free-surface shape η/H measured from the position $z = H$ at rest. Left: Re = 900; Center: Re = 1 500; Right: Re = 6 000 (instantaneous flow sample). The spectral element grid appears in solid white line to enhance the 3D visualization.

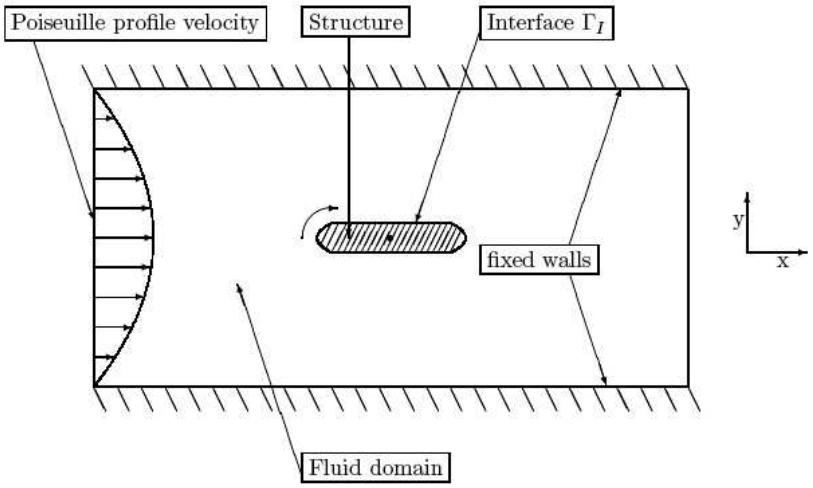


Figure 5: Fluid-structure interaction

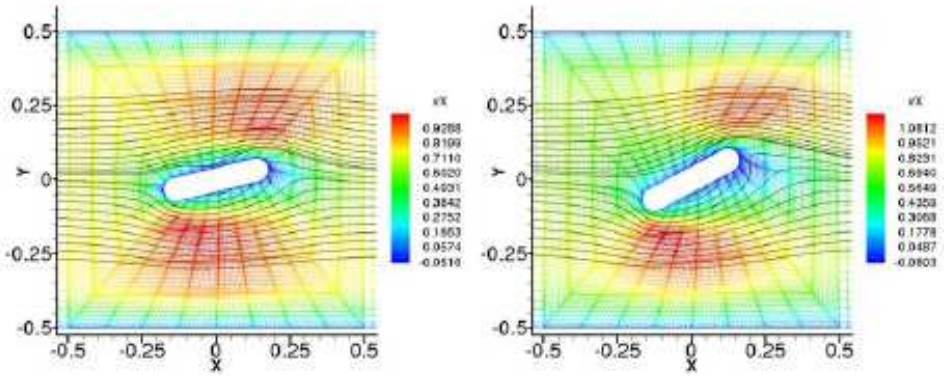
$\omega = 1$ in counterclockwise direction. The height of the channel is 1 and the length of the computational domain is 6. The Reynolds number is $Re = 100$.

The computation was carried out with $E = 112$ elements of degree $N = 12$. Fig. 6 show six successive positions at $t = 0.25, 0.5, 1.0, 1.25, 1.5, 1.75$. We observe recirculation zones downstream the valve after $t = 1.0$. Even though the mesh is quite deformed at the end of the simulation, this does not affect the quality of the results.

3 Large-eddy simulations using SEM

3.1 Introduction

Large-eddy simulation (LES) represents a way of reducing the number of degrees of freedom of the simulation with respect to the requirements of the direct numerical simulation (DNS). This is done by calculating only low-frequency modes in space and modeling high-frequency ones, the scale separation being performed by filtering in space the Navier–Stokes equations. Large-scale structures are obtained by the computed flow dynamics while the behavior of subgrid scales and their interaction



$t = 0.25$ $t = 0.5$

with large eddies are modeled by the additional term in the LES governing equations resulting from filtering the Navier–Stokes equations. The expression of the additional term as a function of the resolved field is referred to as subgrid modeling.

Large-eddy simulations of the turbulent flow in a lid-driven cubical cavity have been carried out at a Reynolds number of 12 000 using the Legendre spectral element method. Distinct subgrid-scales models have been considered: a dynamic Smagorinsky model (DSM), a dynamic mixed model (DMM), and also approximate deconvolution models (ADM) based on the van Cittert method. Approximate deconvolution models (ADM) constitute a particular family of subgrid models. They rely on the attempt to recover, at least partially, the original unfiltered fields, up to the grid level, by inverting the filtering operator applied to the Navier–Stokes equations. The focus here is on the approximate iterative method introduced by Stolz & Adams [28] which is based on the van Cittert procedure.

These subgrid models have been implemented and used to perform long-lasting simulations required by the relevant time scales of the flow. All filtering levels make use of explicit filters applied in the physical space (on an element-by-element approach) and spectral (modal) spaces. The subgrid-scales models have been validated and compared to available experimental and numerical reference results, showing very good agreement. Specific features of lid-driven cavity flows in the turbulent regime, such as inhomogeneity of turbulence, turbulence production near the downstream corner eddy, small-scale localization and helical properties have been investigated and discussed in the large-eddy simulation framework. Time histories of quantities such as the total energy, total turbulent kinetic energy or helicity exhibit different evolutions but only after a relatively long transient period. However, the average values remain extremely close.

The implementation of the ADM-based model is a novel variant of existing approximate deconvolution models blended with a mixed scale model using a dynamic evaluation of the subgrid-viscosity constant is proposed.

Subgrid modeling in the case of a flow with coexisting laminar, transitional and turbulent zones such as the lid-driven cubical cavity flow represents a challenging problem. Moreover, the coupling with the spectral element method having very low numerical dissipation and dispersion builds a well suited framework to analyze the efficiency of a subgrid model. First- and second-order statistics obtained using this new model are showing very good agreement with the direct numerical simulation.

The complete set of results are reported in three journal articles [4, 5, 31].

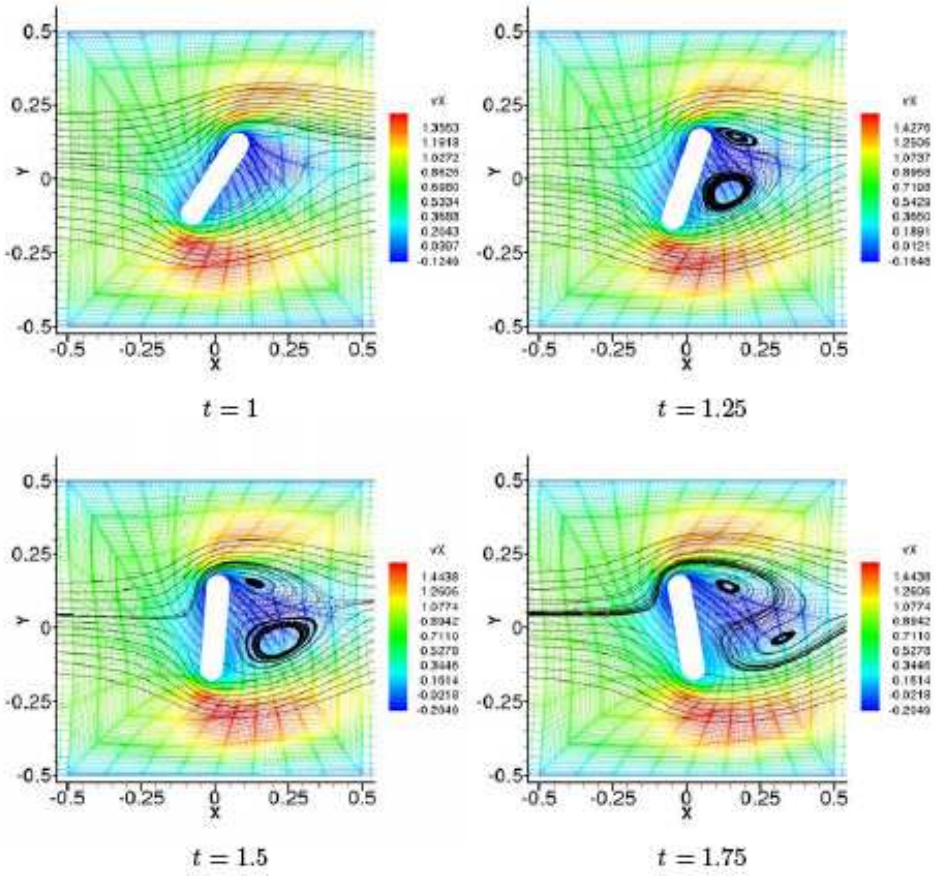


Figure 6: Horizontal velocity component and streamlines (black)

3.2 Governing equations and numerical method

3.2.1 Governing equations

In the case of isothermal flows of Newtonian incompressible fluids, the LES governing equations for the filtered quantities denoted by an *overbar*, obtained by applying a convolution filter $\mathcal{G}\star$ to the Navier–Stokes equations, read

$$\frac{\partial \overline{u}_i}{\partial t} + \frac{\partial}{\partial x_j}(\overline{u}_i \overline{u}_j) = -\frac{\partial \overline{p}}{\partial x_i} + \nu \frac{\partial}{\partial x_j}\left(\frac{\partial \overline{u}_i}{\partial x_j} + \frac{\partial \overline{u}_j}{\partial x_i}\right) - \frac{\partial \tau_{ij}}{\partial x_j}, \tag{70}$$

$$\frac{\partial \overline{u}_j}{\partial x_j} = 0, \tag{71}$$

the filtered velocity field $\overline{\mathbf{u}} = \mathcal{G}\star\mathbf{u}$ satisfying the divergence-free condition (71) through the filtered reduced pressure field $\overline{p}$. The components of the subgrid tensor $\boldsymbol{\tau}$ are given by

$$\tau_{ij} = \overline{u_i u_j} - \overline{u}_i \overline{u}_j, \tag{72}$$

and ν is the kinematic viscosity. The closure of the filtered momentum equation (70) requires $\boldsymbol{\tau}$ to be expressed in terms of the filtered field which reflects the subgrid scales modeling and the interaction among all space scales of the solution.

3.2.2 Space discretization

The numerical method and the space discretization are the same as the ones introduced in Section 1.2. For the sake of clarity, the central lines and ideas are briefly recalled here. The numerical method treats Eqs. (70)–(71) within the weak Galerkin formulation framework. The SEM consists of dividing the computational domain into a given number of spectral elements. In each spectral element, the velocity and pressure fields are approximated using Lagrange–Legendre polynomial interpolants. The velocity and pressure are expressed in the $\mathbb{P}_N - \mathbb{P}_{N-2}$ functional spaces where $\mathbb{P}_N$ is the set of polynomials of degree lower than N in each space direction. This spectral element method avoids the presence of spurious pressure modes as it was proved by Maday & Patera [20]. The quadrature rules are based on a Gauss–Lobatto–Legendre (GLL) grid for the velocity nodes and a Gauss–Legendre grid (GL) for the pressure nodes.

Borrowing the notation from [7], the semi-discrete filtered Navier–Stokes equations resulting from space discretization are

$$\mathbf{M}\frac{\mathrm{d}\underline{\overline{\mathbf{u}}}}{\mathrm{d}t} + \mathbf{C}\underline{\overline{\mathbf{u}}} + \nu\mathbf{A}\underline{\overline{\mathbf{u}}} - \mathbf{D}^T\underline{\overline{p}} + \mathbf{D}_{\boldsymbol{\tau}}\underline{\boldsymbol{\tau}} = 0, \tag{73}$$

$$-\mathbf{D}\underline{\overline{\mathbf{u}}} = 0. \tag{74}$$

The diagonal mass matrix $\mathbf{M}$ is composed of three blocks, namely the mass matrices M. The global vector $\underline{\overline{\mathbf{u}}}$ contains all the nodal velocity components while $\underline{\overline{p}}$ is made of all nodal pressures. The matrices $\mathbf{A}$, $\mathbf{D}^T$, $\mathbf{D}$, $\mathbf{D}_{\boldsymbol{\tau}}$ are the discrete Laplacian, gradient, divergence and tensor divergence operators, respectively. The matrix operator $\mathbf{C}$ represents the action of the nonlinear term written in convective form $\underline{\overline{\mathbf{u}}} \cdot \boldsymbol{\nabla}$, on the velocity field and depends on $\underline{\overline{\mathbf{u}}}$ itself. The semi-discrete equations constitute a set of nonlinear ordinary differential equations (73) subject to the weak incompressibility condition (74).

3.2.3 Time integration

Standard time integrators in the SEM framework handle the viscous linear term and the pressure implicitly by a backward differentiation formula of order 2 (BDF2) to avoid stability restrictions such that $\nu\Delta t \leq C/N^4$, while all nonlinearities, including the discretized subgrid term $-\mathbf{D}\underline{\boldsymbol{\tau}}$, are computed explicitly, e.g. by a second order extrapolation method (EXT2), under the CFL restriction. The implicit part is solved by a generalized block LU decomposition with a pressure correction algorithm [7, 22]. The overall order-in-time of the afore-presented numerical method is two.

3.3 *A posteriori* validations

In this section, results of the LES are compared with the available reference results by analyzing first- and second-order statistics. The measurements reported by Leriche and Gavrilakis [29] were taken in the mid-plane of the square cavity $z/h = 0$, which is the statistical symmetry plane of the flow domain. For the sets of DNS data, the total velocity field is considered whereas in the case of LES, only its resolved part is presented. In consequence, the statistical moments computed from the resolved field cannot be equal to those computed from the DNS. One solution to overcome this issue would have been to apply the same filtering as is used for the scale separation to the reference solution. The statistics for all LES and under-resolved DNS (UDNS) are based on a sampling approximately 10 times smaller than the one of the original DNS, but about twice longer than the one of the LES of Zang et al. [30]; more precisely 400 samples are collected over 80 dimensionless time units. The original reference DNS statistics were performed using 4'000 samples extracted over an integration range of

1'000 time units. Therefore, the LES statistics are not expected to be identical to the reference ones, especially the second-order ones.

The comparisons with the DNS results are performed by plotting identical series of contour levels of the average velocity. Figures 7 displays the average values of the velocity field for the dynamic mixed scale (DMS) model without ADM, and with ADM (ADM-DMS), and the DNS in the mid-plane of the cavity. A rapid overview of these figures indicates that ADM-DMS provides results very close to the DNS references, which has to be compared with the UDNS results [31]. In addition, it appears that ADM-DMS results are more satisfactory than those from DMS. Secondary corner eddies located above the bottom wall and below the lid next to the upstream wall are correctly resolved in the mean flow. The flow below the lid and near the corner with the downstream wall presents wiggles in the LES contours for $\langle \overline{u}_y \rangle$. More limited effects are noticeable for the equivalent x-component field. We assume that these very limited defects find their origin in a local too important under-resolution due to the very high shear rate near the downstream corner right below the lid [5]. The previous comparisons of ADM-DMS with the DNS and DMS for first-order moments require to be complemented by plotting identical series of contours of three components of the resolved Reynolds stress tensor. Figure 8 showcases the improvement achieved in terms of subgrid modeling by coupling ADM with DMS. Indeed, the variations of the root-mean-square (rms) quantities $\langle \overline{u}_x^{\circ 2} \rangle^{1/2}$, $\langle \overline{u}_y^{\circ 2} \rangle^{1/2}$ and $\langle \overline{u}_x^{\circ} \overline{u}_y^{\circ} \rangle$ for ADM-DMS reproduce quite accurately the intense-fluctuations zones in the mid-plane $z/h = 0$, and more specifically in the vicinity of the downstream corner eddy. DMS appears clearly not as effective as ADM-DMS. The lower intensity of the Reynolds stress components for ADM-DMS as compared to the DNS are induced by the lower sampling of all LES. A longer dynamic range would produce more intense results as reported in [5].

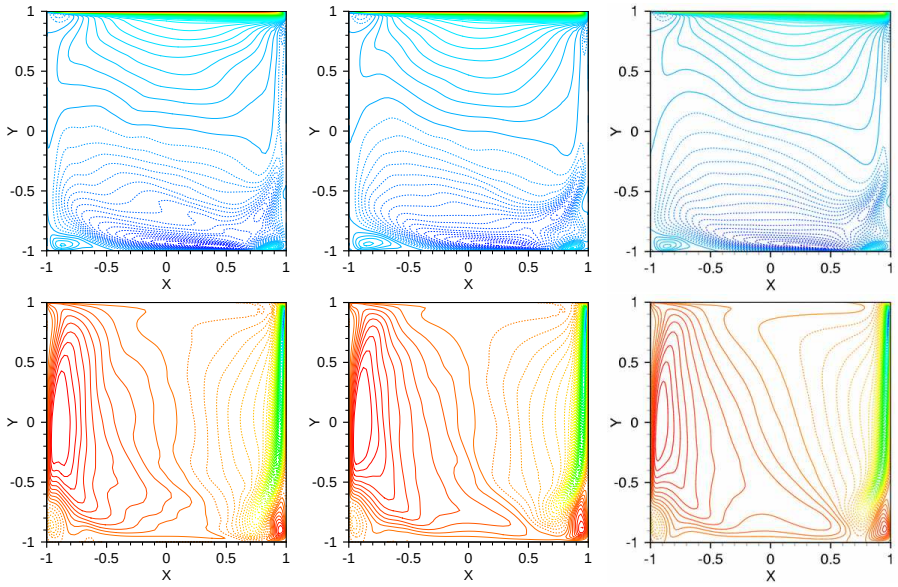


Figure 7: In the mid-plane $z/h = 0$, DMS (left column), ADM-DMS (central column) and DNS (right column). Top row: contours of $\langle \overline{u}_x \rangle$ from -0.2 to 1 by increments of 0.01. Bottom row: contours of $\langle \overline{u}_y \rangle$ from -0.7 to 0.1 by increments of 0.01. Color scale from blue to red. Dashed contours correspond to negative levels. Levels in U_0 units.

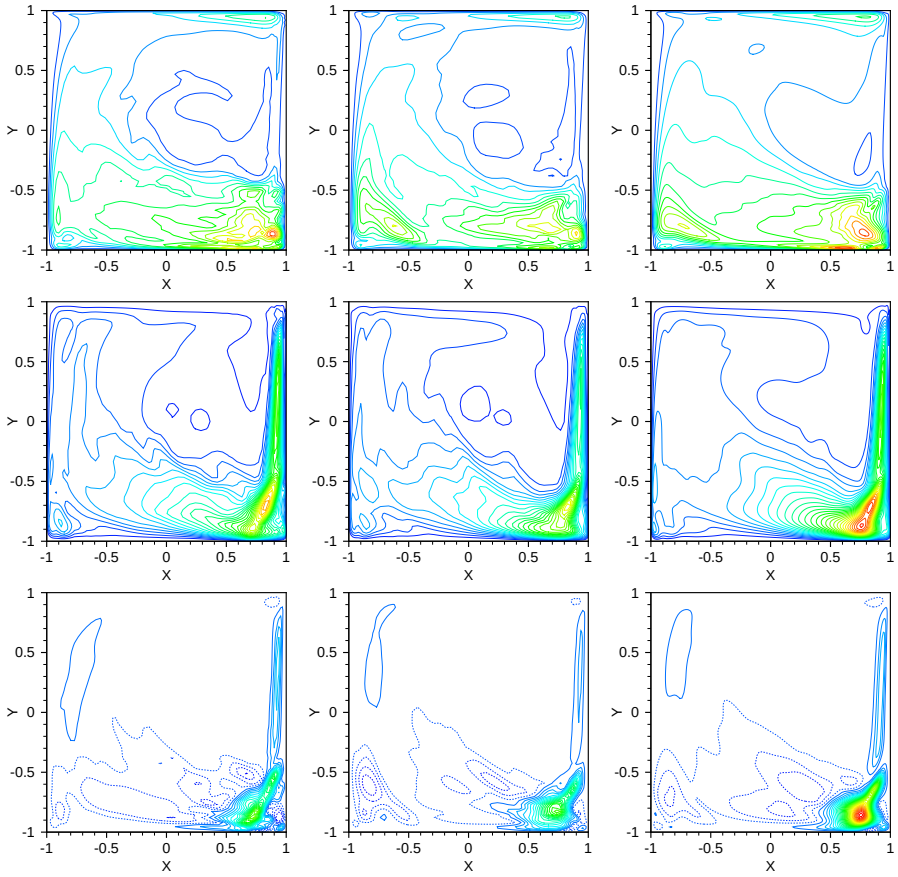


Figure 8: In the mid-plane $z/h = 0$, DMS (left column), ADM-DMS (central column) and DNS (right column). Top row: contours of $\langle \overline{u}_x^{\circ 2} \rangle^{1/2}$ from 0 to 0.07 by increments of 0.005. Central row: contours of $\langle \overline{u}_y^{\circ 2} \rangle^{1/2}$ from 0 to 0.15 by increments of 0.005. Bottom row: contours of $\langle \overline{u}_x^{\circ} \overline{u}_y^{\circ} \rangle$ from -0.0007 to 0.0065 by increments of 0.0002. Color scale from blue to red. Dashed contours correspond to negative levels. Levels in U_0 units for $\langle \overline{u}_x^{\circ 2} \rangle^{1/2}$ and $\langle \overline{u}_y^{\circ 2} \rangle^{1/2}$ and in U_0^2 units for $\langle \overline{u}_x^{\circ} \overline{u}_y^{\circ} \rangle$.

3.4 Flows of viscoelastic fluids

3.4.1 Introduction

Non-Newtonian fluids present many effects that can not be predicted by classical viscous fluid mechanics. Among them, viscoelastic fluids are characterized by specific features like the shear-rate dependence of the shear viscosity, the presence of normal stresses in viscometric flows, the high resistance to elongational deformations and the memory effects related to their elastic properties.

The mathematical model of viscoelastic flows consists of the set of three coupled equations which are to be solved simultaneously, namely the continuity and momentum equations (which are related to the conservation of mass and momentum) in addition to the constitutive equation (in which viscous and elastic characteristics of the fluid appear). The last equation has a tremendous impact on the results of numerical simulations and on the stability of the method.

It is possible for the time integration scheme to use well-proven splitting techniques that have been applied to the transient Navier-Stokes equations. Some specific developments are introduced to handle the additional terms and equations brought by the non-Newtonian model.

3.4.2 Governing equations

The continuity equation for incompressible fluids enforces a divergence-free velocity field as in the Newtonian case :

$$\nabla \cdot \mathbf{v} = 0 \tag{75}$$

The momentum equation is modified with respect to the Newtonian case in order to account for the additional viscoelastic contribution due to the extra-stress tensor.

$$\frac{\partial \mathbf{v}}{\partial t} = -Re(\mathbf{v} \cdot \nabla)\mathbf{v} - \nabla p + R_\mu \nabla^2 \mathbf{v} + \nabla \cdot \boldsymbol{\tau} \tag{76}$$

with:

$$\boldsymbol{\tau} = \boldsymbol{\sigma} + p\,\mathbf{I} - 2\,R_\mu \mathbf{D} \tag{77}$$

The symbols $\boldsymbol{\sigma}$, $\mathbf{I}$ and $\mathbf{D}$ are respectively the Cauchy stress, identity and rate-of-deformation tensors. The latter is defined by:

$$\mathbf{D} = \frac{1}{2}(\nabla \mathbf{v} + (\nabla \mathbf{v})^T) \tag{78}$$

The pressure scalar field enforces the continuity constraint imposed by Eq. (75). The extra-stress tensor is split into two parts, namely a Newtonian one $2\,R_\mu \mathbf{D}$ and a viscoelastic one $\boldsymbol{\tau}$. The parameters R_μ and Re are the ratio of (Newtonian) solvent to total viscosities ($\frac{\mu_N}{\mu_t}$) and the Reynolds number ($\frac{\rho V L_r}{\mu_t}$). In Eq. (78), $(\nabla \mathbf{v})^T$ is the transpose of $\nabla \mathbf{v}$, the velocity gradient.

Let us consider here particular models of the FENE-dumbbell type. It is then customary to introduce a non-dimensional configuration tensor $\mathbf{A}$, which is related to the viscoelastic contribution $\boldsymbol{\tau}$ of the extra-stress tensor. As opposed to other models, the constitutive equation is separated into two parts: an evolution equation for the dumbbells and an expression for the polymer stress in order to produce a clear representation of the polymer microstructure and of its dynamic influence. The FENE-CR model has been introduced to obtain an equation as simple as possible while including the constant shear viscosity characteristics of a Boger fluid and allowing only finite extensibility of the dumbbells. The FENE-P model is a close version, which enables to take into account shear-thinning effects present in many viscoelastic fluids.

For both FENE models, the relation between the viscoelastic stress and configuration tensors is given by:

$$\boldsymbol{\tau} = \frac{1 - R_\mu}{We} \frac{1}{1 - \frac{tr(\mathbf{A})}{L_e^2}} \left(\mathbf{A} - f(tr(\mathbf{A}))\,\mathbf{I}\right) \tag{79}$$

The configuration tensor $\mathbf{A}$ satisfies the following differential equation:

$$(1 - \frac{tr(\mathbf{A})}{L_e^2})(E\frac{\partial \mathbf{A}}{\partial t} + We((\mathbf{v} \cdot \nabla)\mathbf{A} - \mathbf{A} \cdot (\nabla \mathbf{v})^T - \nabla \mathbf{v} \cdot \mathbf{A})) + \mathbf{A} = f(tr(\mathbf{A}))\,\mathbf{I} \tag{80}$$

with:

$$f(tr(\mathbf{A})) = \frac{1 - \frac{tr(\mathbf{A})}{L_e^2}}{K} \tag{81}$$

for the FENE-P model or

$$f(tr(\mathbf{A})) = 1 \tag{82}$$

for the FENE-CR model.

In Eq. (79), $1 - R_\mu$ is the non-dimensional viscosity ($1 - R_\mu = \frac{\mu_P}{\mu_t}$ with $\mu_P = \mu_t - \mu_N$) of the polymeric component of the fluid. The symbol tr denotes the trace.

In Eq. (81), K is usually taken as: $K = 1 - \frac{3}{L_e^2}$. The parameter L_e measures the extensibility of the dumbbells. In some papers, K is equal to 1. When $L_e \to \infty$, the FENE-P and FENE-CR models are equivalent to the Oldroyd-B model. The upper-convected Maxwell (UCM) model corresponds to an Oldroyd-B model where $R_\mu = 0$.

The parameters $We = \lambda \frac{V}{L_r}$ and $E = \frac{We}{Re} = \frac{\lambda \mu_t}{\rho L_r^2}$ are respectively the Weissenberg and elastic number based on the relaxation time λ, density ρ, total dynamic viscosity μ_t, reference velocity V and length L_r. They are indicators of elastic effects in the flow.

It must be emphasized that the presence of E in Eq. (80) is due to the particular scaling used for time ($\frac{\rho L_r^2}{\mu_t}$) related to viscous effects as indicated in the introductory part of this section. As Re is decreased and goes to 0, the factor in front of the partial derivative with respect to time becomes infinite. In order for the whole term $E \frac{\partial \mathbf{A}}{\partial t}$ to remain finite, the partial derivative must vanish, which means that the configuration and viscoelastic stresses remain constant with respect to time so that no memory-induced effects (typical of viscoelastic fluids) are present and the fluid tends to behave as Newtonian. If the inertial time $\frac{L_r}{V}$ would have been used for non-dimensionalization, E should be replaced by We in the factor of the time partial derivative. Both types of scaling have been used in the literature depending on the kind of flows (inertia or viscosity-dominated) to be investigated.

3.4.3 Spectral element formulation

In this section, the spectral element method based on the Galerkin formulation is presented. Since a high-order method which is not numerically dissipative and dispersive is needed in order to avoid erroneous simulation of the physics involved in the stress boundary layers, a spectral element method using Legendre polynomials is considered as a good candidate for discretizing the non-linear set of partial differential equations that model the flow.

Basic formulation

The problem consisting of solving the continuity, momentum and constitutive equations (75)-(80) on a flow domain $\Omega \subset R^d$, where d is the problem dimension, in order to determine the three variables (pressure, velocity and viscoelastic stress) can be expressed with a general weak (Galerkin) formulation, which is also used in the finite element context.

To this end, it is necessary to define first a set of functional spaces X_p, X_v and X_τ to which the functions representing respectively the pressure p, velocity $\mathbf{v}$ and viscoelastic stress $\boldsymbol{\tau}$ will belong. These spaces are respectively subspaces of $L^2(\Omega)$, $H^1(\Omega)^d$ and $L^2(\Omega)^{d \times d}$, where $L^2(\Omega)$ and $H^1(\Omega)$ represent the space of measurable functions

that are square-integrable on Ω and the space of derivable functions of which first order partial derivatives are measurable and square-integrable on Ω. Their definitions include the specification of boundary conditions.

Second, multilinear forms A, B and C involving the three variables can be defined based on integral inner products with test functions belonging to the functional spaces. The initial problem is then expressed in the weak formulation as:

Find $(p,\mathbf{v},\boldsymbol{\tau})\in X_p \times X_v \times X_\tau$ such that:

$$A(\mathbf{v},\Psi_p) = \int_\Omega (\nabla\cdot\mathbf{v})\Psi_p \, d\Omega = 0, \quad \forall\, \Psi_p \in X_p \subset L^2(\Omega) \tag{83}$$

$$\begin{aligned} B(p,\mathbf{v},\boldsymbol{\tau},\boldsymbol{\Psi}_v) = \quad & \int_\Omega \tfrac{\partial \mathbf{v}}{\partial t}\cdot\boldsymbol{\Psi}_v \, d\Omega + \int_\Omega -p(\mathbf{I}:\nabla\boldsymbol{\Psi}_v)\, d\Omega \\ & + R_\mu \int_\Omega (\nabla\mathbf{v}):\nabla\boldsymbol{\Psi}_v \, d\Omega + Re\int_\Omega[(\mathbf{v}\cdot\nabla)\mathbf{v}]\cdot\boldsymbol{\Psi}_v \, d\Omega \\ & + \int_\Omega \boldsymbol{\tau}\cdot\nabla\boldsymbol{\Psi}_v \, d\Omega - \int_{\partial\Omega}(\boldsymbol{\sigma}\cdot\mathbf{n})\cdot\boldsymbol{\Psi}_v \, d\Gamma \\ & = 0, \quad \forall\, \boldsymbol{\Psi}_v \in X_v \subset H^1(\Omega)^d \end{aligned} \tag{84}$$

where the tensor product : is defined for two arbitrary tensors $\mathbf{P}$ and $\mathbf{Q}$ by:

$$\mathbf{P}:\mathbf{Q} = \mathbf{P}_{ij}\mathbf{Q}_{ji},\ 1\le i,j\le d \tag{85}$$

using the usual convention of summation over repeated indices. The FENE-P or FENE-CR constitutive equation in weak form reads:

$$\begin{aligned} C(\mathbf{v},\mathbf{A},\boldsymbol{\Psi}_A) = \quad & \tfrac{We}{Re}\int_\Omega (1-\tfrac{tr(\mathbf{A})}{L_e^2})\tfrac{\partial\mathbf{A}}{\partial t} : \boldsymbol{\Psi}_A \, d\Omega + We\int_\Omega \mathbf{A}:\boldsymbol{\Psi}_A \, d\Omega \\ & + We\int_\Omega (1-\tfrac{tr(\mathbf{A})}{L_e^2})((\mathbf{v}\cdot\nabla)\mathbf{A}) : \boldsymbol{\Psi}_A \, d\Omega \\ & - We\int_\Omega (1-\tfrac{tr(\mathbf{A})}{L_e^2})(\nabla\mathbf{v}\cdot\mathbf{A}) : \boldsymbol{\Psi}_A \, d\Omega \\ & - We\int_\Omega (1-\tfrac{tr(\mathbf{A})}{L_e^2})(\mathbf{A}\cdot(\nabla\mathbf{v})^T) : \boldsymbol{\Psi}_A \, d\Omega \\ & - \int_\Omega f(tr(\mathbf{A})\mathbf{I} : \boldsymbol{\Psi}_A \, d\Omega \\ & = 0, \quad \forall\, \boldsymbol{\Psi}_A \in X_A \subset L^2(\Omega)^{d\times d} \end{aligned} \tag{86}$$

In the previous equations, we have adopted the notation $(\nabla\mathbf{v})_{ij} = \frac{\partial v_i}{\partial x_j}$. The viscoelastic stress and configuration tensors are related by Eq. (79).

Instead of solving the previous continuous problem, an approximate projected solution is sought in a finite dimensional subspace $X_p^{N_p}\times X_v^{N_v}\times X_A^{N_A}$ of $X_p\times X_v\times X_A$, where $X_p^{N_p}$, $X_v^{N_v}$ and $X_A^{N_A}$ are respectively included into the space of polynomials of order less than or equal to N_p, N_v and N_A. In practice, some restrictions occur as far as the selection of the polynomial degrees is concerned. In particular, the inf-sup condition imposes restrictions on the pressure subspace once the velocity subspace is prescribed, to prevent locking and spurious oscillation phenomena. In the spectral element context, the inf-sup condition is satisfied if $N_p = N_v - 2$. No such condition has been clearly defined for the viscoelastic or configuration stress subspace although some relevant investigations [32], [36] suggest that the same subspace should be selected for both configuration and viscoelastic stress tensors ($X_A = X_\tau$, $N_A = N_\tau$).

The discretization process leads to a set of equations, which can be cast in matrix form. Adopting the notations of Deville et al. [7], we obtain:

$$-D_k \underline{V}_k = 0, \;\; k = 1, .., d \tag{87}$$

$$\begin{aligned} M \, \frac{d \, \underline{V}_i}{dt} = \quad & -Re \, C(\underline{V}) \, \underline{V}_i + D_i^T \, \underline{P} - R_\mu \, K_{ij} \, \underline{V}_j \\ & -VE_j \, \underline{T}_{ij} + (M_b)_{ij} \, \underline{S_b}_j \end{aligned} \tag{88}$$

The symbols $\underline{P}$, $\underline{V}_i$ and $\underline{T}_{ij}$ correspond to the vectors relative to the degrees of freedom for the pressure, velocity and viscoelastic stress components ($1 \leq i \leq d$, $1 \leq j \leq d$). The matrices M and K_{ij} are respectively diagonal mass and stiffness matrices. The discrete divergence operator is given by D_i and its transpose D_i^T yields the discrete gradient operator. The convection term is given by $C(\underline{V}) \, \underline{V}_i$ where the nonlinear operator $C(\underline{V})$ depends on the velocity field.

The last two matrix operators are derived from the following integrals:

$$\int_\Omega \boldsymbol{\tau} \cdot \nabla \boldsymbol{\Psi}_v \, d\Omega \Longrightarrow \; VE_j \, \underline{T}_{ij}, \; \int_{\partial\Omega} (\boldsymbol{\sigma} \cdot \mathbf{n}) \cdot \boldsymbol{\Psi}_v \, d\Gamma \Longrightarrow \; (M_b)_{ij} \, \underline{S_b}_j$$

The constitutive equation in semi-discrete form reads:

$$\frac{We}{Re} Tr(\underline{A}_{mn}) \, \frac{d \, \underline{A}_{ij}}{dt} + M \, \underline{A}_{ij} = NL_{ij}(\underline{A}_{mn}, \underline{V}_l) \tag{89}$$

where:

$$NL_{ij}(\underline{A}_{mn}, \underline{V}_l) = -We[CA_{ij}(\underline{V}_l) - GA_{ij}(\underline{V}_l, \underline{A}_{mn}) - (GA)_{ij}^T(\underline{V}_l, \underline{A}_{mn})]$$

Again, these operators written in matrix form are derived from the following integrals:

$$\begin{gathered} \int_\Omega (1 - \tfrac{tr(\mathbf{A})}{L_e^2}) \tfrac{\partial \mathbf{A}}{\partial t} : \boldsymbol{\Psi}_A \, d\Omega \Longrightarrow \; Tr(A_{mn}) \, \tfrac{d \, \underline{A}_{ij}}{dt}, \; \int_\Omega \mathbf{A} : \boldsymbol{\Psi}_A \, d\Omega \Longrightarrow \; M \, \underline{A}_{ij} \\ \int_\Omega (1 - \tfrac{tr(\mathbf{A})}{L_e^2})[(\mathbf{v} \cdot \nabla)\mathbf{A}] : \boldsymbol{\Psi}_A \, d\Omega \Longrightarrow \; CA_{ij}(\underline{V}_l) \\ \int_\Omega (1 - \tfrac{tr(\mathbf{A})}{L_e^2})[\nabla \mathbf{v} \cdot \mathbf{A}] : \boldsymbol{\Psi}_A \, d\Omega \Longrightarrow \; GA_{ij}(\underline{V}_l, \underline{A}_{mn}) \\ \int_\Omega f(tr(\mathbf{A}))\mathbf{I} : \boldsymbol{\Psi}_A \, d\Omega \Longrightarrow \; DT_{ij}(\underline{V}_l, \underline{A}_{mn}) \end{gathered} \tag{90}$$

The function $f(tr(\mathbf{A}))$ is given by either Eq. (81) or (82).
The symbol $\underline{A}_{ij}$ correspond to the vectors relative to the degrees of freedom for the configuration tensor components ($1 \leq i \leq d$, $1 \leq j \leq d$). Following Eq. (79), the configuration and viscoelastic stress vectors are related by a simple transformation:

$$\underline{T}_{ij} = \frac{1 - R_\mu}{We} \frac{1}{1 - \frac{\underline{A}_{kk}}{L_e^2}} \left(\underline{A}_{ij} - f(\underline{A}_{kk}) \, \delta_{ij} \right) \tag{91}$$

where δ_{ij} is the usual Kronecker symbol.

The main problem with this formulation when applied to viscoelastic flows is that it may produce numerical instabilities when the Weissenberg number is increased. Therefore, stabilization techniques have to be introduced to remove these instabilities. In the spectral element context, it is usual to resort to techniques like filtering [35], [34], the SUPG or the discontinuous Galerkin method applied to the constitutive equation and the so-called Discontinuous Elastic Viscous Stress Splitting (DEVSS) method [33] applied to the momentum equation.

Time discretization

The semi-discrete set of nonlinear ordinary equations (87), (88) and (89) must still be approximated in time in order to generate the time marching scheme. However, a careful inspection of this set of equations shows that the coupled solution, especially for a three-dimensional problem, seems to be out of reach, even on the most powerful supercomputers. Therefore, it is advisable to resort to splitting techniques to decouple velocity-pressure and stress computations.

Iterative solvers are used to obtain numerical solutions of the linearized systems appearing once the problem is discretized in space and time with reduced computer memory requirements when the problem size is not small.

Decoupled technique

The present decoupled technique builds upon classical time-dependent Navier-Stokes solvers. The extension to non-Newtonian fluids is based on the explicit time integration of the constitutive relation. Once the viscoelastic stress tensor is known at the new time level, it is incorporated in the Navier-Stokes equations as a source term.

More precisely, a fully decoupled approach consists in solving separately the mass-momentum and constitutive equations. The constitutive equation is integrated at time step n+1 for the configuration tensor while using velocity gradient terms obtained at previous time steps. The viscoelastic stress tensor computed by its expression in terms of the configuration tensor is then inserted as a source term into the momentum equation. The mass-momentum system of equations is then solved for the velocity and pressure at the new time step n+1 using a classical technique for the Navier-Stokes problem. Such stress-velocity decoupled iterative techniques are cheap in terms of CPU time. Various schemes can be used for the temporal discretization of Eqs. (87)-(89) like in particular a backward differentiation formula (BDF) for implicit terms and the extrapolation scheme (EXT) for explicit terms, which leads to the following system of equations. In order to simplify the notation with respect to Eqs. (87)-(89), a shorthand notation is introduced where $\underline{V}$, $\underline{P}$, $\underline{A}$ and $\underline{T}$ represent the full vectors of all velocity pressure, configuration and viscoelastic stress nodal unknowns. The block diagonal matrices $\mathbf{M}$ and $\mathbf{K}$ are respectively composed with d block matrices M and K. The matrix $\mathbf{D}$ corresponds to the full divergence operator and $\mathbf{D}^T$ to the full gradient operator. The symbol $\mathbf{Tr}$ is relative to the trace term in Eq. (89).

$$-\mathbf{D}\underline{V}^{n+1} = 0 \tag{92}$$

$$\left(\tfrac{\beta_{s_i}}{\Delta t}\right)\mathbf{M}\,\underline{V}^{n+1} - \mathbf{D}^T\,\underline{P}^{n+1} + R_\mu\,\mathbf{K}\,\underline{V}^{n+1} = \tag{93}$$

$$\tfrac{1}{\Delta t}\textstyle\sum_{q=1}^{s_i}\beta_{s_i-q}\mathbf{M}\underline{V}^{n+1-q} - \sum_{r=0}^{s_e-1}\alpha_r Re\,\mathbf{C}(\underline{V}^{n-r}) \tag{94}$$

$$-\mathbf{VE}\,\underline{T}^{n+1} + \mathbf{M_b}\,\underline{S_b}^{n+1} \tag{95}$$

$$\underline{T}^{n+1} = \mathbf{TA}\,\underline{A}^{n+1} \tag{96}$$

$$(\frac{We}{Re\,\Delta t}\beta_{s_i}\mathbf{Tr}+\mathbf{M})\underline{A}^{n+1} = \frac{We}{Re\,\Delta t}\sum_{q=1}^{s_i}\beta_{s_i-q}\mathbf{Tr}\underline{A}^{n+1-q} + \sum_{r=0}^{s_e-1}\alpha_r\mathbf{NL}(\underline{A}^{n-r},\underline{V}^{n-r}) \tag{97}$$

where an extrapolation method of order s_e has been used to determine the value of the nonlinear term at time step n+1. The associated BDF scheme is of order s_i. The coefficients α_i and β_j are dependent of the orders of each method, e.g. for a BDF2/EX2

scheme: $\alpha_0 = 2$, $\alpha_1 = -1$, $\beta_0 = \frac{1}{2}$, $\beta_1 = -2$, $\beta_2 = \frac{3}{2}$. Eq. (96) is a short-hand notation for representing Eq. (91).

Since the nonsymmetric (nonlinear) terms in Eqs. (92)-(97) are located in the right-hand sides, cheap preconditioned conjugate gradient solvers can be used to solve the resulting system for the unknowns at the grid points. Eq. (97) is readily solved since there are no implicit terms with non-trivial matrix operators.

The efficiency of this decoupled method depends critically on the availability of an efficient and robust solver for the underlying Stokes problem at each time step. Our approach consists of solving the mass-momentum set of Eqs. (92)-(95) with an efficient solver for Newtonian flows, which can be easily extended to viscoelastic flows by simply adding the divergence term corresponding to the viscoelastic stress tensor as a source term in the momentum equation. The set of mass-momentum equations is treated via a generalized block LU decomposition with pressure correction, which can be summarized by the following steps:

Step 1: Compute the tentative velocity vector $\underline{V}^*$ by solving the equation:

$$\mathbf{H}\,\underline{V}^* = \underline{F} + \mathbf{D}^T\,\underline{P}^n \tag{98}$$

which is a short notation for Eq. (95) where the pressure term has been shifted to the right-hand side. The Helmholtz operator matrix $\mathbf{H}$ comprehends both mass and Laplacian contributions while all other terms including the convective one are in the right-hand side vector $\underline{F}$:

$$\mathbf{H} = \left(\frac{\beta_{s_i}}{\Delta t}\right)\mathbf{M} + R_\mu\,\mathbf{K} \tag{99}$$

$$\underline{F} = \frac{1}{\Delta t}\sum_{q=1}^{s_i}\beta_{s_i-q}\mathbf{M}\underline{V}^{n+1-q} - \sum_{r=0}^{s_e-1}\alpha_r Re\,\mathbf{C}(\underline{V}^{n-r}) \tag{100}$$

$$-\mathbf{VE}\,\underline{T}^{n+1} + \mathbf{M_b}\,\underline{T_b}^{n+1} \tag{101}$$

Step 2: Solve for the pressure:

$$-\mathbf{D}\,\mathbf{Q}\,\mathbf{D}^T\,\underline{\Delta P}^{n+1} = \frac{\beta_{s_i}}{\Delta t}\,\mathbf{D}\,\underline{V}^* \tag{102}$$

$$\underline{P}^{n+1} = \underline{P}^n + \underline{\Delta P}^{n+1} - R_\mu\,\mathbf{D}\,\underline{V}^* \tag{103}$$

The last term in Eq. (103) has been introduced to obtain an overall consistent scheme.
Step 3: Compute the final velocity at step n+1 after a pressure correction:

$$\underline{V}^{n+1} = \underline{V}^* + \frac{\Delta t}{\beta_{s_i}}\mathbf{Q}\,\mathbf{D}^T\,\underline{\Delta P}^{n+1}$$

The choice of the matrix $\mathbf{Q}$ is critical. If one sets :

$$\mathbf{Q} = \mathbf{H}^{-1} \tag{104}$$

one gets back the Uzawa technique or if one writes an approximation of $\mathbf{H}^{-1}$ e.g. for a BDF2/EX2 scheme:

$$\mathbf{Q} = \frac{\Delta t}{\beta_2}\,\mathbf{M}^{-1} - R_\mu\left(\frac{\Delta t}{\beta_2}\right)^2(\mathbf{M}^{-1}\mathbf{K})\,\mathbf{M}^{-1} \tag{105}$$

$$+\,(R_\mu)^2\left(\frac{\Delta t}{\beta_2}\right)^3(\mathbf{M}^{-1}\mathbf{K})^2\,\mathbf{M}^{-1} \tag{106}$$

one obtains a fourth-order approximation in time. If accuracy in time is limited to second order, only the first term in the previous expansion needs to be retained. Efficient preconditioners for the iterative solver of the pressure step Eq. (102), which is by far the most expensive one in terms of CPU time, can be prescribed. Simple diagonal preconditioners are used for the two velocity steps.

3.4.4 Examples of simulations

In this section, a few examples of simulations performed with the spectral element method and algorithm presented before are briefly introduced. All the results presented in the following sections have been obtained with $Re = 1.0$ and a viscosity ratio ($R_{mu} = \frac{1}{9}$) while adjusting the flow rate so that the ratio of characteristic downstream channel velocity to height $\frac{U_d}{D_d}$ would be equal to 1.0.

Generation and decay Poiseuille flow :

Time-dependent Poiseuille flows can been simulated in order to check the validity of the method. A constant non-zero pressure gradient is imposed abruptly to an initially static fluid so that the generation of a steady flow is obtained at sufficiently large times then the gradient is removed when starting from steady-state conditions (at large times, the velocity and viscoelastic stress decay to zero values). Computed values can be compared to analytical ones in the case of an Oldroyd-B fluid. The corresponding time evolution of axial velocity v_x, normal viscoelastic stress τ_{xx} in the flow direction and viscoelastic shear stress τ_{xy} can be seen in Fig. 9 .

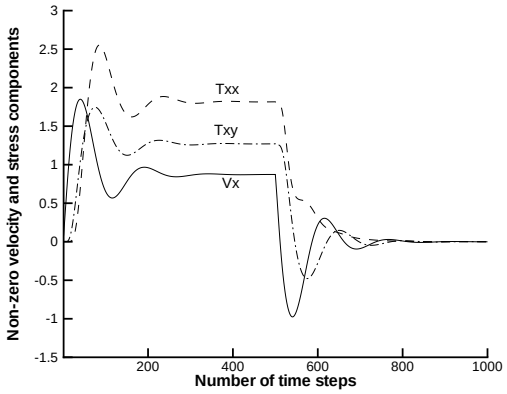


Figure 9: Generation and decay of planar Poiseuille flow of an Oldroyd-B fluid for We=0.5

The viscoelastic character of the flow appears in the oscillations of these various quantities during the unsteady phases of the simulation prior to and after reaching the steady state.

Flow through a 4:1 2D contraction:

A first less trivial example is the simulation of the flow of a viscoelastic fluid of the FENE family in a 2D contraction as shown in Fig. 10. The vortex enhancement mechanism corresponds to the growth of the Newtonian-like recirculation zone located in the salient corner as shown in Fig. 10. Plots (d), (c) and (a) correspond to a FENE-P fluid with $L_e^2 = 6.0$ for respective values of the Weissenberg number equal to 0.0

(Newtonian), 2.0 and 8.0. Interestingly, as the vortex grows under the action of elastic forces, the concavity of its boundary changes. The rotating center moves slightly towards the re-entrant corner as We increases. The reattachment lengths of the vortices visible in Fig. 10 are in agreement with those obtained with other simulation methods.

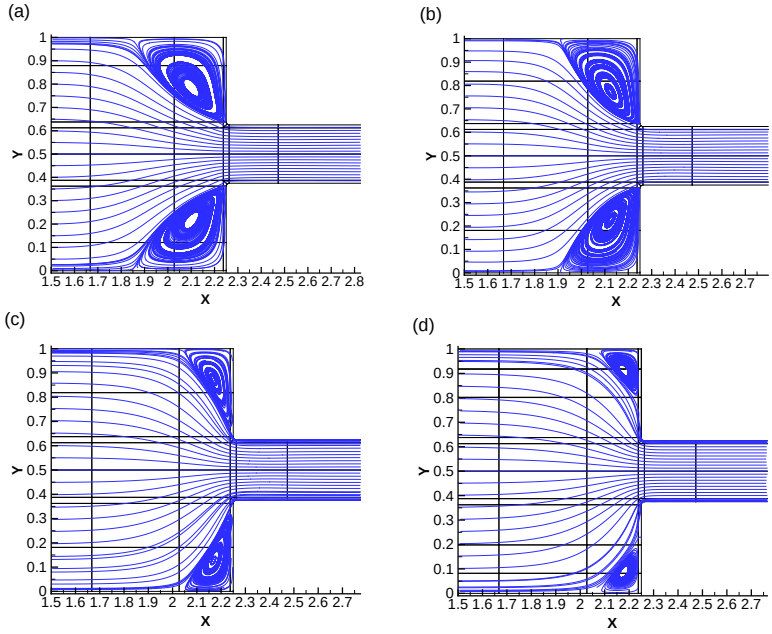


Figure 10: Typical features of the streamlines in a 4:1 contraction at various Weissenberg numbers. (a): FENE-P, $We = 8$; (b): FENE-CR, $We = 8$; (c): FENE-P, $We = 2$; (d): $We = 0$ (Newtonian).

Flow through 3D contractions:

The same type of simulation can also be carried out in 3D for various contraction geometries as shown in Fig. 11. The channel lengths have been taken long enough for accounting for the wrong boundary conditions at entrance (typically $\frac{L_u}{D_u} \approx 10 - 20$). About the same values of the downstream ratio $\frac{L_d}{D_d}$ have also been selected to allow for the obtention of fully developed conditions. We discuss here general features obtained with a given geometry i.e. for a fixed shape of the 45 slanted edge near the re-entrant corner ($\frac{e_s}{D_u} = 0.07$). Uniform polynomial orders equal to 7 were used in all directions. The Reynolds and Weissenberg numbers are defined as in the two-dimensional case. The flow rate is adjusted so that $\frac{U_d}{D_d} = 1.0$. The contraction plane corresponds to $x = 5.0$.

The typical shape of the streamlines for a 4:4:1 square/square contraction is shown in Fig. 12 for a FENE-P fluid. The value of the Weissenberg number is 4.0. Streamlines are represented partly in a half-plane $z = \frac{D_u}{2}$ and in a plane corresponding to a half-diagonal of the contraction channels. The shape of the vortex is visible in that plane.

Three-dimensional effects are also visible in the planar contraction geometry as indicated in Fig. 13. The corresponding value of the Weissenberg number is 1.0. Stream-

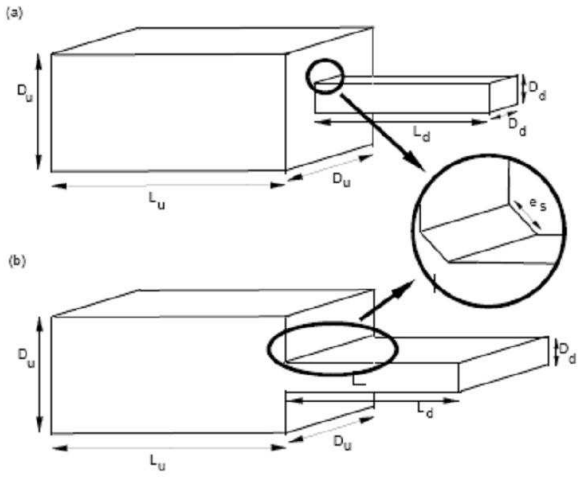


Figure 11: Geometries of the square/square (a) and planar (b) contractions.

lines starting in three planes $z = 0.1\,D_u$, $z = 0.5\,D_u$ and $z = 0.9\,D_u$ are represented. Distortion in the z-direction can be observed near the side walls located in the planes $z = 0$ and $z = D_u$ whereas the contraction is in the y-direction.

Two-dimensional flow through an open cavity

In contrast to what happens in the flow through a contraction depicted in the previous section, the fluid through an open cavity first goes through an expansion before undergoing the effects of the contraction. Depending on the cavity length L, rather different phenomena can be observed as shown in Fig. 14. As opposed to their Newtonian counterparts (see plot (a)), a non-symmetric vortex is present for viscoelastic fluid flows through short cavities (see plots (b)-(d)) in agreement with experimental data. If the ratio between the cavity length and height $\frac{L}{H}$ exceeds some values, the vortex is split into two parts located near the bottom and top corners of the cavity in agreement with what has been observed for the contraction and contraction/expansion geometries.

Two-dimensional flow through a planar T-geometry

Another interesting physical effect with viscoelastic flows is the appearance of vortices when two converging flows with opposite directions are meeting and flowing afterwards together in a third channel as e.g. in a planar T-configuration like the one shown in Fig. 15. Elastic forces induce the onset of recirculation zones of different sizes if the flow rates of the two converging fluids are different, the larger vortex appearing in the channel with the lower flow rate. No such effect is present for a Newtonian fluid (see plot (b)). The vortices are removed if the sharp corners are cut-off and replaced by slanted edges as shown in plot (c). Their presence is closely related to the geometry near corners as noticed for the abrupt contraction.

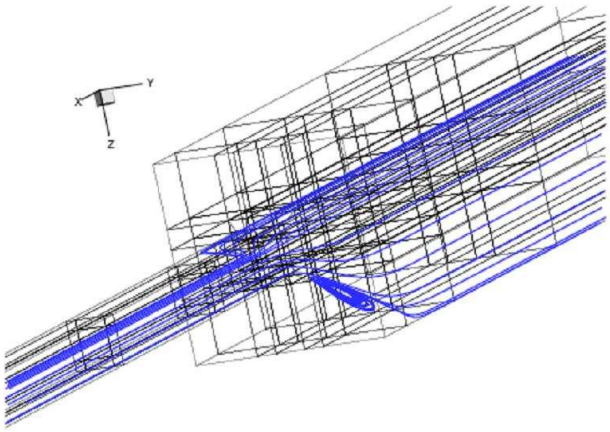


Figure 12: Typical features of the streamlines in a 4:1 square/square contraction for a FENE-P fluid ($We = 4$).

Two-dimensional combined mixing-and-separating flow

Two plates being separated by a gap of width d are located along the axis of a pipe as shown in Fig. 16. This allows flows in opposite directions in the lower left and upper right channels to be injected. Numerically, Dirichlet conditions are imposed at the corresponding locations. Natural boundary conditions are imposed at the two remaining sections where the fluid leaves the computational domain. A sufficiently long pipe has been selected to allow for fully developed conditions at entry and exit. The same fluid is used in the upper and lower part of the pipe.

Typical differences in the streamline pattern between a Newtonian and a viscoelastic liquid are visible in Fig. 16 for $We = 4.0$. The streamlines are perfectly symmetric with respect to the pipe axis in the Newtonian case whereas some asymmetry occurs for a non-Newtonian fluid. The streamline pattern appears to be somewhat twisted under the action of elastic forces. This phenomenon is in agreement with experimental work and computational results. In addition, small recirculation zones are present at the plate rounded tips whereas none are observed for Newtonian fluids. As indicated in plot (a), part of the upper (resp. lower) Newtonian fluid flowing in the vicinity of the right (resp. left) plate is mixed with the lower (resp. upper) fluid flowing in the opposite direction. Therefore, some flow reversal neatly occurs. This mixing process is all the more significant when the gap width d is large. It can be seen in plot (b) that mixing if any is strongly reduced due to elastic effects for the FENE-P fluid used in the simulations. This is in agreement with experimental observations, which indicate that highly elastic fluids tend to prefer unidirectional flow and resist to flow reversal.

Plots (c) and (d) show isocontours of the first two components of the viscoelastic stress tensor τ_{xx} and τ_{xy}. They indicate in particular the location of the stress max-

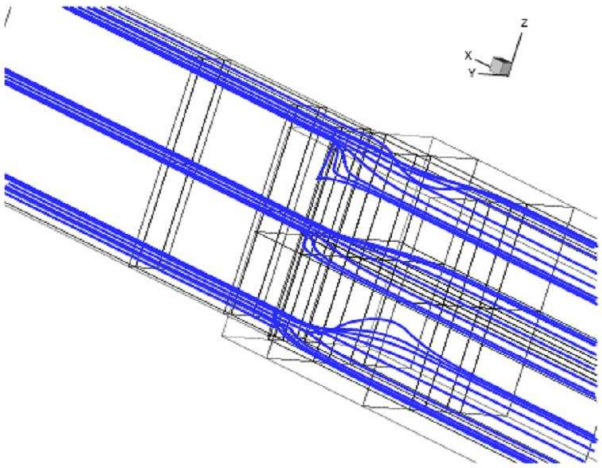


Figure 13: Typical features of the streamlines in a 4:1 planar contraction for a FENE-P fluid ($We = 4$).

ima near the inner plates near the gap where strong shearing between the two fluids flowing in opposite directions occur.

Flow around an array of cylinders in a channel

Another effect appearing in the flow of viscoelastic fluids around an array of cylinders in a channel can be seen in Fig. 17 where streamlines are shown. The difference of the streamline patterns for viscoelastic (plot (b)) and and Newtonian fluids (plot (c)) at the vicinity of one bottom cylinder can be observed. As opposed to the Newtonian case where a perfectly symmetric pattern is obtained near the stagnation points, the position of these points is shifted in the viscoelastic case and the corresponding symmetry disappears.

4 The spectral element code SPECULOOS

4.1 Overview

SPECULOOS is a spectral and mortar element analysis software for the numerical solution of partial differential equations, in particular incompressible unsteady fluid flow problems described by the Navier-Stokes equations. It aims at providing the user with a flexible tool for research in spectral element methods [1]. **SPECULOOS**' main features are:

- One-, two- and three-dimensional curved geometries
- Quadrilateral and hexahedral spectral elements
- Equations models: Helmholtz, heat equation, Stokes and Navier-Stokes

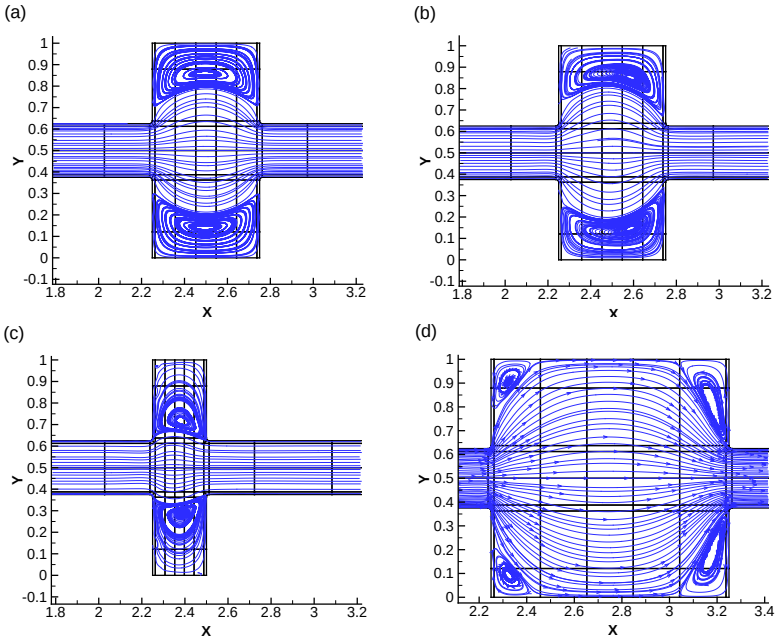


Figure 14: Typical features of the streamlines in open cavities of variouy length. (a): $L/H = 0.5$, Newtonian; (b): $L/H = 0.5$, $We = 5.0$; (c): $L/H = 0.25$, $We = 5.0$; (d): $L/H = 1.0$, $We = 5.0$. The inflow and outflow pipe heights are equal to one fourth the cavity height.

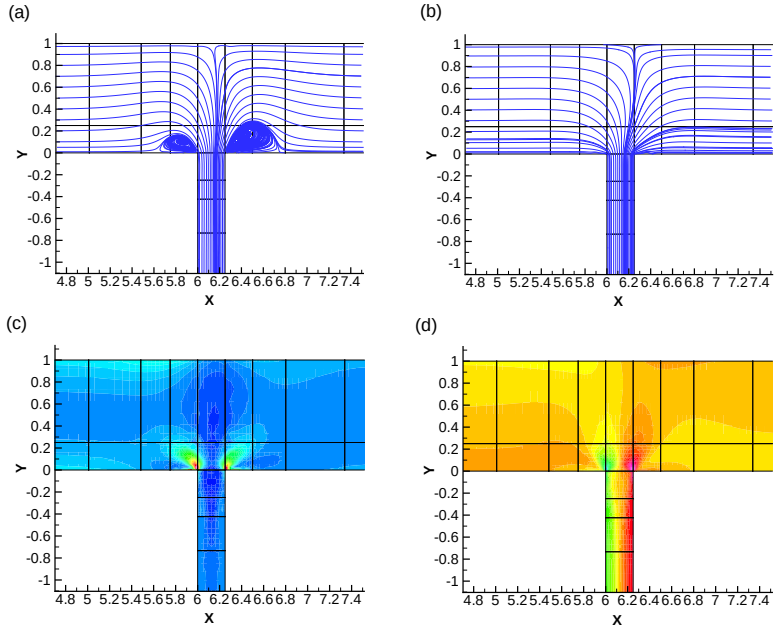


Figure 15: Typical features of the flow characteristics in a T-shape channel. The flow rate of the fluid flowing from the right is half that of the fluid flowing from the left. Streamlines in a FENE-P fluid ($We = 12.0$) (a), in a Newtonian fluid with sharp corner (b) ; Viscoelastic stresses in a T-shape channel (FENE-P fluid, $We = 12.0$); (c): τ_{xx}; (d): τ_{xy};

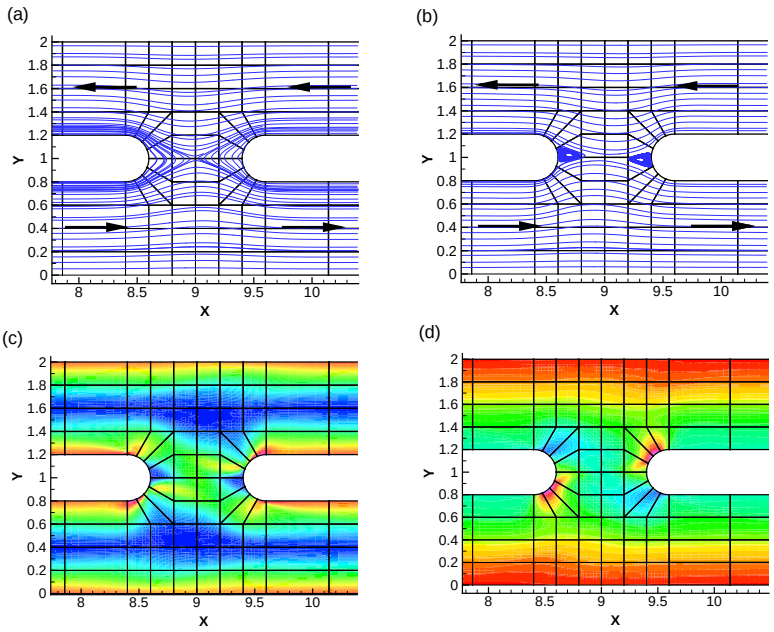


Figure 16: Typical features of the computed flow characteristics in a combined-mixing-and-separating device. (a): streamlines in a Newtonian fluid; (b): streamlines in a viscoelastic (FENE-P, $We = 4.0$) fluid; (c) and (d): τ_{xx} and τ_{xy} components of the viscoelastic stress.

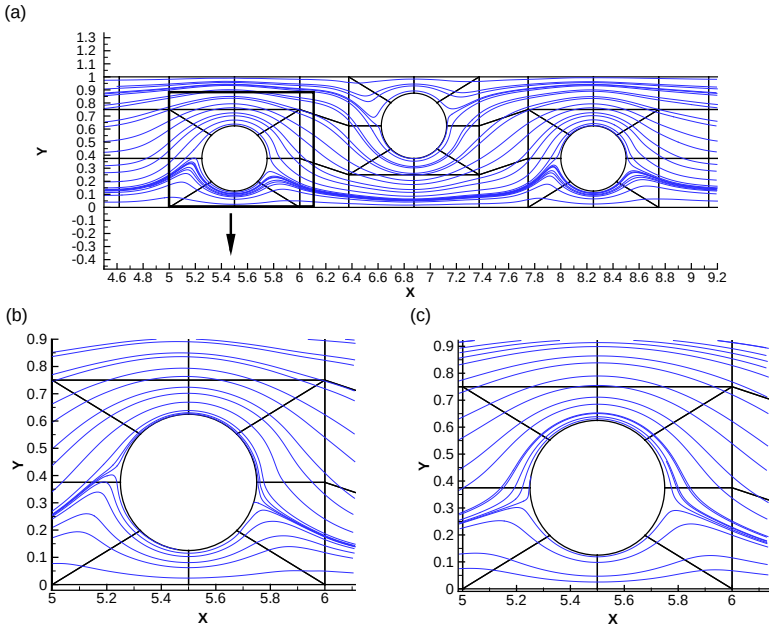


Figure 17: Streamlines in the flow of a fluid around an array of cylinders in a channel. global view (a) ; zoom in a window around one cylinder streamlines related to a viscoelastic (FENE-P) fluid (b) and to a Newtonian fluid (c).

- Time integration by finite-difference models
- Solvers: direct, preconditioned conjugate gradient, preconditioned Uzawa, pressure correction, splitting
- Mortar elements
- Parallelism

4.2 Object-oriented programming

It is evident that mixing spectral elements and mortars in the framework of the Navier-Stokes equations in complex three-dimensional geometries is a programming challenge. A design requirement for this computer implementation is that it should behave as a user-friendly tool for programmers, not just as a computational black box suited for a limited set of predefined problem types. In particular the program should be insofar as possible easily transferable from one programmer to another. This led the authors to choose a high-level programming approach, namely object-oriented programming. It is hoped that consistent use of the powerful concepts of data abstraction (mainly, data encapsulation, class inheritance and polymorphism) will help to fulfill the expectations as regards flexibility. The C++ language was chosen as implementation language, on behalf of several key advantages: beside the basic fact that it encompasses data abstraction, it is a non-proprietary, popular and intrinsically efficient language.

4.3 Collaborative programming and open-source license

The purpose of SPECULOOS is to serve the community of scientists and engineers working with spectral elements. It is therefore developed collaboratively by various researchers, and it is available for download to a general audience, under the code name OpenSPECULOOS. The download page is at the address `www.sourceforge.net/openspeculoos`. It may be used under the terms of a GPLv3 open source license. This license warrants the freedom of downloading, using, copying, modifying and redistributing the code. Modified and redistributed versions of the code must however always be available under a GPLv3 or any compatible license, which means that it is not allowed to leave the context of open source code development. Furthermore, we ask you as a matter of acknowledgment for the titanic work which was invested in the development of SPECULOOS to cite the following papers in journal publications containing numerical results simulated with SPECULOOS: ref. [7], ref. [8], and ref. [9].

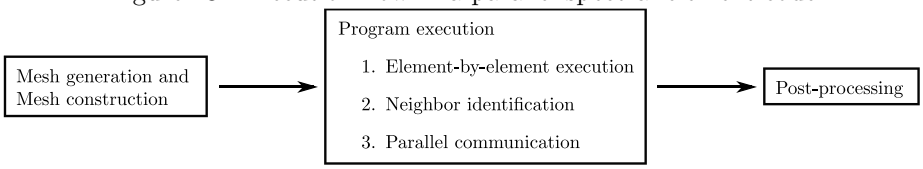
5 High performance computing with spectral elements

5.1 Algorithmic complexity of spectral element codes

A first step towards analyzing the features of a CFD code is made by understanding its algorithmic complexity. A complexity analysis is the asymptotic evaluation of the time needed for the program to solve the problem. Although such an analysis does not tell exactly how fast the program will run, as this obviously depends on many software and hardware parameters, it gives an estimate of how the computation time evolves when either the space resolution or the polynomial degree of the interpolants in the solver are increased.

Let's call E the typical number of elements along one of the space directions, and N the characteristic degree of the polynomials representing the velocity fields. Thus, unless the space resolution or the polynomial degree vary substantially from one space direction to another, the total number of elements amounts roughly to E^3, and the

Figure 18: Execution flow in a parallel spectral element code



number of collocation points inside an element is equal to $(N+1)^3$. Using these variables, we will now estimate the computational effort required to advance one time step in the solver.

In the approach to spectral elements taught in this course, the algorithm is dominated by an iterative solution of a linear equation (*e.g.* for the Helmholtz or for the pressure problem) by a conjugate gradient method. We know from experiment that the number of iterations of the conjugate gradient solver increases linearly with the space resolution E, and linearly as well with the polynomial degree N. Actually, these performance figures are observed with preconditioned problems, even with such simple preconditioners as the diagonal one, whereas they can be substantially worse on a non-preconditioned problems.

At each iteration of the conjugate gradient solver, the same operation is performed on each of the E^3 elements. Therefore, an additional factor E^3 is included in the complexity estimate to account for the act of iterating over each element.

The actual computations consist mostly of applying discrete linear operators to the unknowns, such as, the derivative operator. In a 3D simulation, these operators are represented by a matrix of size $(N+1)^3$-by-$(N+1)^3$. Applying such an operator in a naive way is therefore an operation of complexity N^6. Exploiting the fact that these operators are tensor products of 1D operators, they can however be applied with a complexity N^4.

In summary, stepping from time n to time $n+1$ takes a time T which is asymptotically proportional to

$$T \sim E^4 \cdot N^5. \tag{107}$$

5.2 Parallelizing spectral element codes

A convenient way of parallelizing spectral element codes is to attribute each element to a processor and to execute all operations on the element by this processor. After each conjugate gradient iteration, values on the collocation points on the surface of the elements need to be communicated from one processor to another. Consider a situation of aggressive parallelism in which each processor acts on exactly one element. In this case, the number of operations executed by the processor during the conjugate gradient iteration has a leading complexity of order N^4, while the amount of data communicated by the processor with its neighbors is equal to the number of collocation points on the surface of the element, and thus proportional to N^2. This guarantees an excellent ratio of $N^4/N^2 = N^2$ between computation and communication. Parallel spectral element codes exhibit therefore an excellent scalability. Furthermore, these characteristics improve with increasing polynomial degree. As a disadvantage of this approach, one should point out that the granularity of the parallelization is limited by the size of an element. Unless an alternative parallelization technique is used, it is impossible to use a number of processors which is larger than the number of elements. Moreover, the number of elements needs to exceed the number of processors significantly in order to allow for the possibility to balance the work load between processors in simulations with inhomogeneous domains.

Figure 18 illustrates the global execution flow in a generic CFD program. In a

first step a mesh for the computational domain is either generated or read from a file, previously produced by an external mesh generator. From the mesh information, the data structure is constructed, including for example memory allocation for the collocation points of the elements local to the processor. As a second step follows the execution of the actual program, consisting of the iterative application of operators to each element. In a cyclic way, information is furthermore exchanged between element surfaces. In order to do this, an element must identify its neighbors and instantiate inter-processor communication for neighboring elements located on remote processors. Finally, the last step in the execution flow consists of evaluating the simulated data, or writing the data to a file for future post-processing.

An overwhelming majority of the execution time is usually spent in the second step, the iterative execution of the spectral element algorithm. Luckily enough, this part is also simplest to parallelize, because the conceptual domain decomposition into spectral elements is directly re-interpreted as a domain decomposition into parallel execution units. Parallelizing a spectral element code means therefore, before all, parallelizing this central iterative part of the algorithm. One should nevertheless remain aware of the possible impact of the initial and final stage on the overall program performance. While these stages usually are negligible in a sequential program, their relative importance increases when the main part of the algorithm is parallelized. This can go so far that the total execution time is dominated by the setup of the mesh, and by reading and writing data. It is therefore important to either parallelize the initialization and post-processing as well, or to implement them with algorithms which are sufficiently fast for the considered problem. The latter approach was chosen in SPECULOOS, in which structural operations for generating and setting up the mesh are not parallelized, but sufficiently efficient in most cases thanks to a careful choice of algorithms.

Mesh generation can be an especially challenging task. For a non-structured mesh, a general algorithm for mesh generation implemented in a straightforward way has exponential complexity. This means that the time required to generate the E^3 elements evolves like $\exp(E^3)$ with an increasing number of elements. This prohibitive time can and must be substantially reduced by the use of various heuristic approaches. Moreover, the complexity of mesh generation is dramatically improved by exploiting symmetries, regularities or other type of information on the nature of the domain and the mesh. The simplest case occurs for example when the end-user of the CFD software has a detailed knowledge of the desired mesh and is able to place explicitly the coordinates of each element. In this case, the mesh can be generated with an optimal, linear complexity E^3.

5.3 Extreme-scale parallelism

Extreme-scale parallelism is a term used for program execution on parallel platforms which are at the high-end of current hardware capacities. The typical number of computational cores on such machines ranges from 10^3 to 10^5. The availability of such massive parallel platforms impacts on the type of algorithms and arguments used for the development of parallel programs. Indeed, it is common for programs which are designed for modest parallel platforms (with for example $10^0 - 10^2$ nodes) to be split into parallel and sequential parts. Sequential parts are not parallelized because they are considered to take a negligible part in the overall program execution time. Such a sequential program component appears for example when the communication between nodes of the parallel machine is managed through a centralized approach: one processor (or sometimes even all processors) possesses a global view over the distribution of elements over the processors and determines communication patterns between them. Although this operation is not parallelized (one processor needs to iterate over all elements) it is often negligibly small. After all, it involves only a few

logical operations, and no numerical calculations.

On the other hand, such supposedly negligible program components can become dominant in face of extreme-scale parallelism. A sequential component which uses as little as 0.1% of the total execution time on one processor uses more than 50% of the execution time on 10^3 processors. This type of arguments leads to the "no-sequential rule" for extreme-scale parallelism: taking into account the likelihood that the number of processors increases by orders of magnitude in the future, the only way to guarantee the scalability of a parallel code is to fully eliminate all sequential program parts.

In the following, we characterize the performance of parallel programs, depending on scalability features of both the program and the interconnection network of the parallel machine. The total time of the program is split into three terms as follows:

$$T = \frac{T_{\text{main}}}{p} + T_{\text{non-parallel}} + T_{\text{comm}}, \tag{108}$$

where p stands for the number of processors, T_{main} corresponds to the sequential execution time of fully parallelizable program components, $T_{\text{non-parallel}}$ describes the execution time of entirely sequential components whose execution time is independent on the number of processors. The parameter T_{comm} finally stands for the time during which processors wait to receive messages from other processors through the network. We now assume that the program has been developed in the spirit of extreme-scale parallelism, in which case $T_{\text{non-parallel}} = 0$. This is true in SPECULOOS, except for a non-parallelized component in grid generation and in post-processing.

The communication time depends on hardware characteristics of the network. In a network with ideal scalability the communication time depends on the amount of data transferred between pairs of processors, but not on the total number of processors or the total amount of data communicated in the network. Thus, the communication time is

$$T_{\text{comm}} \sim \frac{E^3 N^2}{p}. \tag{109}$$

At the other extreme one may find a network which is entirely unable to scale, and which takes a time proportional to the number of processors to communicate data:

$$T_{\text{comm}} \sim E^3 N^2. \tag{110}$$

In practice, the communication time is contained between these two extremes, depending on the features of the network.

We now consider the execution time of a problem of a given constant size, which is executed on a varying number of processors. On a single processor, the execution time is

$$T(p = 1) = T_{\text{main}} \tag{111}$$

On a parallel machine, the execution time is

$$T(p) = \frac{T_{\text{main}}}{p} + T_{\text{comm}} \tag{112}$$

The speed up of a program restricted by communication time is

$$S(p) = \frac{T(p = 1)}{T(p)} = \frac{T_{\text{main}}}{\frac{T_{\text{main}}}{p} + T_{\text{comm}}}, \tag{113}$$

and the efficiency of parallelization is defined as the ratio between the speedup and an ideal speedup:

$$\text{Eff}(p) = \frac{S(p)}{p} = \frac{T_{\text{main}}}{T_{\text{main}} + pT_{\text{comm}}}. \tag{114}$$

On a non-scalable network, as characterized by Eq. 109, the obtained efficiency is therefore

$$\mathrm{Eff}(p) = \frac{T_{\mathrm{main}}}{T_{\mathrm{main}} + p\kappa}, \tag{115}$$

where κ is a constant for the total communication load associated to the constant computational domain. In an ideally scalable network, characterized by Eq. 110, the efficiency is

$$\mathrm{Eff}(p) = \frac{T_{\mathrm{main}}}{T_{\mathrm{main}} + \kappa}, \tag{116}$$

and thus independent of the number of processors p. In conclusion, the performance figure can be quite pessimistic, as predicted by Equation 115, or optimistic as in Equation 116, depending on the performance of the interconnecting network. In the next section, a benchmark problem coded in SPECULOOS is tested on a modern parallel computer with excellent interconnecting network, and the results are discussed in view of these predictions.

5.4 A parallel benchmark case with SPECULOOS

In this section, the parallel performance characteristics of a spectral element code using SPECULOOS is presented on a IBM Blue Gene/L parallel computer. The benchmark problem consists of the resolution of the time-dependent, incompressible Navier-Stokes equations in a lid-driven three-dimensional cavity. The computational domain has a cubic $L \times L \times L$ shape which is subdivided into $E \times E \times E$ elements. The underlying quadrature rules involve the Gauss-Lobatto-Legendre nodes for the velocity field and the Gauss-Legendre grid for the pressure. The velocity field is represented by polynomials of degree $N = 12$ in each direction, while the pressure field uses polynomials of degree $N - 2 = 10$. The time integration algorithm uses a backward differentiation formula (Euler) scheme of order two (BDF2). The velocity and the pressure equations are decoupled by using the pressure correction approach described in Section 1.2

Figure 19 displays the speed up curve of executing **SPECULOOS** on a number of cores varying between 1024 and 8192. The total size of the problem is kept constant at a value of $E^3 = 8192$ elements, such that every core holds exactly one element when the full Blue Gene is used. It is seen that the program scales well up to the full size of the Blue Gene. Measurements show that the scalability is only limited by MPI communication between nodes which, in the regime with 8192 cores, makes up for less than 20% of the full execution time. It should be emphasized that the benchmark is extremely ambitious, because it is executed in the so-called strong limit in which the observed performance is often poor. Observing the strong limit means to keep the size of the problem constant as the number of cores is increased, instead of scaling the problem size with the size of the parallel machine. In this limit, the amount of per-core data decreases as the number of nodes is increased, which results in a bad balance between the time used for computations and communications. The efficiency observed on Fig. 19 in a strong limit and on thousands of cores reflects both the excellent scalability of the code and the good interconnecting network of the Blue Gene/L.

References

[1] N. Bodard, Interaction fluide-structure par la méthode des éléments spectraux, Ph. D. thesis 3503, Ecole Polytechnique Fédérale de Lausanne, Switzerland, 2006.

[2] N. Bodard, M. O. Deville, *Fluid-structure interaction by the spectral element method*, J. Sci. Comput., **27**, (2006) 123-136.

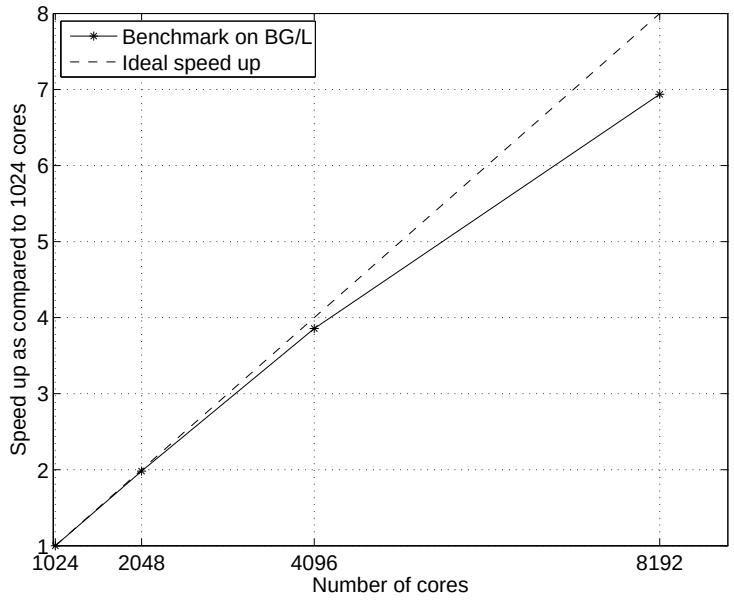


Figure 19: Scalability of **SPECULOOS** on a Blue Gene/L for the resolution of the 3D Navier-Stokes equations

[3] N. Bodard, R. Bouffanais, M. O. Deville, *Solution of moving-boundary problems by the spectral element method*, Appl. Num. Math., **58**, 969–984, 2008.

[4] R. Bouffanais, M. O. Deville, P. F. Fischer, E. Leriche, D. Weill *Large-eddy simulation of the lid-driven cubic cavity flow by the spectral element method*, **27**, (2006) 151–162.

[5] R. Bouffanais, M. O. Deville, E. Leriche, *Large-eddy simulation of the flow in a lid-driven cubical cavity*, Phys. Fluids, 055108, 2007.

[6] H.J.J. Clercx, C.-H. Bruneau, The normal and oblique collision of a dipole with a no-slip boundary, Comput. Fluids, **35**, 245–279, 2006.

[7] M.O. Deville, P. F. Fischer, E. H. Mund. *High-order methods for incompressible fluid flow*. Cambridge University Press, Cambridge, ISBN 0 521 45309 7, 499 pages, 2002.

[8] R. Bouffanais, Simulation of shear-driven flows: transition with a free surface and confined turbulence, Ph. D. thesis, **(3837)**, Ecole Polytechnique Federale de Lausanne, 2007.

[9] Y. Dubois, V. Van-Kemenade, M. Deville, An object-oriented toolbox for spectral element analysis, J. Sci. Comput., **14**, 129, 1999.

[10] N. Fiétier, M.O. Deville, Time-dependent algorithms for the simulation of viscoelastic flows with spectral element methods: applications and stability, J. Comp. Phys., **186**, 93121, 2003.

[11] N. Fiétier, M.O. Deville, Linear stability analysis of time-dependent algorithms with spectral element methods for the simulation of viscoelastic flows, J. Non-Newtonian Fluid Mech., **115**, 157190, 2003.

[12] N. Fiétier, Detecting instabilities in flows of viscoelastic fluids, Int. J. Numer. Meth. Fluids, **42**, 13451361, 2003.

[13] D. Weill, M.O. Deville, "Steady gap flows by the spectral and mortar element method, J. Sci. Comput., **17**, 639–648, 2002.

[14] L. Formaggia, F. Nobile, *A stability analysis for the arbitrary Lagrangian Eulerian formulation with finite elements*, East-West J. Num. Math. 7 (1999) 105–132.

[15] F. Nobile, *Numerical approximation of fluid-structure interaction problems with application to Haemodynamics*, Ph.D. thesis, no. 2458, Swiss Federal Institute of Technology, Lausanne (2001).

[16] R. Bouffanais, M. O. Deville, *Mesh update techniques for free-surface flow solvers using spectral element method*, J. Sci. Comput. 27 (2006) 137–149.

[17] A. Quarteroni, A. Valli, *Numerical Approximation of Partial Differential Equations, Springer Series in Computational Mathematics*, Springer, Berlin, 1994.

[18] L.-W. Ho, A. T. Patera, *Variational formulation of three-dimensional viscous free-surface flows: Natural imposition of surface tension boundary conditions*, Int. J. Numer. Methods Fluids 13 (1991) 691–698.

[19] E. M. Rønquist, *A Domain Decomposition Solver for Three-Dimensional Steady Free Surface Flows*, in: P. Bjørstad, M. Espedal, D. Keyes (Eds.), Domain Decomposition 9 Proceedings, Wiley, New York, 1998, pp. 792–801.

[20] Y. Maday, A. T. Patera, E. M. Rønquist, *The $\mathbb{P}_N \times \mathbb{P}_{N-2}$ method for the approximation of the Stokes problem*, Tech. Rep. 92009, Department of Mechanical Engineering, MIT, Cambridge, MA (1992).

[21] G. E. Karniadakis, M. Israeli, S. A. Orszag, *High-order splitting methods for the incompressible Navier–Stokes equations*, J. Comp. Phys. 97 (1991) 414–443.

[22] J. B. Perot, *An analysis of the fractional step method*, J. Comp. Phys. 108 (1993) 51–58.

[23] J. B. Perot, *Comments on the fractional step method*, J. Comp. Phys. 121 (1995) 190–191.

[24] N. Bodard, M. O. Deville, *Fluid-structure interaction by the spectral element method*, J. Sci. Comput. 27 (2006) 123–136.

[25] L.-W. Ho, A. T. Patera,*A Legendre spectral element method for simulation of unsteady incompressible viscous free-surface flows*, Comput. Methods Appl. Mech. Engrg. 80 (1990) 355–366.

[26] R. Bouffanais, D. Lo Jacono, *Transitional cylindrical swirling flow in presence of a flat free surface*, Computers and Fluids, In Press (2009).

[27] R. Bouffanais, D. Lo Jacono, *Unsteady transitional swirling flow in presence of a moving free surface*, Phys. Fluids, Submitted (2009).

[28] S. Stolz, N. A. Adams, *An approximate deconvolution procedure for large-eddy simulation*, Phys. Fluids. 11 (1999) 1699–1701.

[29] E. Leriche, S. Gavrilakis, *Direct numerical simulation of the flow in the lid-driven cubical cavity*, Phys. Fluids. 12 (2000) 1363–1376.

[30] Y. Zang, R. L. Street, J. R. Koseff, *A dynamic mixed subgrid-scale model and its application to turbulent recirculating flows*, Phys. Fluids A 5 (1993) 3186–3193.

[31] M. A. Habisreutinger, R. Bouffanais, E. Leriche, M.O. Deville, A coupled approximate deconvolution and dynamic mixed scale model for large-eddy simulation, J. Comput. Phys., **224**, 241-266, 2007.

[32] M.I. Gerritsma, T.N. Phillips, Discontinuous spectral element approximations for the velocity-pressure-stress formulation of the Stokes problem, Int. J. Numer. Meth. Engng., **43**, 1401-1419, 1998.

[33] R. Guénette, M. Fortin, A new mixed finite element method for computing viscoelastic flows, J. Non-Newtonian Fluid Mech. **60**, 27-52, 1995.

[34] X. Ma, V. Symeonidis, G.E. Karniadakis, A spectral vanishing viscosity method for stabilizing viscoelastic flows, J. Non-Newtonian Fluid Mech. **115**, 125155, 2003.

[35] J.S. Mullen, P.F. Fischer, Filtering techniques for complex geometry fluid flows, Comm. Numer. Meth. Engrg. **15**, 9-18, 1999.

[36] V. Van Kemenade, M.O. Deville, Application of spectral elements to viscoelastic creeping flows, J. Non-Newtonian Fluid Mech. **51**, 277-308, 1994.