High-order ALE schemes for incompressible capillary flows

Felipe Montefuscolo[1,*], Fabricio S. Sousa[1], Gustavo C. Buscaglia[1]

[a]*Departamento de Matemática Aplicada e Estatística*
Instituto de Ciências Matemáticas e de Computação - ICMC
Universidade de São Paulo - Campus de São Carlos
Caixa Postal 668, 13560-970 São Carlos, SP, Brazil

Abstract

The spatial discretization of problems with moving boundaries is considered, incorporating the temporal evolution of not just the mechanical variables, but also of the nodal positions of the moving mesh. The outcome is a system of Differential-Algebraic Equations (DAE) of index two or, in the case of inertialess flow, just one. From the DAE formulation it its possible to define strategies to build schemes of arbitrary accuracy. We introduce here several schemes of order two and three that avoid the solution of a nonlinear system involving simultaneously the mechanical variables and the geometrical ones. This class of schemes has been the one adopted by the majority of practitioners of Computational Fluid Dynamics up to now. The proposed schemes indeed achieve the design accuracy, and further show stability restrictions that are not significantly more severe than those of popular first order schemes. The numerical experimentation is performed on capillary problems, discretized by both div-stable (P_2/P_1, P_1^+/P_1) and equal-order (P_1/P_1, stabilized) finite elements, and incorporating surface tension and triple (contact) line effects.

Keywords: Finite element method, arbitrary Lagrangian-Eulerian, differential algebraic-equation, geometry extrapolation, capillarity

*Corresponding author

Email addresses: `felipemt@icmc.usp.br` (Felipe Montefuscolo), `fsimeoni@icmc.usp.br` (Fabricio S. Sousa), `gustavo.buscaglia@icmc.usp.br` (Gustavo C. Buscaglia)

1. INTRODUCTION

The numerical simulation of flows with interfaces such as capillary flows is frequently carried out using the arbitrary Lagrangian-Eulerian (ALE) formulation ([1, 2, 3, 4]). In ALE formulations the mesh is moved along the simulation in such a way that its cell-to-cell interfaces coincide with those of the model, which could be material discontinuities or phase boundaries, for example. The mesh update scheme must be able to accompany the model's interfaces without generating severely distorted cells, but it is otherwise free ("arbitrarily" free) to choose the placement of vertices, edges and facets.

In capillary flows very complex phenomena take place that challenge the accuracy of numerical methods. By geometrically aligning the mesh and the interpolation with the model's interfaces, ALE formulations provide the most accurate (in a *per unknown* basis) representations of surface forces, such as surface tension or membrane stresses. Conversely, handling topological changes (e.g. coalescence) with this method is not straightforward and re-computing all matrices at each time step (which can be avoided using, e.g., the immersed interface method [5]) may be expensive. Methods of high order in time may be used to alleviate this cost.

High-order ALE methods are scarce in the available literature. Farhat and coworkers, among others, proposed such methods for the compressible case, considering not fluid interfaces but fluid-structure interaction [6, 7, 8]. These methods adopted the conservative form of the discrete problem, which raises issues of stability linked to the so-called Geometric Conservation Law (GCL) [9, 10, 11, 12, 13, 14, 2]. The GCL is automatically satisfied by ALE formulations in *non-conservative* form, which is the one adopted in this work.

The incompressible case has been considered by Etienne et al [15], who failed to obtain second order accuracy with the non-conservative form, and by Liu [16], who proposed schemes of up to fifth order. These authors, however, only analyze problems with prescribed mesh motion and, in particular, no capillary effects. To our knowledge, ALE methods for incompressible flows with significant capillary effects (and thus with the interface location as an unknown of the problem) have only been studied by Ganesan and Tobiska [17], by

Gerbeau et al [2] and by Bänsch and Weller [18]. The first two references consider just first-order schemes in time, while the third one discusses second-order fully implicit methods which are extremely costly in memory and computer effort (though they are quite attractive in terms of stability).

An attempt at a compact and rigorous presentation of a non-conservative ALE formulation of incompressible flows with significant capillary effects is carried out below. Upon space discretization by finite elements, the coupled problem involving both the fluid unknowns and the mesh variables naturally leads to a differential-algebraic equation (DAE) system of index two in the four variables $\mathbf{U}$, $\mathbf{P}$, $\mathbf{V}$ and $\mathbf{X}$ (which correspond to the coefficient vectors of fluid velocity, fluid pressure, mesh velocity and nodal position, respectively).

Temporal discretizations developed for generic DAEs (such as those in, e.g., [19, 20]) are not necessarily the most appropriate methods for the considered problem. They lead to huge, nonlinearly coupled systems that are very expensive to solve in terms of computer time and memory [18]. Instead, we propose here several discretizations of the DAE in which the mesh variables are solved separately from the fluid ones. This decoupling of unknowns involves *extrapolation* of the nodal positions, in the same spirit of what is done by Farhat and coworkers [8] for fluid–structure interaction (FSI) problems. The proposed methods follow a general strategy that can, in principle, produce methods of arbitrary order of temporal accuracy.

Numerical experiments show that optimal accuracy of order two or higher can be attained for all variables, even in challenging problems such as those involving contact lines (lines at which the fluid interface intersects a solid boundary). Experiments also show that the increased accuracy comes with practically no cost in terms of stability, since the maximum time steps allowed by the proposed high-order methods are practically the same as those of the basic, first-order staggered scheme (used, for example, in [17]).

2. The dynamic wetting equations

As in most of the literature on capillary phenomena, we focus on incompressible flows of Newtonian fluids in which surface inertia and surface viscosity are neglected. The specific

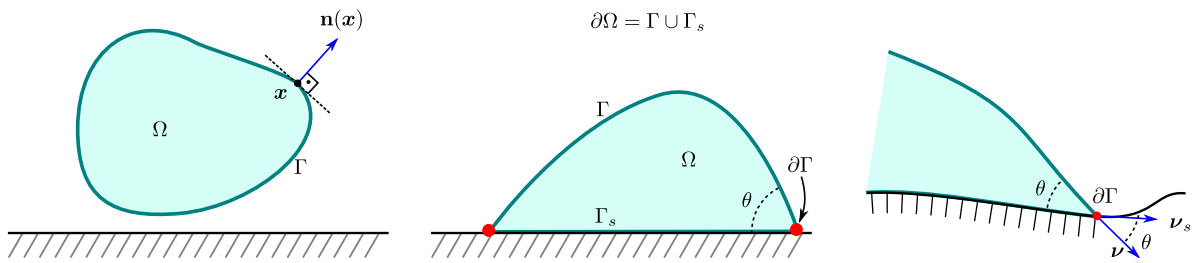

Figure 1: Geometrical setting. Red dots represent the contact lines. The angle θ is called contact angle. The unitary vectors $\boldsymbol{\nu}$ and $\boldsymbol{\nu}_s$ are both perpendicular to contact line and parallel to free surface and solid surface, respectively.

case of solid-fluid-fluid flow is considered in which the second fluid is modeled as a constant pressure boundary condition, as is typical for solid-liquid-gas systems. The solid is assumed rigid. The results extend easily to problems in which both fluids have relevant dynamics. The numerical treatment of a possible deformability of the solid is not addressed.

The Navier-Stokes equations describing the motion of a liquid in time dependent region $\Omega(t) \subset \mathbb{R}^d$ read:

$$\frac{\partial(\rho\mathbf{u})}{\partial t} + \mathbf{u}\cdot\nabla(\rho\mathbf{u}) = -\nabla p + \nabla\cdot(2\mu D\mathbf{u}) + \rho\boldsymbol{g}, \qquad \mathbf{x}\in\Omega(t), \tag{1a}$$

$$\nabla\cdot\mathbf{u} = 0, \qquad \mathbf{x}\in\Omega(t), \tag{1b}$$

where $\mathbf{u}$ is the fluid velocity, p the pressure, ρ the mass density, μ the dynamic viscosity, $\boldsymbol{g}$ a body force (e.g. gravity) and $D = {}^1\!/_2(\nabla + \nabla^T)$ the symmetric gradient operator. To model the capillary forces at the liquid-gas interface (Γ), at the liquid-solid interface (Γ_s) and at the triple-contact line ($\partial\Gamma_s$), the following boundary conditions are adopted [21]:

$$\boldsymbol{\sigma}\cdot\mathbf{n} = -\gamma\kappa\mathbf{n} + \nabla_\Gamma\gamma, \qquad \mathbf{x}\in\Gamma, \tag{2a}$$

$$(\mathbf{I}-\mathbf{n}\mathbf{n})\cdot\boldsymbol{\sigma}\cdot\mathbf{n} = -\beta(\mathbf{I}-\mathbf{n}\mathbf{n})\cdot(\mathbf{u}-\mathbf{u}_s), \qquad \mathbf{x}\in\Gamma_s, \tag{2b}$$

$$\mathbf{u}\cdot\mathbf{n} = \mathbf{u}_s\cdot\mathbf{n}, \qquad \mathbf{x}\in\Gamma_s, \tag{2c}$$

$$\cos\theta(\mathbf{x},t) - \cos\theta_s = -\zeta(\mathbf{u}-\mathbf{u}_s)\cdot\boldsymbol{\nu}_s/\gamma, \qquad \mathbf{x}\in\partial\Gamma, \tag{2d}$$

where $\mathbf{I}$ is the identity tensor, $\mathbf{n}$ the outward normal, θ the dynamic contact angle, $\nabla_\Gamma \doteq (\mathbf{I}-\mathbf{n}\mathbf{n})\cdot\nabla$ the surface gradient operator, $\kappa = \nabla_\Gamma\cdot\mathbf{n}$ the mean curvature positively counted

with respect to the normal, $\boldsymbol{\sigma}$ the Cauchy stress tensor, γ the liquid-gas surface tension, $\mathbf{u}_s$ the solid velocity, θ_s the static contact angle. The parameters β and ζ correspond to the Navier-like slip laws that are frequently added both distributed over the solid surface and localized at the triple-contact line, respectively. The contact angle θ appearing in (2d) is defined by

$$\cos\theta = \boldsymbol{\nu} \cdot \boldsymbol{\nu}_s, \tag{3}$$

where $\boldsymbol{\nu}$ and $\boldsymbol{\nu}_s$ are the liquid-gas and solid-liquid surface conormals, respectively (see Fig. 1). The location of the fluid interfaces is an unknown of the problem, since they are assumed to preserve their material identity. In mathematical terms, this means that

$$\mathbf{x} \in \Gamma(t=0) \Leftrightarrow \varphi(\mathbf{x}, t) \in \Gamma(t), \tag{4}$$

where $\varphi(\mathbf{x}, t)$ describes the trajectory of the fluid particle that is at position $\mathbf{x}$ at time $t = 0$. To close the system, a compatible initial velocity field $\mathbf{u}_0(\mathbf{x})$ and an initial configuration $\Gamma(t = 0)$ are given.

3. The ALE formulation: A presentation without moving frames

The domain $\Omega(t)$ over which the Navier-Stokes equations are assumed to hold is allowed to vary with time, so that the variational formulation involves function spaces that depend parametrically on time:

$$W(t) \doteq \left\{ \mathbf{w} \in \left(H^1(\Omega(t)) \right)^d \right\}, \tag{5a}$$

$$W_0(t) \doteq \{ \mathbf{w} \in W(t) \,|\, \mathbf{w} \cdot \mathbf{n} = 0 \text{ on } \Gamma_s \}, \tag{5b}$$

$$W_u(t) \doteq \{ \mathbf{w} \in W(t) \,|\, \mathbf{w} \cdot \mathbf{n} = \mathbf{u}_s \cdot \mathbf{n} \text{ on } \Gamma_s \}, \tag{5c}$$

$$Q(t) \doteq L^2(\Omega(t)) \quad \left(\text{or } L^2(\Omega(t))/\mathbb{R} \text{ if } \Gamma = \emptyset \right), \tag{5d}$$

We will however often drop the time dependence in $\Omega(t)$, $W_0(t)$, etc., from the notation, since no confusion can arise.

The variational formulation of (1) together with (2) then imposes, for all times $t > 0$, that

$$\mathbf{u}(\cdot, t) \in W_u, \qquad p(\cdot, t) \in Q,$$

together with $\mathbf{u}(\mathbf{x}, t = 0) = \mathbf{u}_0(\mathbf{x})$ and

$$\int_\Omega (\rho\,(\partial_t \mathbf{u} + (\mathbf{u}\cdot\nabla)\mathbf{u})\cdot\mathbf{w} + (2\mu D\mathbf{u} - p\mathbf{I}) : D\mathbf{w}] + \int_{\Gamma_s} \beta(\mathbf{u}-\mathbf{u}_s)\cdot\mathbf{w} +$$
$$+ \int_{\partial\Gamma} \zeta((\mathbf{u}-\mathbf{u}_s)\cdot\boldsymbol{\nu}_s)(\mathbf{w}\cdot\boldsymbol{\nu}_s) = \int_\Omega \rho\boldsymbol{g}\cdot\mathbf{w} - \int_\Gamma \gamma\nabla_\Gamma\cdot\mathbf{w} + \int_{\partial\Gamma} \gamma\cos\theta_s \mathbf{w}\cdot\boldsymbol{\nu}_s, \tag{6a}$$
$$\int_\Omega q\nabla\cdot\mathbf{u} = 0, \tag{6b}$$

which must hold for all $(\mathbf{w}, q) \in W_0 \times Q$.

A detailed justification of the capillary terms in this variational system can be found in [21]. The fully discrete formulation to be discussed here is obtained after first discretizing in space by finite elements, which turns (6a)-(6b) into a differential-algebraic equation system, and then discretizing in time by (variants of) classical methods.

Let $\mathcal{T}_h(t)$ be a mesh partition of the domain $\Omega(t)$ which defines a discrete domain $\Omega_h(t)$. Herein, the velocity and pressure approximations are written, respectively, as

$$\mathbf{u}_h(\mathbf{x}, t) = \sum_j \phi^j(\mathbf{x}, t)\mathbf{u}^j(t), \qquad \phi^j(\cdot, t)\mathbf{e}_\alpha \in W_h(t), \tag{7a}$$
$$p_h(\mathbf{x}, t) = \sum_k \psi^k(\mathbf{x}, t)p^k(t), \qquad \psi^k(\cdot, t) \in Q_h(t), \tag{7b}$$

where $\mathbf{e}_\alpha$, $\alpha = 1, ..., d$, are $\mathbb{R}^d$ canonical base's vectors and $\mathbf{u}^j$ and p^k are variables coefficients with nodal value meaning. Above, $W_h(t)$ and $Q_h(t)$ are the finite element subspaces of $W(t)$ and $Q(t)$, respectively, whose bases are polynomials in each simplex $K \in \Omega_h$ and are chosen to be nodal interpolation functions, for simplicity.

The partition moves with time, in particular each nodal position is a function of time that is denoted by $\mathbf{x}^j(t)$, for $t \geq 0$. This defines nodal velocities

$$\mathbf{v}^j(t) \doteq \frac{d\mathbf{x}^j}{dt}(t). \tag{8}$$

Standard nodal finite element discretization spaces are built at each time, given the instantaneous nodal positions. At any fixed instant t, the geometry of the element K is the image of the "master" or "reference" element $\hat{K}$ through the mapping

$$\mathbf{x}|_K = \Phi_K(\mathbf{y}, t) = \sum_{\alpha=1}^{m} \mathcal{N}^\alpha(\mathbf{y})\,\mathbf{x}^{j(\alpha)}(t), \quad \mathbf{y} \in \hat{K}. \tag{9}$$

where the sum extends over the m nodes of K, $\{\mathcal{N}^\alpha\}$ are the shape functions (on the master element) adopted for geometrical interpolation, and $j(\alpha)$ is the number in the mesh of node number α of the master element. Differentiating in time and using (8),

$$\partial_t \Phi_K(\mathbf{y}, t) = \sum_\alpha \mathcal{N}^\alpha(\mathbf{y})\, \mathbf{v}^{j(\alpha)}(t), \doteq \mathbf{v}_h(\mathbf{x}, t) \tag{10}$$

where we have denoted the velocity field induced by the nodal velocities $\{\mathbf{v}^j\}$ as $\mathbf{v}_h$. Note that the interpolants of $\mathbf{v}_h$ correspond to the geometry interpolation functions.

The basis functions for velocity components, always at instant t, are functions $\phi^j(\mathbf{x}, t)$ which are built as the image of shape functions $\{\mathcal{M}^\alpha\}$ defined on $\hat{K}$, that is,

$$\phi^{j(\alpha)}(\Phi_K(\mathbf{y}, t), t) = \mathcal{M}^\alpha(\mathbf{y}). \tag{11}$$

When the mesh is moving the basis functions defined in this way have the important property

$$\partial_t \phi^j(\mathbf{x}, t) \; + \; \mathbf{v}_h(\mathbf{x}, t) \cdot \nabla \phi^j(\mathbf{x}, t) = 0. \tag{12}$$

Expanding now

$$\partial_t \mathbf{u}_h(\mathbf{x}, t) = \partial_t \left(\sum_j \phi^j(\mathbf{x}, t) \mathbf{u}^j(t) \right) = \sum_j \left\{ \phi^j \, \frac{d\mathbf{u}^j}{dt} + \mathbf{u}^j \partial_t \phi^j \right\},$$

one gets, from (12),

$$\partial_t \mathbf{u}_h = \sum_j \left\{ \phi^j \, \frac{d\mathbf{u}^j}{dt} - \mathbf{u}^j \, (\mathbf{v}_h \cdot \nabla \, \phi^j) \right\} = \sum_j \phi^j \, \frac{d\mathbf{u}^j}{dt} - \; (\mathbf{v}_h \cdot \nabla) \mathbf{u}_h. \tag{13}$$

Inserting (13) into an otherwise classical spatial discretization of (6a)-(6b) at time t, and combining with (8) and (26) yields the Differential Algebraic Equation (DAE) system

$$\mathbf{X}'(t) = \mathbf{V}(t), \qquad \text{(by (8))} \tag{14a}$$

$$M(\mathbf{X})\, \mathbf{U}'(t) = A(\mathbf{X}, \mathbf{V}, \mathbf{U})\, \mathbf{U} + G(\mathbf{X})\, \mathbf{P} + F(\mathbf{X}, t) + \\ + S_1(\mathbf{X}, \mathbf{V}, \mathbf{U}, \mathbf{U}', \mathbf{P}), \qquad \text{(momentum)} \tag{14b}$$

$$0 = D(\mathbf{X})\, \mathbf{U} + S_2(\mathbf{X}, \mathbf{V}, \mathbf{U}, \mathbf{U}', \mathbf{P}), \qquad \text{(incompressibility)} \tag{14c}$$

where $\mathbf{U}(t) = \{\mathbf{u}^i\}$, $\mathbf{P}(t) = \{p^i\}$, $\mathbf{X}(t) = \{\mathbf{x}^i\}$, $\mathbf{V}(t) = \{\mathbf{v}^i\}$ and M, G, D, A, and F are defined by the block matrices

$$M_{ij} \doteq \int_\Omega \rho\, \mathbf{I}\, \phi^i \phi^j, \qquad G_{ij} \doteq \int_\Omega \psi^j \nabla \phi^i, \qquad D \doteq G^T, \tag{15a}$$

$$\begin{aligned} A_{ij} \doteq & -\int_\Omega \Big[\rho\, \mathbf{I}\, \phi^i (\mathbf{u}_h - \mathbf{v}_h) \cdot \nabla \phi^j + \mu(\mathbf{I} \nabla \phi^i \cdot \nabla \phi^j + \nabla \phi^j \otimes \nabla \phi^i)\Big] - \\ & - \int_{\Gamma_s} \beta\, \mathbf{I}\, \phi^i \phi^j - \int_{\partial\Gamma} \zeta\, \boldsymbol{\nu_s} \otimes \boldsymbol{\nu_s}\, \phi^i \phi^j, \end{aligned} \tag{15b}$$

$$F_i \doteq \int_\Omega \rho \boldsymbol{g} \phi^i - \int_\Gamma \gamma \nabla_\Gamma \phi^i + \int_{\partial\Gamma} \gamma \cos\theta_s \boldsymbol{\nu}_s \phi^i. \tag{15c}$$

and L, S_1 and S_2 will defined later on. Notice that:

- In this notation, each element of a vector is a column matrix with d components, while each element of a matrix is itself a $d \times d$-submatrix.

- The convection velocity in matrix A is $\mathbf{u}_h - \mathbf{v}_h$, where $\mathbf{v}_h$ is the interpolant of the nodal velocities $\{\mathbf{v}^j\}$. In the framework adopted in this article, it does not arise from the discretization of an "ALE differential formulation" (see, e.g. [22]) but from the straightforward calculation of $\partial_t \mathbf{u}_h$.

- The first term on the right-hand side of (13) is the interpolant of time derivatives of the nodal values of $\mathbf{u}_h$, which is different from $\partial_t \mathbf{u}_h$ because the basis functions depend on time. In the ALE literature, this term is usually called "ALE derivative of $\mathbf{u}_h$", denoted by $\partial_t^* \mathbf{u}_h$, and obtained from a moving-frame formulation of the continuous problem. Here, we avoid the moving frames and introduce it by definition as

$$\partial_t^* \mathbf{u}_h \doteq \sum_j \phi^j \frac{d\mathbf{u}^j}{dt}, \tag{16}$$

  which is in fact the formula used in actual computations. Notice that $\partial_t^* \mathbf{u}_h$ satisfies

$$\partial_t^* \mathbf{u}_h = \partial_t \mathbf{u}_h + (\mathbf{v}_h \cdot \nabla) \mathbf{u}_h. \tag{17}$$

- The vectors S_1 and S_2 arise from the stabilization terms. We consider algebraic subgrid-scale stabilization [23], which for linear elements coincides with the Galerkin

Least Squares formulation [24, 25]. It requires the computation of terms of the form

$$\int_K \mathbf{R}(\mathbf{u}_h) \cdot \mathbf{L}(\mathbf{w}_h),$$

where $\mathbf{R}(\mathbf{u}_h)$ stands for the residual of the momentum equation

$$\mathbf{R}(\mathbf{u}_h) \doteq \rho \left[\partial_t \mathbf{u}_h + (\mathbf{u}_h \cdot \nabla)\mathbf{u}_h\right] - \rho \boldsymbol{g}, \tag{18}$$

and

$$\mathbf{L}(\mathbf{w}_h) \doteq \rho \left[\partial_t \mathbf{w}_h + (\mathbf{u}_h \cdot \nabla)\mathbf{w}_h\right], \tag{19}$$

but note that neither $\partial_t \mathbf{u}_h$ nor $\partial_t \mathbf{w}_h$ are readily available at the integration points. Instead, one takes advantage of (17) to compute $\mathbf{R}(\mathbf{u}_h)$ and $\mathbf{L}(\mathbf{w}_h)$ as

$$\mathbf{R}(\mathbf{u}_h) \doteq \rho \left(\partial_t^* \mathbf{u}_h + [(\mathbf{u}_h - \mathbf{v}_h) \cdot \nabla]\mathbf{u}_h\right) - \rho \boldsymbol{g}, \tag{20}$$

$$\mathbf{L}(\mathbf{w}_h) \doteq \rho \underbrace{\partial_t^* \mathbf{w}_h}_{=0} + [(\mathbf{u}_h - \mathbf{v}_h) \cdot \nabla]\mathbf{w}_h, \tag{21}$$

where now all quantities are straightforward to evaluate at any integration point. In particular $\partial_t^* \mathbf{w}_h$ is zero because of (12). The stabilization terms thus take the form

$$(S_1)_i = \sum_K \left\{ \tau_K \int_K \rho \left(\mathbf{u}_h - \mathbf{v}_h\right) \cdot \nabla \phi^i \, \mathbf{R}(\mathbf{u}_h) \; + \; \delta_K \int_K \nabla \cdot \mathbf{u}_h \, \nabla \phi^i \right\}, \tag{22}$$

$$(S_2)_i = \sum_K \tau_K \int_K \mathbf{R}(\mathbf{u}_h) \cdot \nabla \psi^i, \tag{23}$$

where

$$\tau_K = \left[4 \frac{\mu}{\rho h_K^2} + 2 \frac{\|\mathbf{u}_h - \mathbf{v}_h\|_{L^2(K)}}{h_K} \right]^{-1}, \qquad \delta_K = 4\mu + 2\rho \|\mathbf{u}_h - \mathbf{v}_h\|_{L^2(K)} h_K, \tag{24}$$

These terms will be omitted in the following, since they are only activated in the simulations with equal-order elements and their time discretization is analogous to that of the other terms.

- If the mesh velocity $\mathbf{V}$ is a datum, Equations (14a)-(14c) constitute a closed system for the unknowns $\mathbf{X}(t)$, $\mathbf{U}(t)$ and $\mathbf{P}(t)$. In general, however, $\mathbf{V}$ is not known and depends on $\mathbf{U}$ through kinematic constraints. This introduces an additional equation in the DAE system as discussed in the next section.

4. Mesh velocity equation and final DAE system

In capillary problems the mesh velocity is some artificially constructed field, resulting from some mesh update algorithm. The only condition on $\mathbf{V}(t)$ (consequently on $\mathbf{v}_h$) for the material identity condition (4) to be fulfilled at discrete level is that

$$(\mathbf{u}_h - \mathbf{v}_h) \cdot \mathbf{n}_h = 0, \qquad \text{weakly on } \ \partial\Omega_h(t), \tag{25}$$

for some discrete normal $\mathbf{n}_h$. In general, the operation of determining $\mathbf{v}_h$ from $\mathbf{u}_h$ can be written as an additional equation

$$\mathcal{L}\,\mathbf{v}_h = \mathcal{B}\,\mathbf{u}_h, \tag{26}$$

The specific strategy adopted for this study corresponds to solving the elasticity operator

$$\nabla \cdot \sigma_e = 0, \quad \text{in}\,\Omega_h(t), \qquad \text{with} \qquad \mathbf{v}_h(\mathbf{x}^i, t) = \mathbf{u}_h(\mathbf{x}^i, t), \quad \mathbf{x}^i \in \partial\Omega_h(t), \tag{27}$$

where

$$\sigma_e = \lambda_e(\nabla \cdot \mathbf{v}_h)\,\mathbf{I} + 2\mu_e D\mathbf{v}_h, \tag{28}$$

and D is the symmetric gradient operator already defined. Note that the kinematic condition (25) is certainly satisfied, since in fact $\mathbf{v}_h(\mathbf{x}^i, t) = \mathbf{u}_h(\mathbf{x}^i, t)$ is imposed over all components. The Lamé elastic constants λ_e and μ_e are chosen as $\lambda_e = -\mu_e = E$, where E is element-wise constant and equal to the inverse of the element volume [26]. This choice has worked very well for us, both in 2D and 3D, which is remarkable given that μ_e and $\lambda_e + \frac{2}{3}\mu_e$ differ in sign. The system of equations (26) can be represented in a matricial form

$$L(\mathbf{X})\,\mathbf{V}(t) = B\,\mathbf{U}(t), \tag{29}$$

where

$$L_{ij} \doteq \int_\Omega \lambda_e \nabla\phi^i (\nabla\phi^j)^T + \mu_e(\mathbf{I}\,\nabla\phi^i \cdot \nabla\phi^j + \nabla\phi^j(\nabla\phi^i)^T),$$

and B is a matrix which has non-zero entries only on the lines corresponding to the boundary nodes, and does not depend on unknowns.

The final DAE system, omitting stabilization for clarity, thus reads:

$$\mathbf{X}'(t) = \mathbf{V}(t), \tag{30a}$$

$$0 = L(\mathbf{X})\,\mathbf{V}(t) - B\,\mathbf{U}(t), \tag{30b}$$

$$M(\mathbf{X})\,\mathbf{U}'(t) = A(\mathbf{X}, \mathbf{V}, \mathbf{U})\,\mathbf{U} + G(\mathbf{X})\,\mathbf{P} + F(\mathbf{X}, t), \tag{30c}$$

$$0 = D(\mathbf{X})\,\mathbf{U}(t). \tag{30d}$$

The following comments are in order:

- The actual matrices and vectors in (30a)-(30d) depend on the finite element spaces chosen for the variables. In our implementation, the following velocity/pressure combinations are available: P_2/P_1 (Taylor-Hood), P_1^+/P_1 (minielement) and P_1/P_1 (equal-order element with residual-based stabilization). The geometrical interpolation (and thus the interpolation of $\mathbf{v}_h$) is P_2 for the P_2/P_1 element, and P_1 for the other two.

- System (30a)-(30d) is a DAE of index 2 [19, 20]. In the case of negligible inertia we have $M = 0$ and thus Equation (30c) turns into an algebraic constraint. The index of the DAE then drops to 1.

- Straightforward application of standard discretization techniques for stiff DAEs couples the four unknowns and yields a huge nonlinear algebraic system at each time step. Notice that $\mathbf{X}$ defines the mesh geometry and is thus involved in all integrals appearing in matrices M, A, G, D and L. For this reason, we discuss here time discretization techniques which decouple Eqs. (30a)-(30b) from Eqs. (30c)-(30d).

5. Discretization in time

5.1. Notations

Throughout the paper, we will use a notation based on that used by Codina et al. [27]. Let f^n be the approximation of a function f at time level $t^n = n\delta t$, where the time step

$\delta t \doteq t^{n+1} - t^n$ is assumed constant. For a parameter $\theta \in [0,1]$, we denote

$$\begin{aligned} f^{n+\theta} &\doteq \theta f^{n+1} + (1-\theta) f^n, \\ \delta f^{n+1} &\equiv \delta^{(1)} f^{n+1} \doteq f^{n+1} - f^n, \\ \delta^{(i+1)} f^{n+1} &\doteq \delta^{(i)} f^{n+1} - \delta^{(i)} f^n, \quad i = 1, 2, 3, ... \end{aligned} \tag{31}$$

We also define the backward difference operators

$$\begin{aligned} D_1 f^{n+1} &= \delta f^{n+1} = f^{n+1} - f^n, \\ D_2 f^{n+1} &= \tfrac{3}{2} f^{n+1} - 2 f^n + \tfrac{1}{2} f^{n-1}, \\ D_3 f^{n+1} &= \tfrac{11}{6} f^{n+1} - 3 f^n + \tfrac{3}{2} f^{n-1} - \tfrac{1}{3} f^{n-2}. \end{aligned} \tag{32}$$

5.2. Problems with given mesh velocity

We first consider problems where the mesh velocity $\mathbf{V}(t)$ is a datum, given by the nodal values of some field $\mathbf{v}(\mathbf{x}, t)$, i.e.,

$$(\mathbf{V}(t))_i = \mathbf{v}(\mathbf{x}^i(t), t).$$

With some abuse of notation, we denote

$$\mathbf{V}(t) = \mathbf{v}(\mathbf{X}(t), t).$$

In this way, the DAE to approximate is (14) in which the nodal positions $\mathbf{X}$ can easily be integrated from Eq. 14a.

A couple of possible methods are described below in which this integration is performed using a four-stage Runge-Kutta technique. The description of each method is accompanied by a simple numerical example, so as to leave the Numerical Experiments section (Section 6) exclusively for the tests in which $\mathbf{V}$ is an unknown of the problem, which are much more interesting.

5.2.1. *Midpoint rule/Runge-Kutta method (MR-RK2)*

Integrating the Navier-Stokes part of (14) by the midpoint rule method, the first method that we present here is:

$$\left\{\begin{array}{rcl} \mathbf{K}^{n,i} & = & \mathbf{v}(\mathbf{X}^n + \Delta t \sum_{j=1}^{s} a_{ij}\mathbf{K}^{n,j},\ t_n + c_i\Delta t), \\ \mathbf{X}^{n+1} & = & \mathbf{X}^n + \Delta t \sum_{i=1}^{s} b_i \mathbf{K}^{n,i}; \end{array}\right. \tag{33a}$$

$$\left\{\begin{array}{rcl} M(\mathbf{X}^{n+1/2})\frac{1}{\delta t}D_1\mathbf{U}^{n+1} & = & A(\mathbf{X}^{n+1/2}, \widetilde{\mathbf{V}}^{n+1/2}, \mathbf{U}^{n+1/2})\ \mathbf{U}^{n+1/2} + \\ & & +G(\mathbf{X}^{n+1/2})\,\mathbf{P}^{n+1/2} + F(\mathbf{X}^{n+1/2}, t^{n+1/2}); \\ 0 & = & D(\mathbf{X}^{n+1/2})\ \mathbf{U}^{n+1/2}. \end{array}\right. \tag{33b}$$

where $\widetilde{\mathbf{V}}^{n+1/2}$ is an approximation of $\mathbf{v}(\mathbf{X}(t^{n+1/2}), t^{n+1/2})$ for which we have chosen

$$\widetilde{\mathbf{V}}^{n+1/2} = \frac{1}{\delta t} D_1 \mathbf{X} = \frac{\mathbf{X}^{n+1} - \mathbf{X}^n}{\delta t}.$$

The coefficients of RK2 are given by the following Butcher tableau:

$$\begin{array}{c|c} c_i & a_{ij} \\ \hline & b_{ij} \end{array} \quad = \quad \begin{array}{c|cc} 0 & & \\ 1/2 & 1/2 & \\ \hline & 0 & 1 \end{array}.$$

To avoid the computation of an initial pressure (which must obey some compatibility conditions, see [28]), $\mathbf{P}^{n+1/2}$ here is not considered as given by definition (31) but as a unknown itself. Therefore, it is necessary a post-processing to compute the approximation of $\mathbf{P}(t_n)$, namely

$$\mathbf{P}(t_n) \simeq \frac{1}{2}\left(\mathbf{P}^{n+1/2} + \mathbf{P}^{n-1/2}\right).$$

As a numerical example, we test the formulation on the following problem referred to as *GMV1* (given-mesh-velocity problem 1). We consider a kind of Couette flow, with $\rho = \mu = 1$, such that the exact solution is given by

$$\mathbf{u} = (x\cos(20t) + y\sin(20t),\ x\sin(20t) - y\cos(20t)), \tag{34}$$

$$p = x\cos(20t) + y\sin(20t), \tag{35}$$

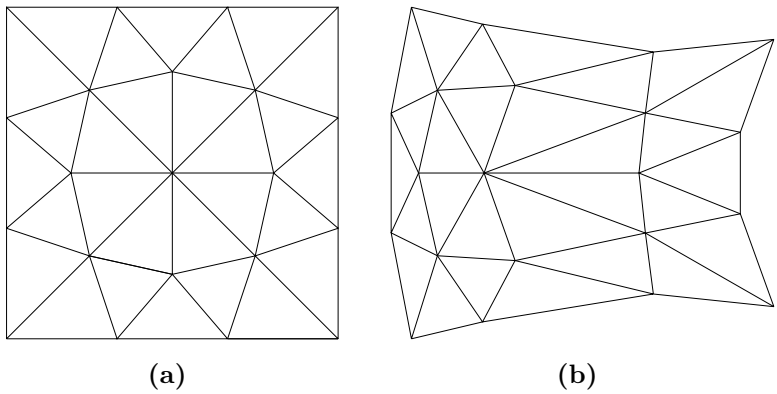

Figure 2: Initial (left) and final (right) meshes of the GMV1 test problem. They correspond to $t = 0$ and $t = 0.2$, respectively.

where $(x, y) \doteq \mathbf{x}$. Note that $\mathbf{u}$ and p are linear in $\mathbf{x}$, so the only source of errors is the time discretization. The prescribed motion of the nodes is

$$\mathbf{v} = 5 \begin{pmatrix} \sin^2(x)\sin(10t) + \sin^2(y)\cos(10t) \\ \sin(x)\sin(y)(\cos(10t) - \sin(10t)) \end{pmatrix}, \tag{36}$$

the initial domain is the square $[-1.2, 1.2]^2$ and the initial mesh is shown in Fig. 2. We have taken the time steps $\delta t = 0.01 \cdot 2^{-k}$, $k = 0, ..., 8$, and we have computed the velocity error E_u and the pressure error E_p as

$$E_u = \|\mathbf{u}(\cdot, t_n) - \mathbf{u}_h^n(\cdot)\|_{H_1(\Omega_h(t_n))}, \qquad E_p = \|p(\cdot, t_n) - p_h^n(\cdot)\|_{L^2(\Omega_h(t_n))},$$

at time $t_n = 0.2$. The choices of t_n and $\mathbf{v}$ are such that the distortion of the final mesh is comparable to the edge size of the coarsest mesh. We have tested the three implemented elements, all of them achieving second order convergence for both $\mathbf{u}$ and p.

The exact solution of problem GMV1 belongs to the finite element space at all times, so that there is no error due spatial discretization. In contrast, the problem proposed in [15], which we refer to as *GMV2 problem*, involves both temporal and spatial discretization

errors. Its analytical solution reads:

$$\mathbf{u} = g(t) \begin{pmatrix} -1 + x + x^2 + y + y^2 + xy + y^3 \\ 1 + x + x^2 - y - y^2/2 - 2xy + x^3 \end{pmatrix}, \tag{37a}$$

$$p = x^2, \tag{37b}$$

$$g(t) = 1 + \tanh(t)\sin(9\pi t), \tag{37c}$$

$$\mathbf{v} = (xyt^2/5, \;\; xy(1 - t^2/5)/10). \tag{37d}$$

We have taken time steps $\delta t_k = 0.01 \cdot 2^{-k}$, $k = 0, ..., 8$, with the same mesh as in GMV1. Estimates of the temporal error in velocity ($E_u^{(k)}$) and pressure ($E_p^{(k)}$) are computed as

$$E_u^{(k)} = \|\mathbf{U}(\delta t_k) - \mathbf{U}(\delta t_{k+1})\|_\infty, \tag{38}$$

$$E_p^{(k)} = \|\mathbf{P}(\delta t_k) - \mathbf{P}(\delta t_{k+1})\|_\infty \tag{39}$$

where $||\cdot||_\infty$ is the usual vector maximum norm defined by $||\mathbf{x}||_\infty = \max(|x_1|, ..., |x_n|)$. Notice that this procedure measures convergence to the exact solution of the DAE system (14).

Second-order convergence in $\mathbf{U}$ and $\mathbf{P}$ is obtained both for the stable (P_2/P_1 and P_1^+/P_1) and for the stabilized (P_1/P_1) finite element formulations. These results are better than those obtained in [15], where linear convergence is reported.

5.2.2. BDF2/Runge-Kutta method (BDF2-RK2)

The temporal discretization of the semi-discrete formulation (14) by BDF2 formula is analogous to the midpoint-rule. We will refer to this approach as BDF2-RK2. Integrating the Navier-Stokes part of (14) by the BDF2 formula, we obtain:

$$\left\{ \begin{array}{rcl} \mathbf{K}^{n,i} & = & \mathbf{v}(\mathbf{X}^n + \delta t \sum_{j=1}^{s} a_{ij}\mathbf{K}^{n,j},\; t_n + c_i\delta t), \\ \mathbf{X}^{n+1} & = & \mathbf{X}^n + \delta t \sum_{i=1}^{s} b_i \mathbf{K}^{n,i}; \end{array} \right. \tag{40a}$$

$$\left\{ \begin{array}{rcl} M(\mathbf{X}^{n+1})\frac{1}{\delta t} D_2 \mathbf{U}^{n+1} & = & A(\mathbf{X}^{n+1}, \widetilde{\mathbf{V}}^{n+1}, \mathbf{U}^{n+1})\, \mathbf{U}^{n+1} + \\ & & +G(\mathbf{X}^{n+1})\, \mathbf{P}^{n+1} + F(\mathbf{X}^{n+1}, t^{n+1}); \\ 0 & = & D(\mathbf{X}^{n+1})\, \mathbf{U}^{n+1}, \end{array} \right. \tag{40b}$$

where $\widetilde{\mathbf{V}}^{n+1}$ is an approximation of $\mathbf{v}(\mathbf{X}(t^{n+1}), t^{n+1})$. We have chosen

$$\widetilde{\mathbf{V}}^{n+1} = \frac{1}{\delta t} D_2 \mathbf{X}. \tag{41}$$

We tested this method on problems GMV1 and GMV2. Again, second order convergence is obtained for all finite elements for both tests.

In [13] two versions of BDF2 are considered. One of them is equivalent to the BDF2 presented here, while the other one consists, in our framework, of taking

$$\widetilde{\mathbf{V}}^{n+1} = \frac{1}{\delta t} D_1 \mathbf{X}. \tag{42}$$

instead of (41). As reported in [13], this change degrades the convergence rate to linear. They interpreted these two versions as a linear and a quadratic (in time) interpolation of the domain deformation, respectively. With the DAE formulation it is easier to understand this degradation: the velocity in (42) is a second order approximation of $\mathbf{V}$ at $t^{n+1/2}$, but the momentum equation requires $\mathbf{V}$ to be approximated at t^{n+1}.

5.3. Algorithms for free boundary problems

For real-life problems in which the mesh velocity is unknown, such as the semi-discrete capillary equations (30a)-(30d), the time integration proposed here is somewhat similar to the one performed previously. However, because the nodal positions are now coupled to the fluid velocity, we use an extrapolation technique to decouple Eqs. (30a)-(30b) from Eqs. (30c)-(30d). In more concrete terms, we extrapolate the fluid velocity in the geometrical update so that the Navier-Stokes part can be solved separately.

In the following we introduce some variants of this idea for which the extrapolation does not lead to accuracy loss; i.e., the convergence order (in time) of the fully-discrete algorithm is the same as that of the time discretization adopted for the Navier-Stokes part.

5.3.1. Midpoint rule/Adams-Bashforth method (MR-AB)

This method applied to the semi-discrete capillary equations (30) reads

$$\tfrac{1}{\delta t} D_1 \mathbf{X}^{n+1} = \mathbf{V}^{n+1/2}, \tag{43a}$$

$$0 = L(\widehat{\mathbf{X}}^{n+1/2})\mathbf{V}^{n+1/2} - B\ \ (\widehat{\mathbf{U}}^{n+1/2}), \tag{43b}$$

$$M(\mathbf{X}^{n+1/2})\tfrac{1}{\delta t} D_1 \mathbf{U}^{n+1} = A(\mathbf{X}^{n+1/2}, \mathbf{V}^{n+1/2}, \mathbf{U}^{n+1/2})\ \mathbf{U}^{n+1/2} + \tag{43c}$$

$$+\, G(\mathbf{X}^{n+1/2})\, \mathbf{P}^{n+1/2} + F(\mathbf{X}^{n+1/2}, t^{n+1/2}), \tag{43d}$$

$$0 = D(\mathbf{X}^{n+1/2})\ \mathbf{U}^{n+1/2}, \tag{43e}$$

where $\mathbf{V}^{n+1/2}$ is taken as unknown in much the same way as $\mathbf{P}^{n+1/2}$. Notice that the algorithm starts each time step by solving (43b) with boundary values given by

$$\widehat{\mathbf{U}}^{n+1/2} = \frac{3}{2}\,\mathbf{U}^{n} - \frac{1}{2}\,\mathbf{U}^{n-1}, \tag{44}$$

which is an Adams-Bashforth-like extrapolation of $\mathbf{U}$ to time $n + 1/2$. Though not strictly necessary, the matrix L of the elasticity operator is also built on the extrapolated geometry $\widehat{\mathbf{X}}^{n+1/2}$.

The mesh velocity $\mathbf{V}^{n+1/2}$ is then plugged into (43a) to find $\mathbf{X}^{n+1}$, from which one computes $\mathbf{X}^{n+1/2}$. The mesh geometry determined by $\mathbf{X}^{n+1/2}$ is used for the Navier-Stokes part (43c)-(43d), to find $\mathbf{U}^{n+1}$ and $\mathbf{P}^{n+1/2}$ and proceed to the next time step.

5.3.2. BDF2/extrapolated BDF2 (BDF2-BDF2e)

This method reads

$$\tfrac{1}{\delta t} D_2 \mathbf{X}^{n+1} = \mathbf{V}^{n+1}, \tag{45a}$$

$$0 = L(\widehat{\mathbf{X}}^{n+1})\mathbf{V}^{n+1} - B\ \ (\widehat{\mathbf{U}}^{n+1}), \tag{45b}$$

$$M(\mathbf{X}^{n+1})\tfrac{1}{\delta t} D_2 \mathbf{U}^{n+1} = A(\mathbf{X}^{n+1}, \mathbf{V}^{n+1}, \mathbf{U}^{n+1})\ \mathbf{U}^{n+1} + \tag{45c}$$

$$+\, G(\mathbf{X}^{n+1})\, \mathbf{P}^{n+1} + F(\mathbf{X}^{n+1}, t^{n+1}), \tag{45d}$$

$$0 = D(\mathbf{X}^{n+1})\ \mathbf{U}^{n+1}, \tag{45e}$$

where the extrapolation $\widehat{\mathbf{U}}^{n+1}$ of the (boundary values of) velocity to time $n+1$ is given by

$$\widehat{\mathbf{U}}^{n+1} = 2\,\mathbf{U}^{n} - \mathbf{U}^{n-1}. \tag{46}$$

The same formula is used to compute $\widehat{\mathbf{X}}^{n+1}$. Note that in this method the mesh-velocity unknown is $\mathbf{V}^{n+1}$.

5.3.3. BDF2/Adams-Bashforth method (BDF2-AB)

In the methods introduced so far, the time instant at which $\mathbf{V}$ is computed from the kinematic equations (30a)-(30b) coincides with the instant needed in the momentum equation (30c). This instant is $n + 1/2$ for the MR-AB method and $n + 1$ for the BDF2-BDF2e method.

It is possible, though not advisable, to devise methods in which the instants do not coincide and still preserve the temporal accuracy. This is the case of the BDF2-AB method, which reads

$$\tfrac{1}{\delta t} D_1 \mathbf{X}^{n+1} = \mathbf{V}^{n+1/2}, \tag{47a}$$

$$0 = L(\widehat{\mathbf{X}}^{n+1/2})\mathbf{V}^{n+1/2} - B\ \ (\widehat{\mathbf{U}}^{n+1/2}), \tag{47b}$$

$$M(\mathbf{X}^{n+1})\tfrac{1}{\delta t} D_2 \mathbf{U}^{n+1} = A(\mathbf{X}^{n+1}, \mathbf{V}^{n+1}, \mathbf{U}^{n+1})\ \mathbf{U}^{n+1} + \tag{47c}$$

$$+ G(\mathbf{X}^{n+1})\,\mathbf{P}^{n+1} + F(\mathbf{X}^{n+1}, t^{n+1}), \tag{47d}$$

$$0 = D(\mathbf{X}^{n+1})\ \mathbf{U}^{n+1}, \tag{47e}$$

where $\widehat{\mathbf{U}}^{n+1/2}$ is defined by (44), and similarly for $\widehat{\mathbf{X}}^{n+1/2}$. The $\mathbf{V}$ unknown in this case is $\mathbf{V}^{n+1/2}$ and the vector $\mathbf{V}^{n+1}$ used in (47c) is obtained from

$$\mathbf{V}^{n+1} = 2\mathbf{V}^{n+1/2} - \mathbf{V}^{n}. \tag{48}$$

This method, unlike the others, requires the storage of $\mathbf{V}^{n+1}$ for use at the next time step.

5.3.4. BDF3/extrapolated BDF3 method (BDF3-BDF3e)

The methodology with which the three schemes proposed above are built is naturally extended to higher orders. It consists of selecting three discretization rules: (a) one for the time derivative in (30a), (b) one for the extrapolation of $\mathbf{X}$ and $\mathbf{U}$ in (30b) and (c) one for the time derivative in (30c). All three discretization rules being of the same order m, the overall scheme will be consistent with order m.

As an example, let us present a scheme in which all three rules (a), (b) and (c) are based on the BDF3 differentiation formula and which has indeed third-order accuracy:

$$\tfrac{1}{\delta t} D_3 \mathbf{X}^{n+1} = \mathbf{V}^{n+1}, \tag{49a}$$

$$0 = L(\widehat{\mathbf{X}}^{n+1})\mathbf{V}^{n+1} - B\ \ (\widehat{\mathbf{U}}^{n+1}), \tag{49b}$$

$$M(\mathbf{X}^{n+1})\tfrac{1}{\delta t} D_3 \mathbf{U}^{n+1} = A(\mathbf{X}^{n+1}, \mathbf{V}^{n+1}, \mathbf{U}^{n+1})\ \mathbf{U}^{n+1} + \tag{49c}$$

$$+\, G(\mathbf{X}^{n+1})\, \mathbf{P}^{n+1} + F(\mathbf{X}^{n+1}, t^{n+1}), \tag{49d}$$

$$0 = D(\mathbf{X}^{n+1})\ \mathbf{U}^{n+1}, \tag{49e}$$

where

$$\widehat{\mathbf{U}}^{n+1} = 3\,\mathbf{U}^{n} - 3\,\mathbf{U}^{n-1} + \ \mathbf{U}^{n-2}, \tag{50}$$

and similarly for $\widehat{\mathbf{X}}^{n+1}$.

6. Numerical experiments

6.1. Oscillating droplet

We first test the methods on a three-dimensional oscillating droplet problem. Taking advantage of the azimuthal symmetry, only 1/4 of the drop is discretized. The drop is initially perturbed by 10% of its radius in the fundamental oscillation mode, i.e., its initial radius r in function of polar angle θ is

$$r(\theta) = R_0 \left(1 + 0.1\ \frac{1}{2}(3\cos^2(\theta) - 1)\right), \qquad \theta \in [-\pi/2,\, \pi/2],$$

where R_0 is the radius of the unperturbed drop. Note that this perturbation does not preserve volume.

In this problem the dimensional data are four (R, ρ, μ and γ), with one dimensionless group for which we choose the Ohnesorge number

$$\mathrm{Oh} = \frac{\mu}{\sqrt{\rho\, R\, \gamma}}, \tag{51}$$

where R is the effective radius induced by the volume V_0 of the perturbed drop, namely

$$R = \left(\frac{3V_0}{4\pi}\right)^{(1/3)}.$$

To render the numerical results dimensionless, we adopt the scales

$$\text{Length: } R, \qquad \text{Velocity: } \frac{\gamma}{\mu}, \qquad \text{Time: } \frac{\mu\, R}{\gamma}, \qquad \text{Pressure: } \frac{\gamma}{R},$$

When Oh $\ll 1$ the drop exhibits weakly-damped oscillations with a period and decay time that can be estimated as $\sqrt{\rho\,\pi^2\,R^3/(2\,\gamma)}$ and $\rho R^2/(5\mu)$, respectively (see [29]), which upon adimensionalization results in

$$\tau = \frac{\pi}{\sqrt{2}\ \mathrm{Oh}}, \qquad b = \frac{1}{5\ \mathrm{Oh}^2}.$$

For this study we take $\mathrm{Oh} = 1.95 \times 10^{-2}$, so that $\tau = 113.92$ and $b = 525.97$. We take the time instant $T = 118.8$ (a little more than one period) as final time in the convergence tests. The maximum velocity in the first period, as obtained from the simulation is $3.4\ 10^{-3}$, which corresponds to Reynolds number of $\mathrm{Re} = 8.9$ and indicates significant inertial effects. The radius of the unperturbed drop R was taken based on the volume of the initial mesh (which is obviously equal for all methods). For illustrative purposes, we show in Fig. 3 the second time step configuration of the mesh (1163 P_2/P_1 elements), and in Fig. 4 the time history (converged in δt using the BDF3-BDF3e method) of the vertical position of the uppermost node of the mesh ($\theta = \pi/2$). The numerical period obtained was $\tau_h = 114.71$. The difference with $\tau = 113.92$ is due to spatial discretization errors and to inaccuracy of the theoretical prediction, which is valid just for infinitesimal amplitudes.

We now proceed to describe the numerical experiments performed to assess the temporal accuracy of the proposed methods, and after that include a basic assessment of stability.

6.1.1. Accuracy

We have applied all four methods (MR-AB, BDF2-BDF2e, BDF2-AB and BDF3-BDF3e) to the oscillating droplet problem, with time steps $\delta t = 0.095 \cdot 2^{-k}$, $k = 0, ..., 5$. The methods need one or two time steps to be already computed so as to apply the extrapolation formula. For these first steps no extrapolation is performed, so that the fully coupled nonlinear system for $\mathbf{X}$, $\mathbf{V}$, $\mathbf{U}$ and $\mathbf{P}$ is solved by Picard iteration between the geometrical and dynamical parts of the system (typically 4 to 8 iterations were needed).

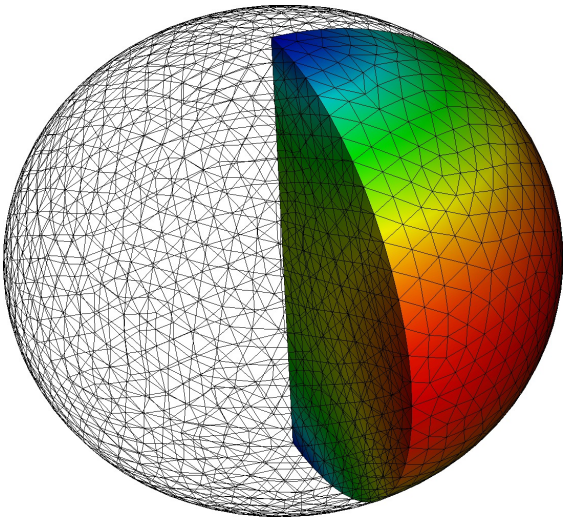

Figure 3: Second time step configuration of the simulation of the oscillating drop. The the color scale represents the pressure (red: 2.18, blue: 1.52). The mesh has 1163 quadratic elements.

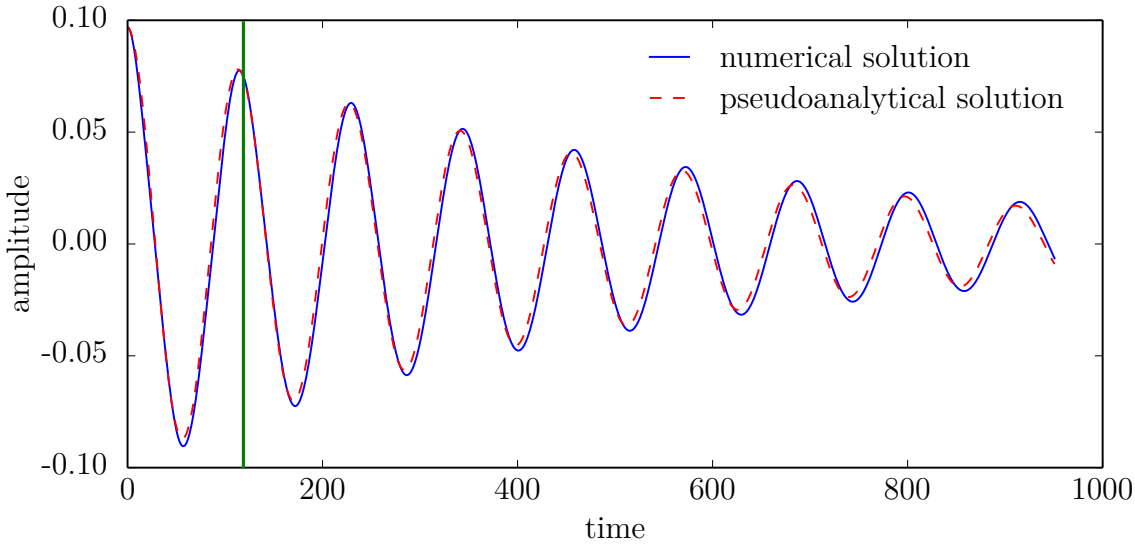

Figure 4: Comparison of the amplitude obtained numerically with the pseudoanalytically solution $A(t) = R\cos(\Omega t)\exp(-t/b)$ where $\Omega = 2\pi/\tau$, $\tau = 113.92$, $b = 525.97$ and $R = 0.097$ (effective radius). The vertical line corresponds to the final instant $T = 118.8$ where errors are computed in the convergence tests.

All three finite element discretizations were used (P_2/P_1, P_1^+/P_1 and P_1/P_1(stab.)). The temporal accuracy did not show any dependence on the element type, so that just the P_2/P_1 results will be detailed here.

The errors for both velocity and pressure are computed as in (38). Tables 1 and 2 present the results. The methods achieve their expected convergence order in all variables and with all methods, either Galerkin or stabilized, either quadratic or linear in space. Among the second order methods the BDF2-AB seems to be the more accurate, the BDF2-BDF2e being the least. Since the estimates $E_u^{(k)}$ and $E_p^{(k)}$ are not true errors, these results are not conclusive and may even be case dependent.

6.1.2. Stability

The methods proposed in this article are all extrapolatory and thus stability restrictions in the time step are to be expected. The most popular algorithm, namely the basic "staggered scheme" in which velocity and pressure are updated with fixed geometry and then geometry is updated according to the last computed velocity (used in, e.g., [17]), has a stability restriction which reads ([30])

$$\delta t \;<\; \delta t_{\rm lim} \;\simeq\; \frac{1}{2}\left\{ C_2 \frac{\mu\, h}{\gamma} + \sqrt{\left(C_2 \frac{\mu\, h}{\gamma}\right)^2 + 4 C_1 \frac{\rho\, h^3}{\gamma}} \right\}, \tag{52}$$

where h is the space step of the discretization, and C_1 and C_2 do not depend on the physical and discretization data of the problem.

To assess whether the second and third order schemes proposed here bring with them tolerable (or intolerable) reductions in $\delta t_{\rm lim}$, we present here experiments in which the maximum time step that yields a well-behaved solution is computed for all methods and all finite element spaces, and compared to that of the following basic (first order) staggered scheme:

$$\tfrac{1}{\delta t} D_1 \mathbf{X}^{n+1} = \mathbf{V}^n, \tag{53a}$$

$$0 = L(\mathbf{X}^n)\mathbf{V}^n - B\;\;(\mathbf{U}^n), \tag{53b}$$

$$M(\mathbf{X}^{n+1}) \tfrac{1}{\delta t} D_1 \mathbf{U}^{n+1} = A(\mathbf{X}^{n+1}, \mathbf{V}^n, \mathbf{U}^{n+1})\; \mathbf{U}^{n+1} + \tag{53c}$$

$$+\, G(\mathbf{X}^{n+1})\, \mathbf{P}^{n+1} + F(\mathbf{X}^{n+1}, t^{n+1}), \tag{53d}$$

$$0 = D(\mathbf{X}^{n+1})\; \mathbf{U}^{n+1}. \tag{53e}$$

The resulting values of $\delta t_{\rm lim}$ are displayed in Table 3. One observes that none of the methods looses stability significantly with respect to the basic staggered one. The one that

allows for the larger time steps is the BDF2-BDF2e, at least 25% higher than those allowed by the basic scheme. Remarkably, the third order scheme also allows for larger time steps than the basic one.

The element P_2/P_1 exhibits a much more restrictive stability limit than those of the linear elements, which is natural because quadratic elements have more degrees of freedom at the free boundary.

k	MR-AB	order	BDF2-BDF2e	order	BDF2-AB	order	BDF3-BDF3e	order
0	1.358e-07	–	1.335e-07	–	2.980e-08	–	3.402e-09	–
1	3.378e-08	2.008	3.296e-08	2.018	7.444e-09	2.001	4.216e-10	3.012
2	8.420e-09	2.004	8.198e-09	2.007	1.859e-09	2.001	5.188e-11	3.022
3	2.122e-09	1.987	2.324e-09	1.818	5.104e-10	1.865	6.362e-12	3.027
4	5.104e-10	2.056	5.882e-10	1.982	1.124e-10	2.182	6.808e-13	3.224

Table 1: Velocity errors $E_u^{(k)}$ and experimental orders of accuracy (computed as $\log_2(E_u^{(k)}/E_u^{(k+1)})$).

k	MR-AB	order	BDF2-BDF2e	order	BDF2-AB	order	BDF3-BDF3e	order
0	1.680e-05	–	1.537e-05	–	3.762e-06	–	1.665e-07	–
1	4.208e-06	1.997	3.857e-06	1.994	9.416e-07	2.001	2.029e-08	3.037
2	1.052e-06	1.999	9.660e-07	1.997	2.355e-07	2.001	2.477e-09	3.034
3	2.704e-07	1.960	2.542e-07	1.926	6.044e-08	1.865	3.114e-10	2.991
4	6.044e-08	2.161	6.518e-08	1.963	6.524e-09	2.182	4.019e-11	2.954

Table 2: Pressure errors $E_p^{(k)}$ and experimental orders of accuracy (computed as $\log_2(E_p^{(k)}/E_p^{(k+1)})$).

Element	MR-AB	BDF2-BDF2e	BDF2-AB	BDF3-BDF3e	Basic
P_2/P_1	0.16	0.29	0.25	0.22	0.19
P_1^+/P_1	0.80	1.04	0.99	0.85	0.76
$P_1/P_1(stab)$	1.09	1.28	1.18	1.18	0.85

Table 3: Maximum time step for which the algorithms are stable ($\delta t_{\rm lim}$ in the text).

6.2. Sliding drop

We now consider a drop sliding down a vertical plane. In this test all capillary terms in (6) are active: surface tension, surface dissipation and concentrated contact line forces. Clearly, all of them depend on the geometry.

Some complex phenomena like hysteresis, formation of cusps and bifurcations at the trailing edge (see [31] for illustrations) can be simulated with a proper modeling of the dynamic contact angle. Popular examples of models that can be incorporated in our formulation by finding an explicit expression for ζ (and possibly for β) are Jiang's model [32], Cox's model [33], de Gennes' model [34], Bracke's model [35], among others.

The drop is initially a hemisphere of radius R_0. Unlike the previous numerical example, the dimensionless parameters are compute with the radius $R = R_0$. The problem is symmetric about the y-z plane so that just one half of the drop is discretized with a mesh of 1696 P_2/P_1 elements.

The non-dimensional parameters that govern this problem are the Ohnesorge number (Oh) and the static contact angle θ_s, together with

$$\Pi_1 = \frac{\rho\, R^2\, g}{\gamma}, \qquad \Pi_2 = \frac{\beta\, R}{\mu}, \qquad \text{and} \qquad \Pi_3 = \frac{\zeta}{\mu}, \tag{54}$$

which are non-dimensional measures of the gravity acceleration g, the Navier friction parameter β and the line dissipation parameter ζ, respectively.

We have taken Oh$=0.707$ in our experiment, so that viscous and surface tension forces are roughly of the same order. Also, we have taken $\theta_s = \pi/2$ so that, when $\beta = \zeta = 0$ the problem reduces to that of a spherical drop falling in vacuum (this simple fact was used as

k	$E_u^{(k)}$	order	$E_p^{(k)}$	order	E_{vol}	order
1	9.375e-06	–	5.640e-05	–	8.611e-06	2.027
2	2.092e-06	2.164	1.284e-05	2.135	2.135e-06	2.011
3	5.190e-07	2.010	3.033e-06	2.082	5.310e-07	2.007
4	1.279e-07	2.021	7.225e-07	2.069	1.323e-07	2.004
5	3.123e-08	2.033	1.798e-07	2.006	3.302e-08	2.002
6	7.595e-09	2.039	4.459e-08	2.011	8.248e-09	2.001

Table 4: Three dimensional sliding drop time convergence using the MR-AB method.

additional validation of the code). It was observed that the mesh deformation increased as the parameters Π_1, Π_2 and Π_3 were increased. The simulation reported here corresponds to $\Pi_1 = 1$, $\Pi_2 = 5$ and $\Pi_3 = 10$. These values are close to the maximum ones for which the problem can be successfully modeled without remeshing the domain along the way.

We have taken the (dimensionless) time steps $\delta t = 0.02 \cdot 2^{-k}$, $k = 0, 1, \ldots, 6$ and the errors have been computed at $T = 4$. In addition to errors $E_u^{(k)}$ and $E_p^{(k)}$ we also report the volume error defined as

$$E_{vol} = \frac{|\text{Volume}_{final} - \text{Volume}_{initial}|}{\text{Volume}_{initial}}. \tag{55}$$

An illustration of the simulated drop can be found in Fig. (5). The maximum velocity obtained was 0.26 which corresponds to Reynolds number of Re = 0.52. The difference between the advancing contact angle from the receding one increases over time due to the presence of the contact line dissipation term. In Fig. 6 one can see that the displacement of the advancing contact point starts to increase faster than the receding contact point at $t = 1$, so that the drop slowly stretches. The slip condition on the wall surface makes that $\mathbf{u}_h$ is not zero there. Figure 7 shows the velocity at the fluid-solid contact. Initially the interior of the disk moves faster than the triple line, later on the maximum velocities take place at the triple line. There is a pattern of convergence on the upper side and of divergence on the lower one. This is in agreement with experimental results that show narrow-tailed sliding

drops [33].

Table 4 shows the temporal convergence results of method MR-AB when using the P_2/P_1 element. One can note that quadratic convergence is maintained for velocity, pressure and volume. Optimal-order behavior is also observed for the other methods and finite element spaces.

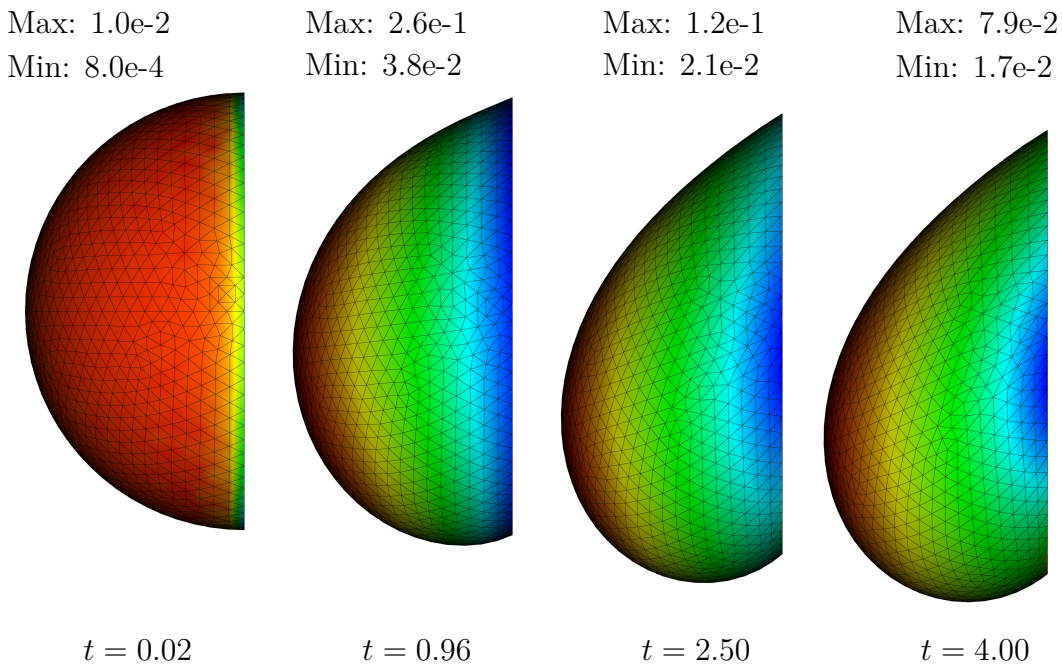

Figure 5: Some frames of the simulation of the sliding droplet. The color corresponds to the velocity magnitude where blue color is the minimum and the red is the maximum value.

7. Conclusions

We have presented a Differential-Algebraic formulation of arbitrary Lagrangian-Eulerian techniques for finite element formulations of free boundary problems.

We introduced a discretization strategy for the resulting DAE system which allowed us to build several second and third order (in time) fully discrete methods. These methods, which decouple the geometrical and dynamical variables by means of extrapolation procedures, were assessed on capillary flows involving surface tension and triple-contact lines. The spatial discretization was carried out with both div-stable (P_2/P_1, P_1^+/P_1) and equal-order

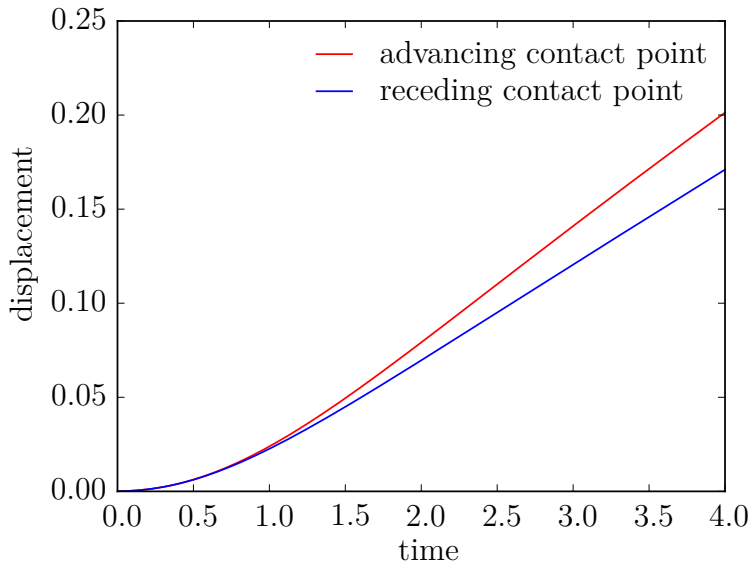

Figure 6: Comparison between the displacement of the advancing contact point and the receding contact point of a drop sliding over a vertical wall.

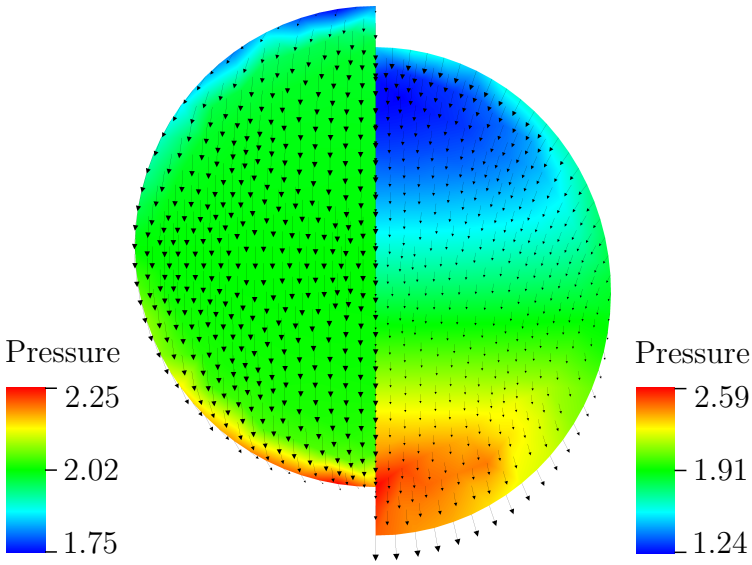

Figure 7: Profiles of the fluid-solid interface of the sliding drop at $t = 0.02$ (left) and $t = 4$ (right). The vector field corresponds to the fluid velocity which maximum magnitude value is 7.47e-3 in the left and 5.25e-2 in the right.

(P_1/P_1, stabilized) finite elements. The results show that the design accuracy is achieved by all methods, and that the proposed extrapolation procedures do not have a significant

negative impact on the stability.

Acknowledgments

The authors gratefully acknowledge the financial support received from FAPESP (Grants. no. 2012/14481-8 and 2012/04560-8) and from the Brazilian National Research and Technology Council (CNPq).

References

[1] M. Walkley, P. Gaskell, P. Jimack, M. Kelmanson, J. Summers, Finite element simulation of three-dimensional free-surface flow problems with dynamic contact lines, International journal for numerical methods in fluids 47 (10-11) (2005) 1353–1359.

[2] J.-F. Gerbeau, T. Lelievre, Generalized navier boundary condition and geometric conservation law for surface tension, Computer Methods in Applied Mechanics and Engineering 198 (5) (2009) 644–656.

[3] J. Sprittles, Y. Shikhmurzaev, Finite element framework for describing dynamic wetting phenomena, International Journal for Numerical Methods in Fluids 68 (10) (2012) 1257–1298.

[4] S. Rabier, M. Medale, Computation of free surface flows with a projection fem in a moving mesh framework, Computer methods in applied mechanics and engineering 192 (41) (2003) 4703–4721.

[5] R. J. Leveque, Z. Li, The immersed interface method for elliptic equations with discontinuous coefficients and singular sources, SIAM Journal on Numerical Analysis 31 (4) (1994) 1019–1044.

[6] P. Geuzaine, C. Grandmont, C. Farhat, Design and analysis of ale schemes with provable second-order time-accuracy for inviscid and viscous flow simulations, Journal of Computational Physics 191 (1) (2003) 206–227.

[7] C. Farhat, P. Geuzaine, Design and analysis of robust ale time-integrators for the solution of unsteady flow problems on moving grids, Computer Methods in Applied Mechanics and Engineering 193 (39) (2004) 4073–4095.

[8] C. Farhat, A. Rallu, K. Wang, T. Belytschko, Robust and provably second-order explicit–explicit and implicit–explicit staggered time-integrators for highly non-linear compressible fluid–structure interaction problems, International Journal for Numerical Methods in Engineering 84 (1) (2010) 73–107.

[9] P. Thomas, C. Lombard, Geometric conservation law and its application to flow computations on moving grids, AIAA journal 17 (10) (1979) 1030–1037.

[10] M. Lesoinne, C. Farhat, Geometric conservation laws for flow problems with moving boundaries and deformable meshes, and their impact on aeroelastic computations, Computer methods in applied mechanics and engineering 134 (1) (1996) 71–90.

[11] H. Guillard, C. Farhat, On the significance of the geometric conservation law for flow computations on moving meshes, Computer Methods in Applied Mechanics and Engineering 190 (11) (2000) 1467–1482.

[12] C. Farhat, P. Geuzaine, C. Grandmont, The discrete geometric conservation law and the nonlinear stability of ale schemes for the solution of flow problems on moving grids, Journal of Computational Physics 174 (2) (2001) 669–694.

[13] L. Formaggia, F. Nobile, Stability analysis of second-order time accurate schemes for ale–fem, Computer methods in applied mechanics and engineering 193 (39) (2004) 4097–4116.

[14] F. Nobile, Numerical approximation of fluid-structure interaction problems with application to haemo-

dynamics, Ph.D. thesis, Politecnico di Milano (2001).

[15] S. Étienne, A. Garon, D. Pelletier, Perspective on the geometric conservation law and finite element methods for ale simulations of incompressible flow, Journal of Computational Physics 228 (7) (2009) 2313–2333.

[16] J. Liu, Simple and efficient ale methods with provable temporal accuracy up to fifth order for the stokes equations on time varying domains, SIAM Journal on Numerical Analysis 51 (2) (2013) 743–772.

[17] S. Ganesan, L. Tobiska, Modelling and simulation of moving contact line problems with wetting effects, Computing and visualization in science 12 (7) (2009) 329–336.

[18] E. BANSCH, S. WELLER, A comparison of several time discretization methods for free surface flows, in: Proceedings of ALGORITMY, 2012, pp. 331–341.

[19] E. Hairer, G. Wanner, Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems, 2nd Edition, Springer, 1996.

[20] U. Ascher, L. Petzold, Computer Methods for Ordinary Differential equations and Differential-Algebraic equations, SIAM, 1998.

[21] G. C. Buscaglia, R. F. Ausas, Variational formulations for surface tension, capillarity and wetting, Computer Methods in Applied Mechanics and Engineering 200 (45) (2011) 3011–3025.

[22] J. Donea, A. Huerta, J.-P. Ponthot, A. Rodríguez-Ferran, Arbitrary lagrangian–eulerian methods, Encyclopedia of computational mechanics.

[23] R. Codina, A stabilized finite element method for generalized stationary incompressible flows, Computer Methods in Applied Mechanics and Engineering 190 (20) (2001) 2681–2706.

[24] T. J. Hughes, L. P. Franca, M. Balestra, A new finite element formulation for computational fluid dynamics: V. circumventing the babuška-brezzi condition: A stable petrov-galerkin formulation of the stokes problem accommodating equal-order interpolations, Computer Methods in Applied Mechanics and Engineering 59 (1) (1986) 85–99.

[25] J. Douglas, J. Wang, An absolutely stabilized finite element method for the Stokes problem, Mathematics of Computation 52 (186) (1989) 495–508.

[26] R. P. Dwight, Robust mesh deformation using the linear elasticity equations, in: Computational Fluid Dynamics 2006, Springer, 2009, pp. 401–406.

[27] R. Codina, S. Badia, On some pressure segregation methods of fractional-step type for the finite element approximation of incompressible flow problems, Computer Methods in Applied Mechanics and Engineering 195 (23) (2006) 2900–2918.

[28] J. G. Heywood, R. Rannacher, Finite-element approximation of the nonstationary navier-stokes problem. part iv: Error analysis for second-order time discretization, SIAM Journal on Numerical Analysis 27 (2) (1990) 353–384.

[29] E. Becker, W. Hiller, T. Kowalewski, Experimental and theoretical investigation of large-amplitude oscillations of liquid droplets, Journal of Fluid Mechanics 231 (1991) 189–210.

[30] C. Galusinski, P. Vigneaux, On stability condition for bifluid flows with surface tension: Application to microfluidics, Journal of Computational Physics 227 (12) (2008) 6140–6164.

[31] T. Podgorski, J.-M. Flesselles, L. Limat, Corners, cusps, and pearls in running drops, Physical review letters 87 (3) (2001) 036102.

[32] T.-S. Jiang, O. Soo-Gun, J. C. Slattery, Correlation for dynamic contact angle, Journal of Colloid and Interface Science 69 (1) (1979) 74–77.

[33] R. Cox, The dynamics of the spreading of liquids on a solid surface. part 1. viscous flow, Journal of Fluid Mechanics 168 (1986) 169–194.

[34] P.-G. de Gennes, Deposition of langmuir-blodgett layers, Colloid and Polymer Science 264 (5) (1986) 463–465.

[35] M. Bracke, F. De Voeght, P. Joos, The kinetics of wetting: the dynamic contact angle, in: Trends in Colloid and Interface Science III, Springer, 1989, pp. 142–149.