

A curvilinear surface ALE formulation for self-evolving Navier-Stokes manifolds – Stabilized finite element formulation

Roger A. Sauer^{a,b,c,*}

^a*Institute for Structural Mechanics, Ruhr University Bochum, 44801 Bochum, Germany*

^b*Department of Structural Mechanics, Gdańsk University of Technology, 80-233 Gdańsk, Poland*

^c*Mechanical Engineering, Indian Institute of Technology Guwahati, Assam 781039, India*

Published[†] in *Comput. Methods Appl. Mech. Eng.*, DOI: [10.1016/j.cma.2025.118331](https://doi.org/10.1016/j.cma.2025.118331)

Submitted on 3 June 2025; Revised on 15 August 2025; Accepted on 17 August 2025

Abstract: This work presents a stabilized finite element formulation of the arbitrary Lagrangian-Eulerian (ALE) surface theory for Navier-Stokes flow on self-evolving manifolds developed in [Sauer \(2025\)](#). The formulation is physically frame-invariant, applicable to large deformations, and relevant to fluidic surfaces such as soap films, capillary menisci and lipid membranes, which are complex and inherently unstable physical systems. It is applied here to area-incompressible surface flows using a stabilized pressure-velocity (or surface tension-velocity) formulation based on quadratic finite elements and implicit time integration. The unknown ALE mesh motion is determined by membrane elasticity such that the in-plane mesh motion is stabilized without affecting the physical behavior of the system. The resulting three-field system is monolithically coupled, and fully linearized within the Newton-Raphson solution method. The new formulation is demonstrated on several challenging examples including shear flow on self-evolving surfaces and inflating soap bubbles with partial inflow on evolving boundaries. Optimal convergence rates are obtained in all cases. Particularly advantageous are C^1 -continuous surface discretizations, for example based on NURBS.

Keywords: Arbitrary Lagrangian-Eulerian formulation, area-incompressibility, fluid-structure interaction, Navier-Stokes equations, nonlinear finite element methods, pressure stabilization

1 Introduction

Simulating fluid flow on evolving surfaces is a very challenging problem due to various physical and computational complexities. There are the usual challenges of the Navier-Stokes equations – their nonlinearity, time-evolution, spatial resolution and stabilization, in particular for incompressibility. This then combines with the nonlinearities and potential instabilities of the large shape changes of an evolving manifold. Since fluids are characterized by velocity, and shapes by deformation, the overall problem is a fluid-structure interaction (FSI) problem. The arbitrary Lagrangian-Eulerian (ALE) formulation is a classical approach for FSI problems, and so it is a natural fit for evolving surface flows. [Sauer \(2025\)](#) provides a new, general ALE formulation for Navier-Stokes flow on self-evolving surfaces. Here now, a corresponding stable finite element implementation is developed, and then tested extensively on several challenging examples.

The theory for surface flows goes back to the sixties with the seminal work of [Scriven \(1960\)](#), while computational ALE formulations for bulk flows go back almost as far, see [Donea et al. \(2004\)](#) and references therein. However, a 3D simulation framework for Navier-Stokes flow on self-evolving surfaces, (where the surface shape is fully unknown), has appeared only recently

*corresponding author, email: roger.sauer@rub.de

[†]This pdf is the personal version of an article whose journal version is available at www.sciencedirect.com

with the concurrent works of [Torres-Sánchez et al. \(2019\)](#) and [Sahu et al. \(2020\)](#). Still, these and subsequent works have restrictions that are the motivation behind the present work.

The literature on evolving surface flows can be grouped into three categories – known surface motion, Lagrangian surface descriptions and ALE surface descriptions.

Many works consider known surfaces – either fixed surfaces or prescribed surface motion. The first general 3D finite element (FE) formulation for area-incompressible flows on fixed surfaces seems to be the work of [Nitschke et al. \(2012\)](#). It uses a vorticity-stream function formulation, and it has been extended to prescribed surface motions ([Reuther and Voigt, 2015](#)) and a discrete exterior calculus discretization ([Nitschke et al., 2017](#)). Subsequent pressure-velocity formulations have appeared using Taylor-Hood FE ([Fries, 2018](#)), trace FE ([Olshanskii et al., 2018](#)), Chorin projection ([Reuther and Voigt, 2018](#)), div-conforming FE ([Bonito et al., 2020](#); [Lederer et al., 2020](#)) and FE exterior calculus ([Toshniwal and Hughes, 2021](#)). Trace FE provide a level-set surface representation from a surrounding 3D FE mesh ([Olshanskii and Reusken, 2018](#)), while in the other cases surface FE are used, e.g. following [Dziuk and Elliott \(2013\)](#). Apart from FE methods, also spectral methods and meshless methods have been used for simulating surface flows ([Gross and Atzberger, 2018](#); [Gross et al., 2020](#)). In subsequent work, known surface formulations have been extended to study Kelvin-Helmholtz instabilities ([Lederer et al., 2020](#); [Jankuhn et al., 2021](#)), active flows ([Rank and Voigt, 2021](#)), immersed particles ([Rower et al., 2022](#)), and phase separations ([Sun et al., 2022](#)) on curved surfaces. The mathematical analysis of known surface flows also remains an ongoing topic ([Olshanskii et al., 2024](#); [Reusken, 2025](#); [Elliott and Mavrakakis, 2024](#)). A recent comparison of different discretization schemes for Stokes flow on fixed surfaces has been investigated in [Hardering and Praetorius \(2024\)](#).

Known surfaces are simpler, as the unknown fluid velocity is only characterized by two unknown tangential components. For Newtonian fluids, these velocity components are governed by the tangential surface Navier-Stokes equations that contain surface-projection operators. For self-evolving surfaces the fluid velocity has an additional unknown out-of-plane component. Instead of using this out-of-plane component together with the two tangential components as unknowns, one can take the three velocity components of a fixed Cartesian background coordinate system as unknowns, which actually facilitates the formulation and discretization of the governing equations (as no projection operators are needed), even when their solution is harder than the fixed surface case. This description has a long tradition in large-deformation FE formulations for solid membranes and shells ([Oden and Sato, 1967](#); [Simo et al., 1990](#)). They are commonly described in a convected, Lagrangian surface frame, which is much more straightforward than an Eulerian frame.

Due to this, initial formulations for evolving (fluid) surfaces have neglected in-plane flow and focussed on shape changes in a Lagrangian framework. The first 3D FE formulation of this kind is the work of [Brown et al. \(1980\)](#) that studies the deformation of static droplets by solving the Young-Laplace equation. Hence there is only one degree-of-freedom (dof) per surface node. This is different in the surface evolution algorithms of [Dziuk \(1991\)](#) and [Brakke \(1992\)](#) that have three nodal dofs. But in this case the surface flow can lead to large mesh distortions. These essentially correspond to zero in-plane shear modes, as was noted by [Feng and Klug \(2006\)](#) in their study of lipid bilayer membranes. Subsequently, [Ma and Klug \(2008\)](#) proposed an iterative mesh redistribution scheme to regularize tangential mesh motions. Mesh redistribution schemes were also employed by [Barrett et al. \(2008b\)](#) and [Dziuk \(2008\)](#) in their study of Willmore flows. On the other hand, the problem of zero in-plane shear modes does not arise in axisymmetric surface evolution, as are studied by [Arroyo and DeSimone \(2009\)](#). Apart from mesh redistribution, also remeshing can be used, as was done by [Elliott and Stinner \(2013\)](#) to study evolving biphasic lipid bilayers. Another approach to stabilize the tangential mesh motion is to add in-plane shear stiffness in such a way that the out-of-plane surface motion is

not affected. Such a scheme was developed in [Sauer et al. \(2014\)](#) and then adapted to deforming droplets ([Sauer, 2014, 2016](#)) and lipid bilayer membranes ([Sauer et al., 2017](#)). A similar mesh stabilization scheme was later proposed by [Dharmavaram \(2021\)](#). A further approach is to eliminate the tangential mesh motion by normal projection ([Sauer, 2014; Rangarajan and Gao, 2015](#)). In subsequent work, Lagrangian surface descriptions have been used to simulate deforming vesicles in 3D flow ([Barrett et al., 2015, 2016](#)), phase separation on deforming surfaces ([Zimmermann et al., 2019; Valizadeh and Rabczuk, 2019](#)), Navier-Stokes surface flows with meshless methods ([Suchde, 2021; Bharadwaj et al., 2022](#)), and particles or filaments embedded in fluidic membranes ([Dharmavaram et al., 2022; Sharma et al., 2023](#)).

As mentioned already, Lagrangian surface descriptions for evolving surfaces are interesting when in-plane shear flows are small or not important. If large in-plane shear flows occur, it is more suitable to consider in-plane Eulerian or in-plane ALE surface descriptions. Initial works of this kind have restricted themselves to axisymmetry ([Rahimi and Arroyo, 2012](#)) or Monge parametrizations ([Rangamani et al., 2013](#)). In the latter, the out-of-plane motion is defined through the in-plane parameterization, which does not allow for arbitrary large surface motions. More general are the ALE formulations of [Torres-Sánchez et al. \(2019\)](#), [Sahu et al. \(2020\)](#), [Reuther et al. \(2020\)](#) and [Al-Izzi and Morris \(2023\)](#). However, they restrict their simulations to Eulerian or near-Eulerian in-plane flow descriptions. Since these can still cause large mesh distortions when the surface evolves ([Sauer, 2025](#)), further generalization on the in-plane mesh motion is required. This is offered by an in-plane ALE formulation. In this way the tangential mesh velocity can be prescribed, e.g. based on the average surface velocity ([de Kinkelder et al., 2021](#)), or computed through suitable mesh equations. [Krause and Voigt \(2023\)](#) use the curvature-dependent mesh redistribution scheme of [Barrett et al. \(2008a\)](#) for this. While the approach has been successfully used to model phase transitions ([Bachini et al., 2023](#)) and wrinkling ([Krause and Voigt, 2024](#)) of fluidic membranes, the scheme can be expected to become unstable under mesh refinement, as the surface curvature is generally invariant with respect to tangential mesh motion. This problem is absent in the ALE formulation of [Sahu \(2024\)](#), which uses in-plane viscosity to control the mesh motion. But also this can become problematic, as it can be expected to lose stability when the mesh velocity approaches zero. This is avoided if the mesh motion is governed by in-plane elasticity, which is the approach taken here. As noted above, this can be done in such a way that the governing physical equations, and hence the material motion, are not affected.

The presented formulation focuses on self-evolving, area-incompressible Navier-Stokes manifolds with arbitrary shape and arbitrary (but unchanging) topology. The problem is governed by four coupled fields – the unknowns being the material velocity, the surface tension (= negative in-plane fluid pressure), the surface deformation and its time derivative, the ALE frame velocity. Pressure stabilization is achieved through the scheme of [Dohrmann and Bochev \(2004\)](#), while mesh stabilization is achieved through the in-plane elasticity scheme of [Sauer \(2014\)](#).

In summary, this work contains several important computational novelties:

- It presents a general surface ALE implementation for transient Navier-Stokes flow on self-evolving manifolds,
- using a stabilized pressure/velocity-based finite element formulation,
- with unknown mesh motion governed by in-plane membrane elasticity.
- The implementation avoids unnecessary splits into local bases and coordinate systems.
- It uses monolithic coupling and second order time integration.
- Its full linearization is provided, including follower loads on evolving inflow boundaries.
- Its proper convergence behavior is demonstrated on several challenging examples,
- that include vortices on deforming soap films and varying inflow on evolving boundaries.

The remainder of this paper is organized as follows. Sec. 2 summarizes the governing equations of the ALE flow description. The area-compressibility constraint is stabilized with the Dohrmann-Bochev scheme as discussed in Sec. 3. The finite element discretization is then provided in Sec. 4. This is illustrated by several numerical examples for fixed, prescribed and self-evolving surfaces in Secs. 5 and 6. The paper concludes with Sec. 7.

2 Governing ALE equations for surface flow

This section provides a (brief) summary of the most important equations for the kinematics, weak form and constitutive relations in the ALE frame. Their detailed presentation and derivation can be found in Sauer (2025).

2.1 ALE surface description

The current fluid surface $\mathcal{S}$ is described by the parameterization

$$\mathbf{x} = \mathbf{x}(\zeta^\alpha, t), \quad (1)$$

where the curvilinear coordinate ζ^α describes the ALE frame. It admits the special cases $\zeta^\alpha = \xi^\alpha$ and $\zeta^\alpha = \theta^\alpha$, that denote the Lagrangian and Eulerian surface coordinates, respectively. In order to distinguish these special cases in the mapping, we write $\mathbf{x} = \hat{\mathbf{x}}(\xi^\alpha, t)$ and $\mathbf{x} = \tilde{\mathbf{x}}(\theta^\alpha, t)$. Fig. 1 shows an illustration of mappings $\mathbf{x} = \mathbf{x}(\zeta^\alpha, t)$ and $\mathbf{x} = \hat{\mathbf{x}}(\xi^\alpha, t)$.

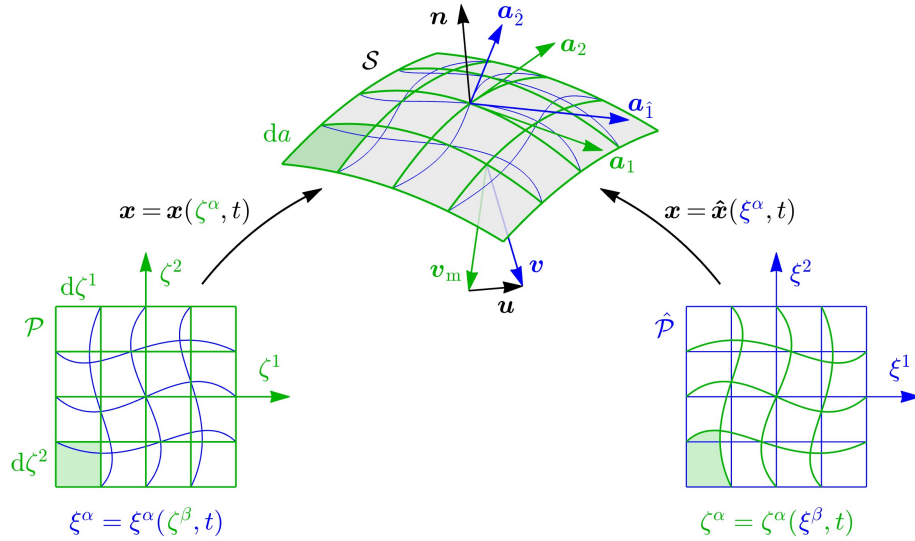


Figure 1: ALE surface description: The fluid surface $\mathcal{S}$ is described by the ALE mapping $\mathbf{x} = \mathbf{x}(\zeta^\alpha, t)$ from the ALE parameter domain $\mathcal{P}$. This mapping is then used for surface integration and discretization. It is generally different from the Lagrangian mapping $\mathbf{x} = \hat{\mathbf{x}}(\xi^\alpha, t)$ from the material parameter domain $\hat{\mathcal{P}}$. From the two mappings follow the surface velocities $\mathbf{v}_m := \partial \mathbf{x} / \partial t|_{\zeta^\alpha}$ and $\mathbf{v} := \partial \hat{\mathbf{x}} / \partial t|_{\xi^\alpha}$, and their in-plane difference $\mathbf{u} := \mathbf{v} - \mathbf{v}_m$.

Given (1), the tangent vectors in the ALE frame follow as

$$\mathbf{a}_\alpha := \frac{\partial \mathbf{x}}{\partial \zeta^\alpha}. \quad (2)$$

Together with the surface normal

$$\mathbf{n} := \frac{\mathbf{a}_1 \times \mathbf{a}_2}{\|\mathbf{a}_1 \times \mathbf{a}_2\|}, \quad (3)$$

they form a basis to decompose vectors and tensors into their in-plane and out-of-plane components. It is noted that other basis vectors can be defined, importantly $\mathbf{a}_{\hat{\alpha}} := \partial \hat{\mathbf{x}} / \partial \xi^\alpha$ and $\mathbf{a}_{\tilde{\alpha}} := \partial \tilde{\mathbf{x}} / \partial \theta^\alpha$. While those are important for the theoretical development given in [Sauer \(2025\)](#), they are not needed for the following computational description. In the following all Greek indices exclusivity refer to the general ALE parameterization.

From (2)-(3) follow the surface metric and its inverse,

$$a_{\alpha\beta} = \mathbf{a}_\alpha \cdot \mathbf{a}_\beta, \quad [a^{\alpha\beta}] = [a_{\alpha\beta}]^{-1}, \quad (4)$$

the dual basis vectors

$$\mathbf{a}^\alpha = a^{\alpha\beta} \mathbf{a}_\beta, \quad (5)$$

the parametric derivative

$$\dots, \alpha := \frac{\partial \dots}{\partial \zeta^\alpha}, \quad (6)$$

the surface curvature tensor components

$$b_{\alpha\beta} := \mathbf{a}_{\alpha,\beta} \cdot \mathbf{n}, \quad (7)$$

and the Christoffel symbols

$$\Gamma_{\alpha\beta}^\gamma := \mathbf{a}_{\alpha,\beta} \cdot \mathbf{a}^\gamma. \quad (8)$$

The latter two are the out-of-plane and in-plane components of $\mathbf{a}_{\alpha,\beta}$, respectively.

A reference configuration is introduced at time $t = 0$, and denoted by $\mathcal{S}_0$. Capital letters are used for the kinematical quantities from above on $\mathcal{S}_0$, i.e. $\mathbf{X} = \mathbf{x}|_{t=0}$, $\mathbf{A}_\alpha = \mathbf{a}_\alpha|_{t=0}$, $\mathbf{N} = \mathbf{n}|_{t=0}$, etc. Area elements in the current and reference configuration can be related to the ALE parameter domain by

$$\begin{aligned} da &= J_a d\zeta^1 d\zeta^2, \quad J_a := \sqrt{\det[a_{\alpha\beta}]}, \\ dA &= J_A d\zeta^1 d\zeta^2, \quad J_A := \sqrt{\det[A_{\alpha\beta}]}, \end{aligned} \quad (9)$$

see Fig. 1. Area changes in the ALE frame are then given by

$$J_m := \frac{J_a}{J_A}. \quad (10)$$

The time derivative of the ALE frame is

$$(\dots)' := \frac{\partial \dots}{\partial t} \Big|_{\zeta^\alpha}. \quad (11)$$

It defines the frame, or mesh, velocity

$$\mathbf{v}_m := \mathbf{x}' := \frac{\partial \mathbf{x}}{\partial t} \Big|_{\zeta^\alpha}. \quad (12)$$

In contrast to this, there is the material time derivative

$$(\dot{\dots}) = \frac{\partial \dots}{\partial t} \Big|_{\xi^\alpha} \quad (13)$$

that defines the fluid velocity

$$\mathbf{v} = \dot{\mathbf{x}} = \left. \frac{\partial \hat{\mathbf{x}}}{\partial t} \right|_{\xi^\alpha}. \quad (14)$$

The two velocities are related by

$$\mathbf{v} = \mathbf{v}_m + \dot{\zeta}^\alpha \mathbf{a}_\alpha. \quad (15)$$

where

$$\dot{\zeta}^\alpha = \mathbf{a}^\alpha \cdot (\mathbf{v} - \mathbf{v}_m), \quad (16)$$

or, equivalently, $\dot{\zeta}^\alpha = v^\alpha - v_m^\alpha$. For convenience we also introduce the relative fluid velocity

$$\mathbf{u} := \mathbf{v} - \mathbf{v}_m \quad (17)$$

that has the components $u^\alpha = \dot{\zeta}^\alpha$ and $u = 0$ in the $\{\mathbf{a}_1, \mathbf{a}_2, \mathbf{n}\}$ basis, see Fig. 1. Eq. (15) is an example of the fundamental surface ALE equation

$$\boxed{(\dot{\dots}) = (\dots)' + \dots_{,\alpha} \dot{\zeta}^\alpha} \quad (18)$$

that is needed to translate time derivatives. For example, in case of the acceleration it reads

$$\dot{\mathbf{v}} = \mathbf{v}' + \mathbf{v}_{,\alpha} (v^\alpha - v_m^\alpha). \quad (19)$$

Another important relation is,

$$\mathbf{v}_{,\alpha} = \dot{\mathbf{a}}_\alpha + \dot{\zeta}_{,\alpha}^\gamma \mathbf{a}_\gamma, \quad (20)$$

where $\dot{\zeta}_{,\alpha}^\gamma$ is the parametric derivative of $\dot{\zeta}^\gamma$, and not the material time derivative of $\zeta_{,\alpha}^\gamma$ (which is zero), as generally spatial and temporal differentiation do not commute for the ALE parameterization. To be precise, spatial differentiation w.r.t. ζ^α does not generally commute with the material time derivative (at fixed ξ^α). Only spatial differentiation w.r.t. to ξ^α does. Likewise, spatial differentiation w.r.t. ζ^α commutes with temporal differentiation at fixed ζ^α .

From the parametric derivative follows the surface divergence,

$$\text{div}_s \mathbf{v} = \mathbf{v}_{,\alpha} \cdot \mathbf{a}^\alpha, \quad (21)$$

and the surface gradient,

$$\nabla_s \mathbf{v} = \mathbf{v}_{,\alpha} \otimes \mathbf{a}^\alpha, \quad (22)$$

of the fluid velocity. The symmetric part of the latter is the rate of surface deformation tensor

$$\mathbf{d} = (\mathbf{v}_{,\alpha} \otimes \mathbf{a}^\alpha + \mathbf{a}^\alpha \otimes \mathbf{v}_{,\alpha})/2. \quad (23)$$

Like $\nabla_s \mathbf{v}$, it has both in-plane and out-of-plane components. Its in-plane components are

$$d^{\alpha\beta} = \mathbf{a}^\alpha \cdot \mathbf{d} \mathbf{a}^\beta = (a^{\alpha\gamma} \mathbf{v}_{,\gamma} \cdot \mathbf{a}^\beta + a^{\beta\gamma} \mathbf{v}_{,\gamma} \cdot \mathbf{a}^\alpha)/2. \quad (24)$$

It is noted that $d^{\alpha\beta} a_{\alpha\beta} = \text{div}_s \mathbf{v}$ and

$$2d^{\alpha\beta} \mathbf{a}_\beta = \mathbf{a}_s^{\alpha\beta} \mathbf{v}_{,\beta}, \quad (25)$$

with the surface tensor

$$\mathbf{a}_s^{\alpha\beta} := a^{\alpha\beta} \mathbf{i} + \mathbf{a}^\beta \otimes \mathbf{a}^\alpha. \quad (26)$$

Eqs. (25)-(26) are needed for the FE formulation later. It is finally noted that $d^{\alpha\beta}$ is symmetric, while $\mathbf{a}_s^{\alpha\beta}$ is not.

2.2 Surface constitution in ALE form

The stress tensor of an area-incompressible Newtonian fluid surface can be written as (Sauer, 2025)

$$\boldsymbol{\sigma} = q \mathbf{i} + 2\eta \mathbf{d}_s, \quad (27)$$

where q is the surface tension, $\mathbf{i} := \mathbf{a}_\alpha \otimes \mathbf{a}^\alpha$ the surface identity tensor, η the surface viscosity, and $\mathbf{d}_s = d^{\alpha\beta} \mathbf{a}_\alpha \otimes \mathbf{a}_\beta$ the in-plane part of $\mathbf{d}$. Generally, q , $\mathbf{i}$ and $\mathbf{d}_s$ are functions of position $\mathbf{x}$. The components of $\boldsymbol{\sigma} = \sigma^{\alpha\beta} \mathbf{a}_\alpha \otimes \mathbf{a}_\beta$ in the ALE frame are

$$\sigma^{\alpha\beta} = q a^{\alpha\beta} + 2\eta d^{\alpha\beta}. \quad (28)$$

Also the velocity gradient within $d^{\alpha\beta}$ is described with respect to the ALE frame according to Eqs. (22) and (6). The constitutive model defined by Eqs. (27) and (28) can be extended to surface fluids with bending resistance, see Sauer (2025), but this is not considered here. The surface tension q is defined through the area-incompressibility constraint

$$\text{div}_s \mathbf{v} = 0. \quad (29)$$

2.3 Governing equations in ALE form

The governing strong form equations are derived in Sauer (2025) and summarized in Box 1 there. They consists of four coupled PDEs for the four unknown fields: fluid velocity $\mathbf{v}$, surface tension q , mesh velocity $\mathbf{v}_m$ and surface position $\mathbf{x}$. Since $\mathbf{v}$, $\mathbf{v}_m$ and $\mathbf{x}$ are vectors, there are 10 unknown components altogether. The associated PDEs for functions $\mathbf{v}(\zeta^\alpha, t)$, $q(\zeta^\alpha, t)$, $\mathbf{v}_m(\zeta^\alpha, t)$ and $\mathbf{x}(\zeta^\alpha, t)$ are the surface Navier-Stokes equation,

$$\text{div}_s \boldsymbol{\sigma}^T + \mathbf{f} = \rho \dot{\mathbf{v}}, \quad (30)$$

(which is a vector-valued equation that has both in-plane and out-of-plane components; also for the body force $\mathbf{f}$), the area-incompressibility constraint (29), suitable mesh equations, and $\mathbf{v}_m = \dot{\mathbf{x}}$. The last equation implies that the problem is time-dependent, even when steady flow conditions are considered ($\rho \mathbf{v}' \approx \mathbf{0}$). It will be approximated by a finite difference-based time integration scheme, effectively eliminating one unknown field, e.g. $\mathbf{x}$. The remaining three unknowns fields are solved by applying the finite element method to the remaining PDEs. For fixed or prescribed surface motion, i.e. known $\mathbf{x}(t)$, only two unknown fields remain: $\mathbf{v}$ and q . This is considered in the examples of Sec. 5. Sec. 6 then consider unknown $\mathbf{v}$, q and $\mathbf{v}_m$.

Two mesh equations are needed: Firstly, the constraint

$$\mathbf{n} \cdot \mathbf{v}_m = \mathbf{n} \cdot \mathbf{v}, \quad (31)$$

which ascertains that the out-of-plane velocity components $v_n := \mathbf{v} \cdot \mathbf{n}$ and $v_{mn} := \mathbf{v}_m \cdot \mathbf{n}$ agree. The equation could be used to eliminate one of these components, but this is not done here. Instead the constraint will be enforced in weak form. Secondly, an equation for the in-plane components of the mesh velocity is needed. Two options will be used for that here: (I) Setting them to zero, which implies that the in-plane description is Eulerian (i.e. $\zeta^\alpha = \theta^\alpha$), or (II) determining the in-plane mesh motion from

$$\sigma_{m;\beta}^{\alpha\beta} = 0, \quad (32)$$

which is the in-plane equilibrium equation of a membrane. Here, the simple nonlinear elasticity model

$$\tau_m^{\alpha\beta} = \mu_m (A^{\alpha\beta} - a^{\alpha\beta}) \quad (33)$$

is used for the membrane stress $\sigma_m^{\alpha\beta} = \tau_m^{\alpha\beta}/J_m$ in (32). Since this membrane stress is not entering the surface Navier-Stokes equations, it does not affect the physical behavior of the system. It only effects the in-plane mesh motion, by effectively stabilizing it through the shear stiffness μ_m , which is a purely numerical parameter. This stabilization was developed in Sauer (2014) to simulate quasi-static droplets.

Both mesh options will also be enforced in weak form.

The weak form equations for $\mathbf{v}$, q and $\mathbf{v}_m$ are given by the statements

$$\begin{aligned} G &:= G_{\text{in}} + G_{\text{int}} - G_{\text{ext}} = 0 & \forall \delta \mathbf{x} \in \mathcal{V}, \\ \bar{G} &:= \bar{G}_{\text{div}} = 0 & \forall \delta q \in \mathcal{Q}, \\ \tilde{G} &:= \tilde{G}_i + \tilde{G}_o = 0 & \forall \mathbf{w} \in \mathcal{W}. \end{aligned} \quad (34)$$

derived in Sauer (2025). They can also be combined in the single statement

$$G + \bar{G} + \tilde{G} = 0, \quad \forall \delta \mathbf{x} \in \mathcal{V}, \delta q \in \mathcal{Q}, \mathbf{w} \in \mathcal{W}. \quad (35)$$

Here and in (34), $\delta \mathbf{x}$, δq and $\mathbf{w}$ denote the test functions associated with $\mathbf{v}$, q and $\mathbf{v}_m$. The weak form is complemented by boundary and initial conditions, as well as the Dohrmann-Bochev stabilization scheme, discussed in Sec. 3. This section also contains a summary of all the individual contributions of Eq. (34).

2.4 Boundary and initial conditions in ALE form

The governing equations of Sec. 2.3 require boundary and initial conditions in order to be uniquely solvable. For the fluid flow the usual velocity and traction boundary conditions,

$$\begin{aligned} \mathbf{v}(\zeta^\alpha, t) &= \bar{\mathbf{v}}(\zeta^\alpha, t) \quad \forall \mathbf{x} \in \partial_v \mathcal{S} \quad \text{and} \quad \forall t \in \mathcal{T}, \\ \mathbf{T}(\zeta^\alpha, t) &= \bar{\mathbf{T}}(\zeta^\alpha, t) \quad \forall \mathbf{x} \in \partial_T \mathcal{S} \quad \text{and} \quad \forall t \in \mathcal{T}, \end{aligned} \quad (36)$$

apply. Here $\partial_v \mathcal{S}$ and $\partial_T \mathcal{S}$ denote the boundaries of the fluid surface where velocities or tractions are prescribed, while $\mathcal{T}$ denotes the considered time interval. $\partial_v \mathcal{S}$ contains inflow and outflow boundaries. These do not require to observe mass conservation, as long as the surface area can change. An example is provided in Sec. 6.3. This example also shows that traction $\bar{\mathbf{T}}$ can be surface deformation-dependent (a so-called follower load), which requires linearization in implicit time-stepping schemes. There may also be no boundary ($\partial_v \mathcal{S} = \partial_T \mathcal{S} = \emptyset$), as in the examples of Secs. 5-6.2. The traction BC (36.2) provides a condition for surface tension q . In the absence of traction BCs ($\partial_T \mathcal{S} = \emptyset$), q has to be specified at some location, to ascertain its uniqueness. If mesh velocity $\mathbf{v}_m$ is an unknown (instead of a prescribed) field, it also requires a boundary condition (if a boundary is present). In the example of Sec. 6.3 the zero Dirichlet BC,

$$\mathbf{v}_m(\zeta^\alpha, t) = \mathbf{0} \quad \forall \mathbf{x} \in \partial \mathcal{S} \quad \text{and} \quad \forall t \in \mathcal{T}, \quad (37)$$

is used. In the subsequent FE formulation, BCs (36.1) and (37) are enforced by direct elimination, while BC (36.2) enters the external virtual work G_{ext} , see Box 1 and Sec. 4.2.1. For fixed surfaces, the additional velocity boundary condition

$$v_n(\zeta^\alpha, t) = 0 \quad \forall \mathbf{x} \in \mathcal{S} \quad \text{and} \quad \forall t \in \mathcal{T}, \quad (38)$$

appears. It is a Dirichlet BC that can be replaced by the equivalent Neumann BC on the surface pressure p , see Sec. 5.2.5 for an example.

The initial conditions at $t = 0$ are

$$\begin{aligned} \mathbf{v}(\zeta^\alpha, 0) &= \mathbf{v}_0(\zeta^\alpha) \quad \forall \mathbf{x} \in \mathcal{S}, \\ q(\zeta^\alpha, 0) &= q_0(\zeta^\alpha) \quad \forall \mathbf{x} \in \mathcal{S}, \\ \mathbf{v}_m(\zeta^\alpha, 0) &= \mathbf{v}_m^0(\zeta^\alpha) \quad \forall \mathbf{x} \in \mathcal{S}, \\ \mathbf{x}(\zeta^\alpha, 0) &= \mathbf{X}(\zeta^\alpha) \quad \forall \mathbf{x} \in \mathcal{S}. \end{aligned} \tag{39}$$

The chosen initial fluid velocity, surface tension, mesh velocity and mesh position fields $\mathbf{v}_0$, q_0 , $\mathbf{v}_m^0$ and $\mathbf{X}$ have to be compatible, i.e. agree with Eqs. (29)-(32).

3 Dohrmann-Bochev stabilization and resulting weak form

Incompressible flows – such as the one described here – are mixed problems. Their discretization needs to satisfy the Ladyzhenskaya-Babuška-Brezzi (LBB) condition (Ladyzhenskaya, 1969; Babuška, 1973; Brezzi, 1974), or use a stabilization technique. Popular approaches are LBB-conforming velocity-pressure – e.g. Taylor-Hood – discretizations, div-conforming discretizations, or stabilization methods that circumvent the LBB condition such as the pressure stabilizing Petrov Galerkin (PSPG) approach (Hughes et al., 1986) and the polynomial pressure projection stabilization (PPPS) approach by Dohrmann and Bochev (2004). Most existing surface Navier-Stokes flows use Taylor-Hood (Fries, 2018; Krause and Voigt, 2023) or Taylor-Hood-like discretizations (Torres-Sánchez et al., 2019). The Dohrmann-Bochev (DB) scheme has only recently been adapted to surface flows (Sahu et al., 2020). The scheme is also based on velocity-pressure discretizations, but instead of picking the pressure from a lower polynomial space in order to satisfy the LBB condition directly, it can be picked from the same space, and then projected to a lower dimensional space that satisfies the LBB condition. For instance one can use bi-quadratic velocity and pressure approximations, and then project the latter onto a linear subspace, which is the case used here. The projection is achieved by square error minimization. This projection is simple and can be treated purely on the element level, which makes the method comparably easy to implement. Local, inexpensive static condensation can then be used to eliminate the auxiliary pressure/surface tension variable.

The DB scheme subtracts the stabilization term

$$G_{\text{DB}} = \frac{\alpha_{\text{DB}}}{\eta} \int_{\mathcal{S}} (\delta q - \delta \tilde{q})(q - \tilde{q}) \, da \tag{40}$$

from weak form (35). Here, $q(\zeta^\alpha, t)$ is the actual surface tension, while $\tilde{q}(\zeta^\alpha, t)$ is an auxiliary surface tension that is of lower interpolation order than q . Further, η is the viscosity and α_{DB} is a chosen stabilization parameter. Eq. (40) can be derived from the variation of the potential

$$\Pi_{\text{DB}} = \frac{\alpha_{\text{DB}}}{2\eta} \int_{\mathcal{S}} (q - \tilde{q})^2 \, da. \tag{41}$$

In the continuum limit of the discretization, $\tilde{q}$ becomes equal to q , and hence nothing is changed by including (40) in the weak form. Contrary to Sahu et al. (2020), we choose to integrate the stabilization term over the current surface $\mathcal{S}$ instead of the initial, reference surface $\mathcal{S}_0$, as this leads to more accurate results for large surface stretches. When surface stretches are small, (40) can be integrated over $\mathcal{S}_0$ without significant accuracy loss, as is seen in Sec. 6.3.

Box 1 summarizes all the weak form equations resulting from (34) and (40). The Euler-Lagrange equations of this weak form are the equation of motion (30), the mesh equations ($\mathbf{v}_m \cdot \mathbf{n} = \mathbf{v} \cdot \mathbf{n}$ and either (32) or $\mathbf{v}_m \cdot \mathbf{a}_\alpha = 0$) and

$$\begin{aligned} \eta \operatorname{div}_s \mathbf{v} - \alpha_{\text{DB}}(q - \tilde{q}) &= 0, \\ q - \tilde{q} &= 0, \end{aligned} \tag{42}$$

Coupled weak form equations:

$$\begin{aligned}
G &:= G_{\text{in}} + G_{\text{int}} - G_{\text{ext}} = 0 & \forall \delta \mathbf{x} \in \mathcal{V}, & \quad (\text{weak form for fluid velocity } \mathbf{v}) \\
\bar{G} &:= \bar{G}_{\text{div}} - \bar{G}_{\text{DB}} = 0 & \forall \delta q \in \mathcal{Q}, & \quad (\text{weak form for surface tension } q) \\
\check{G} &:= \check{G}_{\text{DB}} = 0 & \forall \delta \check{q} \in \check{\mathcal{Q}}, & \quad (\text{weak form for stabilization field } \check{q}) \\
\tilde{G} &:= \begin{cases} \tilde{G}_0 & (\text{option I}) \\ \tilde{G}_{\text{el}} & (\text{option II}) \end{cases} = 0 & \forall \mathbf{w} \in \mathcal{W}, & \quad (\text{weak form for mesh velocity } \mathbf{v}_m)
\end{aligned}$$

with:

$$\begin{aligned}
G_{\text{in}} &:= \int_S \delta \mathbf{x} \cdot \rho \dot{\mathbf{v}} \, da = G_{\text{trans}} + G_{\text{conv}}, & G_{\text{trans}} &:= \int_S \delta \mathbf{x} \cdot \rho \mathbf{v}' \, da, \\
G_{\text{conv}} &:= \int_S \delta \mathbf{x} \cdot \rho \mathbf{v}_{,\alpha} (v^\alpha - v_m^\alpha) \, da, & G_{\text{int}} &:= \int_S \delta \mathbf{x}_{,\alpha} \cdot \sigma^{\alpha\beta} \mathbf{a}_\beta \, da, \\
G_{\text{ext}} &:= \int_S \delta \mathbf{x} \cdot \mathbf{f} \, da + \int_{\partial_T S} \delta \mathbf{x} \cdot \bar{\mathbf{T}} \, ds, & \bar{G}_{\text{div}} &:= \int_S \delta q \operatorname{div}_s \mathbf{v} \, da, \\
\tilde{G}_0 &:= \alpha_m \int_{S_0} \mathbf{w} \cdot (\mathbf{v}_m - (\mathbf{n} \otimes \mathbf{n}) \mathbf{v}) \, dA, & \bar{G}_{\text{DB}} &:= \frac{\alpha_{\text{DB}}}{\eta} \int_S \delta q (q - \check{q}) \, da, \\
\tilde{G}_{\text{el}} &:= \int_{S_0} w_{\alpha;\beta} \tau_m^{\alpha\beta} \, dA + \alpha_m \int_{S_0} w \mathbf{n} \cdot (\mathbf{v}_m - \mathbf{v}) \, dA, & \check{G}_{\text{DB}} &:= \frac{\alpha_{\text{DB}}}{\eta} \int_S \delta \check{q} (q - \check{q}) \, da.
\end{aligned}$$

Box 1: Coupled weak form equations for stabilized area-incompressible surface flows described in the ALE frame. Their finite element discretization is discussed in Sec. 4. For this, the auxiliary variable $\check{q}$ of the Dohrmann-Bochev stabilization scheme can be eliminated locally, such that only three discretized weak form equations remain. Two options are used for the in-plane components of $\mathbf{v}_m$: Setting them to zero (I), or determining them from the in-plane equations of an elastic membrane (II).

implying $\check{q} = q$ and $\operatorname{div}_s \mathbf{v} = 0$. Thus nothing changes in the continuum limit. In the discrete setting however, different approximation spaces are chosen for q and $\check{q}$, and so extra terms appear. As noted above, here q is chosen from the same bi-quadratic approximation space as $\mathbf{v}$, while $\check{q}$ is picked from a linear subspace such that it can be eliminated efficiently on the element level. This is presented in the following section.

Remark 3.1: Written separately, the weak forms $G = 0$, $\bar{G} = 0$, $\check{G} = 0$ and $\tilde{G} = 0$ do not need to have agreeing units. Variations $\delta \mathbf{x}$ and $\mathbf{w}$ can thus be interpreted as displacements, and δq and $\delta \check{q}$ as surface tensions, giving units of energy for G and $\tilde{G}$, and power for $\bar{G}$ and $\check{G}$. But written as the sum $G + \bar{G} + \check{G} + \tilde{G} = 0$, the unit of each term should agree. This can either be achieved by including factors in front of each term, such as α_{DB} and α_m , or by reinterpreting $\delta \mathbf{x}$ and $\mathbf{w}$ as velocities.

Remark 3.2: Apart from pressure stabilization, there is also convective stabilization, such as streamline upwind Petrov-Galerkin (SUPG) stabilization (Brooks and Hughes, 1982). It is required when the convective velocity $\mathbf{u} = \mathbf{v} - \mathbf{v}_m$ is large in comparison to the computational grid. The need for convective stabilization can be assessed from the local, mesh-specific Péclet number

$$\text{Pe} := \frac{h \|\mathbf{u}\|}{2\nu} = \frac{\rho h \|\mathbf{u}\|}{2\eta}, \quad (43)$$

where h is the grid spacing in flow direction (half the element length for quadratic elements), ρ is the surface density, and ν and η the kinematic and dynamic (surface) viscosities. Instabilities can arise when Pe exceeds unity (Donea and Huerta, 2003). In the subsequent examples of Sec. 5 and 6 meshes are resolved to ensure $\text{Pe} < 1$.

4 Finite element discretization

This section discussed the finite element discretization and solution of the preceding weak form. First the spatial discretization of the different fields and their integrals is discussed in Secs. 4.1-4.2, followed by a suitable implicit time stepping scheme in Sec. 4.3, the resulting Newton-Raphson procedure in Sec. 4.4, and its normalization in Sec. 4.5.

4.1 FE interpolation

4.1.1 Surface deformation and mesh motion

The surface is discretized into n_{el} finite elements using n_{no} nodes. Within each finite element Ω^e , the surface description (1) is approximated by the interpolation

$$\mathbf{x} \approx \mathbf{x}^h(\zeta^\alpha, t) = \sum_{I=1}^{n_e} N_I(\zeta^\alpha) \mathbf{x}_I(t), \quad (44)$$

where N_I denotes the shape function, $\mathbf{x}_I$ denotes the position of node I , and n_e denotes the number of nodes of element Ω^e . Eq. (44) can be rewritten in the compact form

$$\mathbf{x}^h = \mathbf{N}_e \mathbf{x}_e, \quad (45)$$

with $\mathbf{N}_e := [N_1 \mathbf{1}, N_2 \mathbf{1}, \dots, N_{n_e} \mathbf{1}]$ and $\mathbf{x}_e := [\mathbf{x}_1^T, \mathbf{x}_2^T, \dots, \mathbf{x}_{n_e}^T]^T$. Here, $\mathbf{1}$ denotes the identity tensor in $\mathbb{R}^3$, while

$$\mathcal{C}_e := \{e_1, e_2, \dots, e_{n_e}\} \quad (46)$$

is the set of global node numbers of element e and thus specifies the element connectivity. Evaluating $\mathbf{x}^h$ at $t = 0$ lets us define a fixed reference configuration for the surface, written as

$$\mathbf{X}^h = \mathbf{N}_e \mathbf{X}_e, \quad \mathbf{X}_e = \mathbf{x}_e(0). \quad (47)$$

Note that, in general (as long as $\zeta^\alpha \neq \xi^\alpha$) $\mathbf{x}_I(t)$ is not the current position of the fluid particle that was initially at $\mathbf{x}_I(0)$. Instead the initial position of the particle follows from the integration of $\mathbf{v}$ along the path defined by fixed ξ^α .

According to (2), the discretized tangential basis vectors then follow from the interpolation

$$\mathbf{a}_\alpha^h = \mathbf{N}_{e,\alpha} \mathbf{x}_e, \quad (48)$$

where $\mathbf{N}_{e,\alpha} := [N_{1,\alpha} \mathbf{1}, N_{2,\alpha} \mathbf{1}, \dots, N_{n_e,\alpha} \mathbf{1}]$ and $N_{I,\alpha} = \partial N_I / \partial \zeta^\alpha$. This then allows to evaluate all the surface quantities defined in Eqs. (4)-(8) for the current surface. Further, evaluating (48) at $t = 0$ defines the initial basis

$$\mathbf{A}_\alpha^h = \mathbf{N}_{e,\alpha} \mathbf{X}_e, \quad (49)$$

which allows to evaluate the surface quantities defined in Eqs. (4)-(8) for the initial surface.

According to definition (12), the mesh velocity is then approximated by the interpolation

$$\mathbf{v}_m^h = \sum_{I=1}^{n_e} N_I(\zeta^\alpha) \mathbf{x}'_I(t), \quad (50)$$

where $\mathbf{x}'_I := \partial \mathbf{x}_I / \partial t =: \mathbf{v}_{mI}$ is the nodal mesh velocity. In compact notation this becomes

$$\mathbf{v}_m^h = \mathbf{N}_e \mathbf{x}'_e, \quad (51)$$

with $\mathbf{x}'_e := \partial \mathbf{x}_e / \partial t =: \mathbf{v}_m^e$.

Remark 4.1: The discretized time derivative of $\mathbf{a}_\alpha$ in the ALE frame follows from (48) as

$$\mathbf{a}'_\alpha{}^h = \mathbf{N}_{e,\alpha} \mathbf{x}'_e. \quad (52)$$

This is the same as $\mathbf{v}_{m,\alpha}^h$, as Eq. (51) indicates. However, $\mathbf{a}'_\alpha$ is generally not the same as the material time derivative of $\mathbf{a}_\alpha$, which according to (18) is

$$\dot{\mathbf{a}}_\alpha = \mathbf{a}'_\alpha + \dot{\zeta}^\beta \mathbf{a}_{\alpha,\beta}. \quad (53)$$

In the discrete setting this becomes

$$\dot{\mathbf{a}}_\alpha^h = \mathbf{N}_{e,\alpha} \mathbf{x}'_e + \dot{\zeta}^{\beta h} \mathbf{N}_{e,\alpha\beta} \mathbf{x}_e, \quad (54)$$

where $\dot{\zeta}^{\beta h}$ follows from Remark 4.2.

4.1.2 Fluid velocity and its derivatives

Analogous to interpolation (44), the fluid velocity field is interpolated by

$$\mathbf{v}^h = \sum_{I=1}^{n_e} N_I(\zeta^\alpha) \mathbf{v}_I(t), \quad (55)$$

or in compact notation,

$$\mathbf{v}^h = \mathbf{N}_e \mathbf{v}_e, \quad (56)$$

with $\mathbf{v}_e := [\mathbf{v}_1^T, \mathbf{v}_2^T, \dots, \mathbf{v}_{n_e}^T]^T$. Note, that in general, $\mathbf{v}_I \neq \partial \mathbf{x}_I / \partial t = \mathbf{v}_{mI}$ (as long as $\zeta^\alpha \neq \xi^\alpha$). Instead $\mathbf{v}_I$ is an independent variable – independent from $\mathbf{x}_I$. It characterizes the material fluid velocity, while $\mathbf{x}_I$ characterizes the mesh motion. Their relative velocity according to (17) now follows from expressions (51) and (56) as

$$\mathbf{u}^h = \mathbf{N}_e (\mathbf{v}_e - \mathbf{x}'_e). \quad (57)$$

Remark 4.2: Given (57) one can obtain a discretized expression for its in-plane component $u^\alpha = \dot{\zeta}^\alpha$ according to (16), i.e.

$$\dot{\zeta}^{\alpha h} = \mathbf{a}^{\alpha h} \cdot \mathbf{N}_e (\mathbf{v}_e - \mathbf{x}'_e), \quad (58)$$

where $\mathbf{a}^{\alpha h}$ is the dual basis to $\mathbf{a}_\alpha^h$.

From (11) and (56) follows

$$\mathbf{v}'^h = \mathbf{N}_e \mathbf{v}'_e, \quad (59)$$

where $\mathbf{v}'_e := \partial \mathbf{v}_e / \partial t$. From (56) further follows

$$\mathbf{v}_{,\alpha}^h = \mathbf{N}_{e,\alpha} \mathbf{v}_e. \quad (60)$$

It is reiterated that in the ALE frame, spatial differentiation (w.r.t. ζ^α) does not commute with the material time derivative, so that $\mathbf{v}_{,\alpha}^h \neq \dot{\mathbf{a}}_\alpha^h$. Instead (20) applies.

Remark 4.3: If needed, $\dot{\zeta}_{,\alpha}^\gamma$ can now be obtained as follows. From (20) first follows $\dot{\zeta}_{,\alpha}^\gamma = \mathbf{a}^\gamma \cdot (\mathbf{v}_{,\alpha} - \dot{\mathbf{a}}_\alpha)$. With (53) and (8) this then yields

$$\dot{\zeta}_{,\alpha}^\gamma = \mathbf{a}^\gamma \cdot (\mathbf{v}_{,\alpha} - \mathbf{a}'_\alpha) - \dot{\zeta}^\beta \Gamma_{\alpha\beta}^\gamma. \quad (61)$$

In the discrete setting this becomes

$$\dot{\zeta}_{,\alpha}^{\gamma h} = [\mathbf{a}^{\gamma h} \cdot \mathbf{N}_{e,\alpha} - \Gamma_{\alpha\beta}^{\gamma h} \mathbf{a}^{\beta h} \cdot \mathbf{N}_e] (\mathbf{v}_e - \mathbf{x}'_e), \quad (62)$$

due to (58).

The surface velocity gradient and divergence can be evaluated from (22), (21), (48), (5) and (60). The discretized surface divergence, in particular, can conveniently be written as

$$\operatorname{div}_s^h \mathbf{v}^h = \mathbf{D}_e \mathbf{v}_e, \quad (63)$$

where the surface divergence operator

$$\mathbf{D}_e := \mathbf{a}^{\alpha h} \cdot \mathbf{N}_{e,\alpha} \quad (64)$$

depends on the surface and needs to be linearized if the surface is changing. Further, the discretization of Eq. (25) is

$$2d_h^{\alpha\beta} \mathbf{a}_\beta^h = \mathbf{a}_s^{\alpha\beta} \mathbf{N}_{e,\beta} \mathbf{v}_e. \quad (65)$$

It is needed for the internal FE force vector in Sec. 4.2.

4.1.3 Surface tension and its stabilization

The Lagrange multiplier q is interpolated similarly to $\mathbf{x}$ and $\mathbf{v}$, i.e.

$$q^h = \sum_{I=1}^{m_e} L_I(\zeta^\alpha) q_I(t), \quad (66)$$

where L_I , $I = 1, \dots, m_e$ denote the nodal shape functions for q that can be generally different from N_I . Also, m_e , the number of corresponding nodes can be generally different from n_e . In compact notation, (66) becomes

$$q^h = \mathbf{L}_e \mathbf{q}_e, \quad (67)$$

where $\mathbf{L}_e := [L_1, L_2, \dots, L_{m_e}]$ and $\mathbf{q}_e := [q_1, q_2, \dots, q_{m_e}]^T$. A common approach is to pick L_I and m_e such that the LBB condition is satisfied. An alternative is to pick $L_I = N_I$ and $m_e = n_e$ and use a stabilization method, such as the Dohrmann-Bochev method introduced in Sec. 3. This latter approach is used with bi-quadratic Lagrange shape functions in all subsequent numerical examples. It requires the interpolation of the auxiliary field $\check{q}$, written as

$$\check{q}^h = \sum_{I=1}^{\check{m}_e} \check{L}_I(\zeta^\alpha) \check{q}_I(t) = \check{\mathbf{L}}_e \check{\mathbf{q}}_e. \quad (68)$$

A possible choice is to use $\check{m}_e = 3$ with

$$\check{\mathbf{L}}_e = [1, \zeta^1, \zeta^2]. \quad (69)$$

It provides effective stabilization and allows to eliminate unknown $\check{\mathbf{q}}_e$ locally for each element, as is shown in Sec. 4.2.2 below. It is noted that for choice (69), $\check{q}_I$ are not nodal values (that can be associated with nodes), but rather correspond to the coefficients of the first order polynomial induced by $\check{\mathbf{L}}_e$.

The variations $\delta \mathbf{x}$, $\mathbf{w}$, δq and $\delta \check{q}$ are interpolated in the same way as $\mathbf{x}$, $\mathbf{v}_m$, q and $\check{q}$, i.e.

$$\begin{aligned} \delta \mathbf{x}^h &= \mathbf{N}_e \delta \mathbf{x}_e, \\ \mathbf{w}^h &= \mathbf{N}_e \mathbf{w}_e, \\ \delta q^h &= \mathbf{L}_e \delta \mathbf{q}_e, \\ \delta \check{q}^h &= \check{\mathbf{L}}_e \delta \check{\mathbf{q}}_e. \end{aligned} \quad (70)$$

From this follows $\delta \mathbf{a}_\alpha^h = \mathbf{N}_{e,\alpha} \delta \mathbf{x}_e$.

All the above equations show that the entire set of field variables is described exclusively by the parameters ζ^α and t . Parameters ξ^α and θ^α , introduced earlier, are not needed for the FE description.

To shorten notation, superscript h is skipped in the subsequent presentation.

4.2 Discretized weak form

Using the expressions from Sec. 4.1 the four weak form equations in Box 1 are now discretized.

4.2.1 Equation of motion

The first weak form in Box 1 contains the three parts G_{in} , G_{int} and G_{ext} . Discretizing them leads to the elemental contributions $G_{\text{in}}^e = \delta \mathbf{x}_e^T \mathbf{f}_{\text{in}}^e$, $G_{\text{int}}^e = \delta \mathbf{x}_e^T \mathbf{f}_{\text{int}}^e$ and $G_{\text{ext}}^e = \delta \mathbf{x}_e^T \mathbf{f}_{\text{ext}}^e$, where the vectors $\mathbf{f}_{\text{in}}^e$, $\mathbf{f}_{\text{int}}^e$ and $\mathbf{f}_{\text{ext}}^e$ contain the corresponding nodal forces of finite element Ω^e .

Discretizing G_{in} leads to the elemental inertia vector $\mathbf{f}_{\text{in}}^e = \mathbf{f}_{\text{trans}}^e + \mathbf{f}_{\text{conv}}^e$ with the transient part

$$\mathbf{f}_{\text{trans}}^e := \mathbf{m}_e \mathbf{v}'_e, \quad \mathbf{m}_e := \int_{\Omega^e} \rho \mathbf{N}_e^T \mathbf{N}_e da, \quad (71)$$

and convective part

$$\mathbf{f}_{\text{conv}}^e := \int_{\Omega^e} \rho \mathbf{N}_e^T \mathbf{v}_{,\alpha} u^\alpha da. \quad (72)$$

Here $u^\alpha = \dot{\zeta}^\alpha$ and $\mathbf{v}_{,\alpha}$ follow from (58) and (60). The transient part is linear in $\mathbf{v}'$. The mass matrix $\mathbf{m}_e$ is only constant as long as the mesh is fixed or described in a Lagrangian frame. If the ALE frame is used, the mass can change, and hence $\mathbf{f}_{\text{trans}}^e$ needs to be linearized. Its linearization, along with that of $\mathbf{f}_{\text{conv}}^e$, is given in Appendix A.2.1.

The discretization of G_{int} leads to the internal FE force vector (Sauer et al., 2014)

$$\mathbf{f}_{\text{int}}^e := \int_{\Omega^e} \mathbf{N}_{e,\alpha}^T \sigma^{\alpha\beta} \mathbf{a}_\beta da. \quad (73)$$

In view of (28), (25) and (64), this can be written as

$$\mathbf{f}_{\text{int}}^e = \mathbf{f}_{\text{int}q}^e + \mathbf{f}_{\text{int}v}^e \quad (74)$$

with

$$\begin{aligned} \mathbf{f}_{\text{int}q}^e &:= \int_{\Omega^e} \mathbf{D}_e^T q da, \\ \mathbf{f}_{\text{int}v}^e &:= \int_{\Omega^e} \eta \mathbf{N}_{e,\alpha}^T \mathbf{a}_s^{\alpha\beta} \mathbf{v}_{,\beta} da. \end{aligned} \quad (75)$$

Their linearization is given in Appendix A.2.2.

The discretization of G_{ext} leads to the two contributions

$$\begin{aligned} \mathbf{f}_f^e &:= \int_{\Omega^e} \mathbf{N}_e^T \mathbf{f} da, \\ \mathbf{f}_T^e &:= \int_{\partial_T \Omega^e} \mathbf{N}_e^T \bar{\mathbf{T}} ds, \end{aligned} \quad (76)$$

due to external surface loads $\mathbf{f}$ within the domain and boundary tractions $\bar{\mathbf{T}}$ on Neumann boundaries. The numerical examples in Secs. 5-6 either prescribe a given manufactured body force $\mathbf{f}$, or prescribe pressure loading in the form $\mathbf{f} = p \mathbf{n}$. The linearization of the latter case is given in Appendix A.2.3. The examples in Sec. 5 all have closed surfaces, such that there are no traction boundaries ($\partial_T \Omega^e = \emptyset$), but the examples in Sec. 6 contain a Neumann boundary, where deformation dependent tractions are prescribed as described in Appendix A.2.3.

Remark 4.4: $\mathbf{f}_{\text{int}v}^e$ captures the in-plane resistance of the flow due to in-plane viscosity parameter η . It can be advantageous to stabilize the surface motion by the out-of-plane resistance

$$p_{\text{visc}} := -\eta_n \mathbf{n} \cdot \mathbf{v}, \quad (77)$$

where $\eta_n \geq 0$ physically corresponds to an out-of-plane viscosity parameter. Without (77), the out-of-plane surface motion is undamped and can become unbounded, which has been observed in the steady flow example of Sec. 6.1.2 and the transient inflation example of Fig. 27. The pressure p_{visc} can be simply added to $\mathbf{f} = p \mathbf{n}$ within Eq. (76). Its linearization is given in Sauer (2016), see also Appendix A.2.3.

4.2.2 Area-incompressibility constraint

The discretization of the second weak form in Box 1 leads to the elemental pseudo force vector¹

$$\bar{\mathbf{f}}^e = \bar{\mathbf{f}}_{\text{div}}^e - \bar{\mathbf{f}}_{\text{DB}}^e, \quad (78)$$

with

$$\begin{aligned} \bar{\mathbf{f}}_{\text{div}}^e &:= \int_{\Omega^e} \mathbf{L}_e^T \text{div}_s \mathbf{v} \, da, \\ \bar{\mathbf{f}}_{\text{DB}}^e &:= \frac{\alpha_{\text{DB}}}{\eta} \int_{\Omega^e} \mathbf{L}_e^T (q - \check{q}) \, da, \end{aligned} \quad (79)$$

where $\text{div}_s \mathbf{v}$, q and $\check{q}$ are discretized by Eqs. (63), (67) and (68). Further, the discretization of the third weak form in Box 1 leads to

$$\check{\mathbf{f}}^e := \frac{\alpha_{\text{DB}}}{\eta} \int_{\Omega^e} \check{\mathbf{L}}_e (q - \check{q}) \, da. \quad (80)$$

Since $\check{\mathbf{L}}_e$ is defined locally, the third weak form leads to $\check{\mathbf{f}}^e = \mathbf{0}$, which can be used to eliminate $\check{\mathbf{q}}_e$ locally: Inserting (67) and (68) into $\check{\mathbf{f}}^e = \mathbf{0}$ and solving for $\check{\mathbf{q}}_e$ then gives

$$\check{\mathbf{q}}_e = \mathbf{h}_e^{-1} \mathbf{g}_e \mathbf{q}_e, \quad (81)$$

with

$$\begin{aligned} \mathbf{g}_e &:= \int_{\Omega^e} \check{\mathbf{L}}_e^T \mathbf{L}_e \, da, \\ \mathbf{h}_e &:= \int_{\Omega^e} \check{\mathbf{L}}_e^T \check{\mathbf{L}}_e \, da. \end{aligned} \quad (82)$$

Inserting this together with (67) and (68) into (79.2) then gives

$$\bar{\mathbf{f}}_{\text{DB}}^e = \bar{\mathbf{d}}_e \mathbf{q}_e, \quad (83)$$

with

$$\begin{aligned} \bar{\mathbf{d}}_e &:= \frac{\alpha_{\text{DB}}}{\eta} [\bar{\mathbf{m}}_e - \mathbf{g}_e^T \mathbf{h}_e^{-1} \mathbf{g}_e], \\ \bar{\mathbf{m}}_e &:= \int_{\Omega^e} \mathbf{L}_e^T \mathbf{L}_e \, da. \end{aligned} \quad (84)$$

The linearization of Eq. (78) based on this elimination is provided in Appendix A.3. In the examples of Sec. 5-6.2 the integration of $\bar{\mathbf{m}}_e$, $\mathbf{g}_e$ and $\mathbf{h}_e$ is chosen to be taken over dA , instead of $da = J_m dA$, in order to simplify the linearization. This is justified, as J_m is very close to one in those examples. In some of the examples of Sec. 6.3, on the other hand, surface inflow leads to large J_m , and integrating over da or dA makes a difference as is shown.

¹Its unit is $[\text{m}^2/\text{s}]$.

4.2.3 Mesh equations

The discretization of the fourth weak form in Box 1 leads to the following elemental pseudo force vector.² For Eulerian mesh motion (option I), this vector becomes

$$\tilde{\mathbf{f}}^e := \hat{\mathbf{m}}_e \mathbf{x}'_e - \hat{\mathbf{n}}_e \mathbf{v}_e, \quad (85)$$

with the constant matrix

$$\tilde{\mathbf{m}}_e := \alpha_m \int_{\Omega_0^e} \mathbf{N}_e^T \mathbf{N}_e \, dA, \quad (86)$$

and the surface-dependent matrix

$$\tilde{\mathbf{n}}_e := \alpha_m \int_{\Omega_0^e} \mathbf{N}_e^T (\mathbf{n} \otimes \mathbf{n}) \mathbf{N}_e \, dA. \quad (87)$$

This follows from inserting the elemental interpolations (51), (56) and (70.2) into $\tilde{G}_0$ of Box 1. For elastic mesh motion (option II), the elemental pseudo force vector becomes

$$\tilde{\mathbf{f}}^e := \tilde{\mathbf{f}}_i^e + \tilde{\mathbf{f}}_o^e, \quad (88)$$

with

$$\begin{aligned} \tilde{\mathbf{f}}_i^e &:= \int_{\Omega_0^e} [\mathbf{N}_{e,\alpha}^T \mathbf{a}_\beta + \mathbf{N}_e^T \mathbf{n} b_{\alpha\beta}] \tau_m^{\alpha\beta} \, dA, \\ \tilde{\mathbf{f}}_o^e &:= \alpha_m \tilde{\mathbf{n}}_e (\mathbf{x}'_e - \mathbf{v}_e). \end{aligned} \quad (89)$$

This follows from inserting $w_{\alpha;\beta} = \mathbf{w}_e^T [\mathbf{N}_{e,\alpha}^T \mathbf{a}_\beta + \mathbf{N}_e^T \mathbf{n} b_{\alpha\beta}]$ (Sauer et al., 2014), $w = \mathbf{w} \cdot \mathbf{n}$, (51), (56) and (70.2) into $\tilde{G}_{el}$ of Box 1. The in-plane part $\tilde{\mathbf{f}}_i^e$ was already used in Sauer (2014) to stabilize quasi-static droplet simulations. Here, the stabilization stress $\tau_m^{\alpha\beta}$ is taken from the simple elasticity model in (33).

The linearization of both options is provided in Appendix A.4. It simplifies due to the fact that all the above contributions are integrated over the initial configuration. This integration simplification is justified as the mesh contributions all appear in a separate weak form equation. This is different for the stabilization contributions of Sec. 4.2.2: They are combined with the incompressibility constraint in Eq. (78).

4.2.4 Numerical quadrature

All integrals are mapped to ζ^α -space and evaluated there by a numerical quadrature rule. The mapping follows from the expressions in (9). Gaussian quadrature with 3×3 quadrature points is used for all quadratic elements.

4.2.5 Coupled system of equations

The system of equations to be solved – after element assembly and elimination of Dirichlet boundary conditions – is

$$\begin{aligned} \mathbf{f}(\mathbf{v}', \mathbf{v}, \mathbf{x}', \mathbf{x}, \mathbf{q}) &:= \mathbf{f}_{in}(\mathbf{v}', \mathbf{v}, \mathbf{x}', \mathbf{x}) + \mathbf{f}_{int}(\mathbf{v}, \mathbf{x}, \mathbf{q}) - \mathbf{f}_{ext}(\mathbf{x}) = \mathbf{0}, \\ \tilde{\mathbf{f}}(\mathbf{v}, \mathbf{x}', \mathbf{x}) &:= \tilde{\mathbf{f}}_i(\mathbf{v}, \mathbf{x}', \mathbf{x}) + \tilde{\mathbf{f}}_o(\mathbf{v}, \mathbf{x}', \mathbf{x}) = \mathbf{0}, \\ \bar{\mathbf{f}}(\mathbf{v}, \mathbf{x}, \mathbf{q}) &:= \bar{\mathbf{f}}_{div}(\mathbf{v}, \mathbf{x}) - \bar{\mathbf{f}}_{DB}(\mathbf{q}) = \mathbf{0}. \end{aligned} \quad (90)$$

²Its unit is $[\text{m}^3/\text{s}]$.

These are the governing ordinary differential equations for the unknown nodal flow variables $\mathbf{v} = [\mathbf{v}_I]$, $\mathbf{q} = [q_I]$ and surface position $\mathbf{x} = [\mathbf{x}_I]$. In order to solve them, time integration needs to be used. If the mesh motion $\mathbf{x}(t)$ is known (e.g. prescribed) the second equation in (90) becomes unnecessary.

4.3 Time integration

Using $\mathbf{a} := \mathbf{v}'$ and $\mathbf{v}_m = \mathbf{x}'$ the system of coupled ODEs (90) can be written as

$$\begin{aligned} \mathbf{f}_v &:= \mathbf{f}(\mathbf{a}, \mathbf{v}, \mathbf{v}_m, \mathbf{x}, \mathbf{q}) = \mathbf{0}, \\ \mathbf{f}_x &:= \tilde{\mathbf{f}}(\mathbf{v}, \mathbf{v}_m, \mathbf{x}) = \mathbf{0}, \\ \mathbf{f}_q &:= \bar{\mathbf{f}}(\mathbf{v}, \mathbf{x}, \mathbf{q}) = \mathbf{0}. \end{aligned} \tag{91}$$

Here, $\mathbf{f}_v = \mathbf{0}$ is a first order ODE, predominately for $\mathbf{v}$, $\mathbf{f}_x = \mathbf{0}$ is a first order ODE, predominantly for $\mathbf{x}$, and $\mathbf{f}_q = \mathbf{0}$ is an algebraic equation, predominantly for $\mathbf{q}$. Only in the Lagrangian case (where $\mathbf{v}_m = \mathbf{v}$ eliminates the second equation) does $\mathbf{f}_v = \mathbf{0}$ become a second order ODE. System (91) is discretized in time by classical time stepping: Given $\mathbf{a}_n$, $\mathbf{v}_n$, $\mathbf{v}_m^n$, $\mathbf{x}_n$ and $\mathbf{q}_n$ at time step t_n the new values $\mathbf{a}_{n+1}$, $\mathbf{v}_{n+1}$, $\mathbf{v}_m^{n+1}$, $\mathbf{x}_{n+1}$ and $\mathbf{q}_{n+1}$ at t_{n+1} are determined. Since these are five unknowns, two extra equations are needed. These are provided through finite difference approximations of $\mathbf{v}' = \mathbf{a}$ and $\mathbf{x}' = \mathbf{v}_m$, in the form

$$\begin{aligned} \mathbf{v}_{n+1} &= \mathbf{v}_n + \Delta t_{n+1} ((1 - \gamma) \mathbf{a}_n + \gamma \mathbf{a}_{n+1}), \\ \mathbf{x}_{n+1} &= \mathbf{x}_n + \Delta t_{n+1} ((1 - \gamma) \mathbf{v}_m^n + \gamma \mathbf{v}_m^{n+1}). \end{aligned} \tag{92}$$

Taking $\gamma = 0$ and $\gamma = 1$ here, gives the classical explicit and implicit Euler scheme, respectively. However those are only 1st order accurate in time. The choice $\gamma = 1/2$ corresponds to the trapezoidal rule, which is used in all subsequent examples, since it is 2nd order accurate in time and implicit, making it suitable for strongly coupled systems.

The time integration can be easily extended to the generalized- α scheme (Chung and Hulbert, 1993; Jansen et al., 2000). This is shown in Appendix B even if the following examples do not use this.

4.4 Solution procedure

Using Eq. (92) to eliminate $\mathbf{a}_{n+1}$ and $\mathbf{v}_m^{n+1}$ gives

$$\begin{aligned} \mathbf{f}_{n+1} &:= \mathbf{f}(\mathbf{v}_{n+1}, \mathbf{x}_{n+1}, \mathbf{q}_{n+1}) = \mathbf{0}, \\ \tilde{\mathbf{f}}_{n+1} &:= \tilde{\mathbf{f}}(\mathbf{v}_{n+1}, \mathbf{x}_{n+1}) = \mathbf{0}, \\ \bar{\mathbf{f}}_{n+1} &:= \bar{\mathbf{f}}(\mathbf{v}_{n+1}, \mathbf{x}_{n+1}, \mathbf{q}_{n+1}) = \mathbf{0}, \end{aligned} \tag{93}$$

or more compactly,

$$\mathbf{r}_{n+1}(\mathbf{u}_{n+1}) = \mathbf{0}, \tag{94}$$

with the vector of unknowns and the residual vector defined by

$$\mathbf{u} := \begin{bmatrix} \mathbf{v} \\ \mathbf{x} \\ \mathbf{q} \end{bmatrix}, \quad \mathbf{r} := \begin{bmatrix} \mathbf{f} \\ \tilde{\mathbf{f}} \\ \bar{\mathbf{f}} \end{bmatrix}. \tag{95}$$

The nonlinear equation system (94) is solved with the Newton-Raphson method. It is based on the linearization of (94), which leads to the iterative solution of $\mathbf{K}_n \Delta \mathbf{u}_{n+1} = -\mathbf{r}_n$, where

$$\mathbf{K}_n := \begin{bmatrix} \frac{\partial \mathbf{f}_n}{\partial \mathbf{v}_n} & \frac{\partial \mathbf{f}_n}{\partial \mathbf{x}_n} & \frac{\partial \mathbf{f}_n}{\partial \mathbf{q}_n} \\ \frac{\partial \tilde{\mathbf{f}}_n}{\partial \mathbf{v}_n} & \frac{\partial \tilde{\mathbf{f}}_n}{\partial \mathbf{x}_n} & \frac{\partial \tilde{\mathbf{f}}_n}{\partial \mathbf{q}_n} \\ \frac{\partial \bar{\mathbf{f}}_n}{\partial \mathbf{v}_n} & \frac{\partial \bar{\mathbf{f}}_n}{\partial \mathbf{x}_n} & \frac{\partial \bar{\mathbf{f}}_n}{\partial \mathbf{q}_n} \end{bmatrix} =: \begin{bmatrix} \mathbf{K}_{vv} & \mathbf{K}_{vx} & \mathbf{K}_{vq} \\ \mathbf{K}_{xv} & \mathbf{K}_{xx} & \mathbf{K}_{xq} \\ \mathbf{K}_{qv} & \mathbf{K}_{qx} & \mathbf{K}_{qq} \end{bmatrix}_n \quad (96)$$

is the tangent matrix at t_n . Obviously $\mathbf{K}_{xq}^n = \partial \tilde{\mathbf{f}}_n / \partial \mathbf{q}_n = \mathbf{0}$. The remaining eight blocks of $\mathbf{K}_n$ are given in Appendix B.

4.5 Normalization

The above equations are implemented in normalized form. A length scale L_0 , time scale T_0 and force F_0 are used for this. All lengths, times, forces and combined units are then expressed in terms of these quantities. Pressures, for example are then multiples of $p_0 = F_0/L_0^2$. The examples in Sec. 5-6.2 use unit values for all quantities (i.e. $L_0 = T_0 = F_0 = 1$), while the example in Sec. 6.3 uses $L_0 = 1\text{mm}$, $T_0 = 1\text{ms}$ and $p_0 = 1\text{Pa}$.

5 Numerical examples for fixed surface flows

The following two sections present several numerical examples, first for fixed surfaces (Sec. 5), then deforming surfaces (Sec. 6). All examples are full Navier-Stokes examples – they contain the nonlinear convective part. Their complexity is increased successively in order to test the computational formulation thoroughly. They use spheres as initial configuration, as these are stable configurations of free fluid films. Other shapes, such as cylinders, are unstable beyond certain lengths (Sauer, 2014; Sahu et al., 2020) and not considered here.

5.1 Problem setup

Sec. 5 studies two simple verification examples for rotational flow on fixed spheres. They have manufactured analytical solutions, see Sauer (2025). The comparison between numerical and analytical solution allows to examine convergence rates and verify both FE implementation and manufactured solution.

For fixed surfaces the out-of-plane velocity v_n is given, e.g. as zero. It makes a difference if this is explicitly enforced or not, which is illustrated below. Correspondingly, two different out-of-plane boundary conditions are considered: (i) imposing zero out-of-plane velocity (Dirichlet BC), or (ii) imposing the surface pressure corresponding to zero v_n (Neumann BC). Both cases are complemented with essential BCs to fix all six rigid body modes, as well as the datum value of the surface tension q . The first BC is imposed by local projection (the dofs of each node are first transformed into the local spherical coordinate system and then the normal component is eliminated).

Since no curvature-depending forces are present in the considered examples, classical C^0 -continuous finite elements are sufficient. In all subsequent examples quadrilateral bi-quadratic Lagrange FE are used. The spheres are meshed with six square patches, containing $2m \times 2m$ elements each. Thus, $n_{\text{el}} = 24m^2$ is the total number of elements of the sphere and $n_{\text{eq}} = 8m$ is the number of elements around the equator. Further, the element size h is proportional to m^{-1} and n_{el}^{-2} . The sequence of meshes used here is shown in Table 1. They are used to examine the

m	n_{el}	h_{eq}/r	n_{no}	3-dof	4-dof	7-dof	$n_{t1}/4$	$n_{t2}/4$
1	24	$\pi/4$	98	294	392	686	4	2
2	96	$\pi/8$	386	1,158	1,544	2,702	8	6
4	384	$\pi/16$	1,538	4,614	6,152	10,766	16	16
8	1,536	$\pi/32$	6,146	18,438	24,584	43,022	32	45
16	6,144	$\pi/64$	24,578	73,734	98,312	172,046	64	128
32	24,576	$\pi/128$	98,306	294,918	393,224	688,142	128	362

Table 1: Shear flow on spheres: Considered FE meshes based on bi-quadratic Lagrange elements and corresponding dofs for three (v^α, q), four ($\mathbf{v}, q$) and seven ($\mathbf{v}, q, \mathbf{v}_m$) unknowns. Three and four unknowns are used in the examples of Sec. 5, seven in the examples of Sec. 6. $h_{\text{eq}} = 2\pi r/(8m)$ is the element length at the equator; $n_{ti}/4$ are the number of time steps per quarter period used in ALE case d (see Table 2).

algebraic L^2 error measure

$$e(\mathbf{u}) := \frac{\|\mathbf{u}_{\text{FE}} - \mathbf{u}_{\text{ana}}\|}{\|\mathbf{u}_{\text{ana}}\|} \quad (97)$$

applied to the nodal values of $\mathbf{v}$, $\gamma = q$, ω and p . The latter two are obtained by post-processing.

Remark 5.1: All the following convergence figures plot the error against n_{el} on a logscale. Plotted against h , simply makes all slopes twice as steep, since h scales with n_{el}^{-2} .

5.2 Simple shear flow

The first example in Sec. 5 considers the axisymmetric flow field

$$\mathbf{v} = r\omega_0 \sin \theta \cos \theta \mathbf{e}_\phi, \quad (98)$$

on the sphere ($0 < \phi < 2\pi$ and $-\pi/2 < \theta < \pi/2$). Here, ϕ and θ are the azimuthal and elevation angle, respectively, and $\mathbf{e}_\phi$ the unit basis vector along ϕ ; ω_0 is constant so that the flow is steady. Mesh velocity $\mathbf{v}_m$ is considered given, such that no ALE ODE needs to be solved. The four different ALE cases shown in Table 2 are studied. They are combined with

	load case	1	2	3	4
ALE case		$f_2 = 0$ DBC $v_n = 0^*$	$q = \text{const.}$ DBC $v_n = 0^*$	$f_2 = 0$ NBC $p = \bar{p}$	$q = \text{const.}$, NBC $p = \bar{p}$
a	$\mathbf{v}_m = \mathbf{0}, \quad \theta_0 = 0$	Fig. 2 & 3	Fig. 29	Fig. 7	Fig. 30
b	$\mathbf{v}_m = \mathbf{c}_0, \quad \theta_0 = 0$	Fig. 4	Fig. 29	Fig. 7	Fig. 30
c	$\mathbf{v}_m = \mathbf{0}, \quad \theta_0 = 0.5$	Fig. 5	Fig. 29	Fig. 7	Fig. 30
d	$\mathbf{v}_m = \mathbf{v}_m(t), \quad \theta_0 = 0.5$	Fig. 6	Fig. 29	Fig. 7	Fig. 30

Table 2: Simple shear flow on a rigid sphere: Considered ALE mesh motion cases and surface load cases. The ALE cases consider either no mesh motion and distortion (case a), the constant mesh velocity of Eq. (100) (case b), the fixed mesh distortion of Eq. (101) (case c), or the transient mesh motion of (101) & (103) (case d). The load cases consider different body forces, and different out-of-plane boundary conditions (DBC = Dirichlet boundary condition, NBC = Neumann boundary condition). Load cases 1 & 2 have three unknowns (v^α, q), while load cases 3 & 4 have four ($\mathbf{v}, q$). *In ALE case b, $v_n = \mathbf{c}_0 \cdot \mathbf{n} \neq 0$.

four different load cases, and discussed in the following four sub-sub-sections. Load cases 3 & 4 have four unknowns ($\mathbf{v}, q$). For load cases 1 & 2, the normal velocity v_n is eliminated by projection along the surface normal, leaving only three unknowns (v^α, q). Flow (98) is caused by the body force field (Sauer, 2025)

$$\mathbf{f} = f_\alpha \mathbf{a}^\alpha + p \mathbf{n}, \quad \begin{cases} f_1 = 4\eta\omega_0 \sin \theta \cos^2 \theta, \\ f_2 = \rho r^2 \omega_0^2 \sin^3 \theta \cos \theta - q_{,2}, \end{cases} \quad (99)$$

where surface tension q is picked such that either $f_2 = 0$ or $q_{,2} = 0$, and surface pressure p either follows as the reaction to the Dirichlet boundary condition on v_n or is prescribed by the Neumann boundary condition $p = \bar{p}$, see Table 2. Unit values are used for r , ω_0 , η and ρ , while $\eta_n = 0$ in all cases of Secs. 5.2 and 5.3.

5.2.1 Zero mesh velocity (ALE case a)

The first ALE case considers no mesh motion and no mesh distortion ($\mathbf{v}_m = \mathbf{0}$), i.e. the surface parameterization based on ϕ and θ is fixed in time. Fig. 2a shows the surface velocity for this

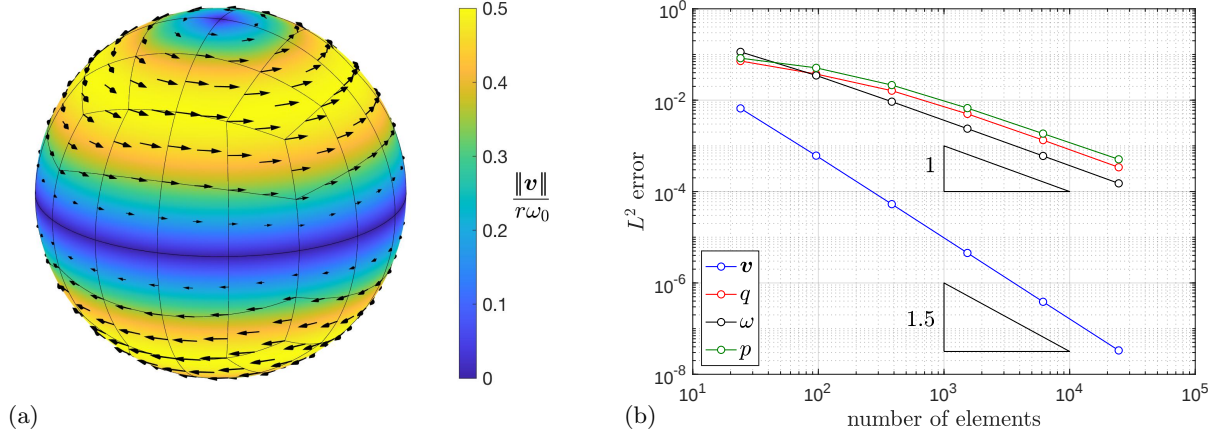


Figure 2: Simple shear flow on a rigid sphere: (a) Flow field $\mathbf{v}$ and (b) convergence behavior for ALE case a and load case 1. Optimal convergence rates are obtained for all fields: At least $O(h^3) = O(n_{\text{el}}^{-1.5})$ for $\mathbf{v}$, and $O(h^2) = O(n_{\text{el}}^{-1})$ for q , p and ω .

case. It is caused by a manufactured body force that admits two special cases:

Load case 1: Picking $f_2 = 0$ (zero body force along elevation angle θ) leads to varying surface tension q , as shown in Fig. 3a. Fig. 3 also shows the vorticity ω and the outward surface

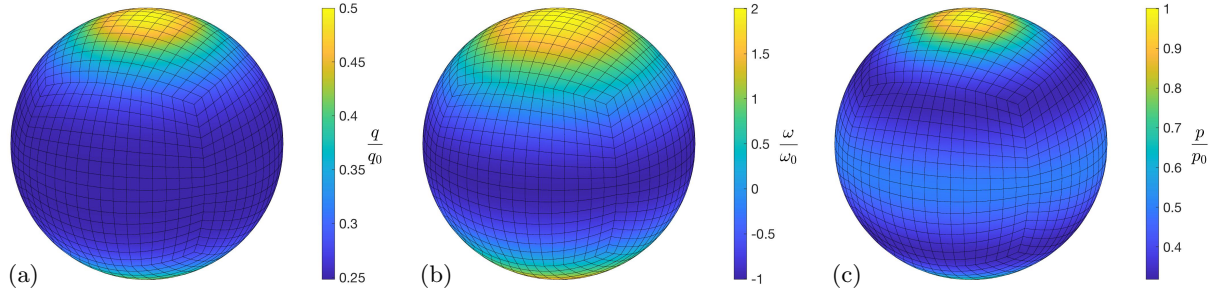


Figure 3: Simple shear flow on a rigid sphere: (a) Surface tension q , (b) surface vorticity ω and (c) lateral surface pressure p for ALE case a and load case 1. The plots are normalized by $q_0 = \rho r^2 \omega_0^2$, w_0 and $p_0 = \rho r \omega_0^2$.

pressure p required to hold the surface in place (i.e. the lateral boundary reactions). A much finer mesh is needed for q , ω and p to yield comparable accuracy than for $\mathbf{v}$. The flow field is invariant w.r.t. to a chosen datum pressure. Here this is fixed by specifying the pole pressure as $p_p = p_0 = \rho r \omega_0^2$. As shown in Fig. 2b, the FE convergence rates for $\mathbf{v}$, q , ω and p are optimal (or better): $O(h^3) = O(n_{\text{el}}^{-1.5})$ for $\mathbf{v}$, and $O(h^2) = O(n_{\text{el}}^{-1})$ for q according to [Dohrmann and Bochev \(2004\)](#).

Load case 2: Picking f_2 such that q becomes constant, see [Sauer \(2025\)](#), leads to different p , but the same $\mathbf{v}$ and ω as before. The convergence rates for load case 2 are also optimal, as Fig. 29a in Appendix C shows.

So far the mesh motion was zero. The next three ALE cases consider three different prescribed mesh motions/distortions.

Remark 5.2: The solution above contains surface Stokes flow for the special choice $\rho = 0$. For Stokes flow, both q and p remain constant in both load cases, see [Sauer \(2025\)](#).

5.2.2 Constant mesh velocity (ALE case b)

Now the constant mesh velocity

$$\mathbf{v}_m = \mathbf{c}_0 \quad (100)$$

is considered, where the constant $\mathbf{c}_0$ is chosen as $\mathbf{c}_0 = v_{m0}(\mathbf{e}_1 + \mathbf{e}_2 + \mathbf{e}_3)/\sqrt{3}$ with $v_{m0} = r\omega_0/2$. Thus $\mathbf{v}$ has out-of-plane components, as Fig. 4a shows. This $\mathbf{v}_m$ represents a constant translation

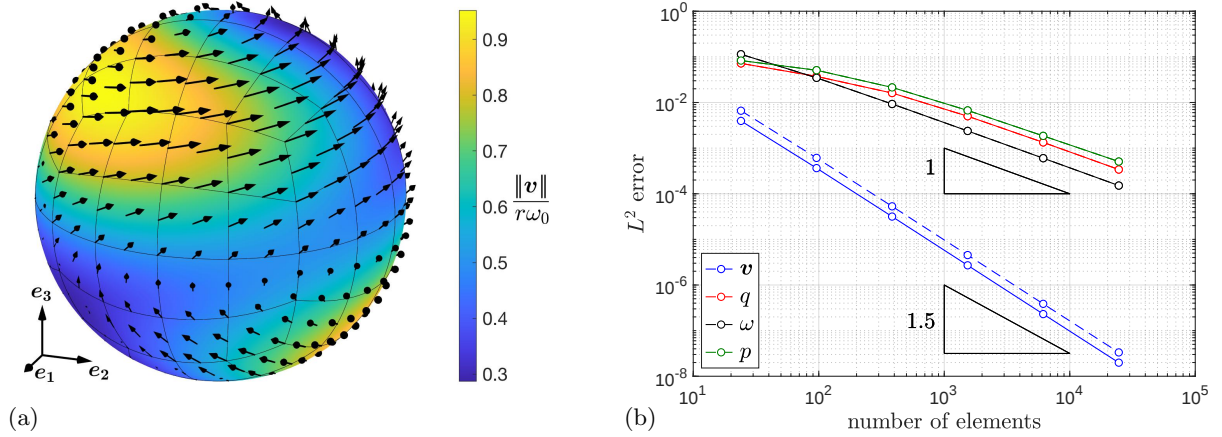


Figure 4: Simple shear flow on a rigid sphere: (a) Flow field $\mathbf{v}$ and (b) convergence behavior for the prescribed constant mesh velocity $\mathbf{v}_m = \mathbf{c}_0$ given in (100) (ALE case b, load case 1). In this case, $\mathbf{v}_m$, and hence also $\mathbf{v}$, have out-of-plane components. As in Fig. 2, optimal convergence rates are obtained for all fields. The dashed line in (b) is the line from Fig. 2b.

that tests all three components in $\mathbb{R}^3$. The case works well – the optimal convergence rates are maintained as Fig. 4b shows. Load case 2 behaves similarly, as Fig. 29b in Appendix C shows: Again optimal convergence rates are obtained.

5.2.3 Fixed mesh distortion (ALE case c)

Next, the fixed mesh distortion

$$\mathbf{x} = r \mathbf{e}_r(\theta), \quad (101)$$

with

$$\theta = \Theta + \theta_0 \sin \phi \cos^2 \Theta, \quad (102)$$

is considered. It leads to $\mathbf{v}_m = \mathbf{0}$ and is shown in Fig. 5a for $\theta_0 = 1/2$ and load case 1. Also this case works well – the optimal convergence rates are maintained as Fig. 5b shows. They are compared here to the ones in Fig. 2, which was the case of no mesh distortion ($\theta_0 = 0$). Load case 2 behaves similarly, as Fig. 29c in Appendix C shows: Again optimal convergence rates are obtained.

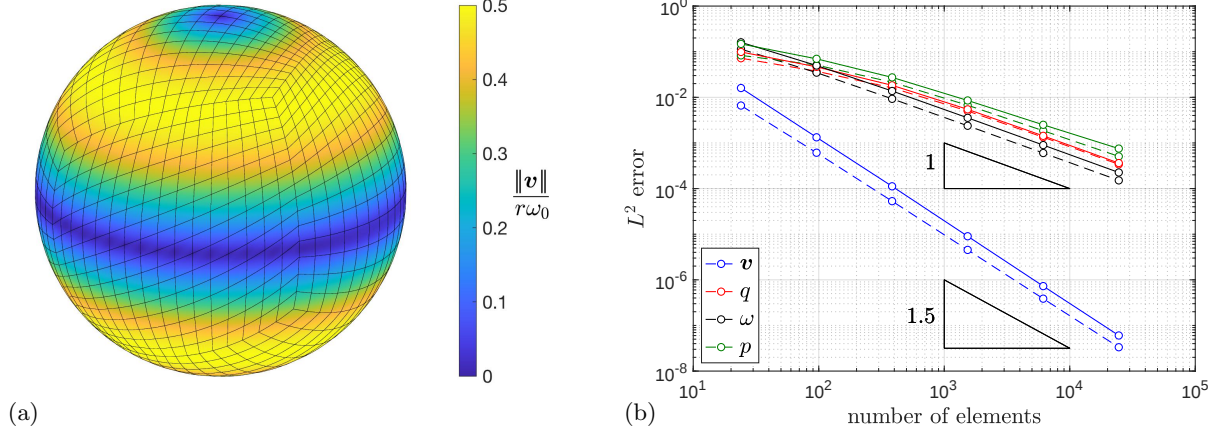


Figure 5: Simple shear flow on a rigid sphere: (a) Flow field $\mathbf{v}$ and (b) convergence behavior due to the prescribed fixed mesh distortion $\mathbf{x}_m$ given in (101) & (102) (ALE case c, load case 1). Again, optimal convergence rates are obtained for all fields. The dashed lines in (b) are the lines from Fig. 2b.

5.2.4 Varying mesh velocity (ALE case d)

Next, the transient mesh motion

$$\theta(t) = \Theta + \theta_0 \cos(\omega_m t) \sin \phi \cos^2 \Theta \quad (103)$$

is used inside Eq. (101). It is shown at quarter intervals of the period $T = 2\pi/\omega_m$ in Fig. 6a for $\theta_0 = 1/2$ and $\omega_m = \omega_0$ (the parameter from Eq. (98)). Since $\mathbf{e}_{r,\theta} = \mathbf{e}_\theta$ (see Sauer (2025)),

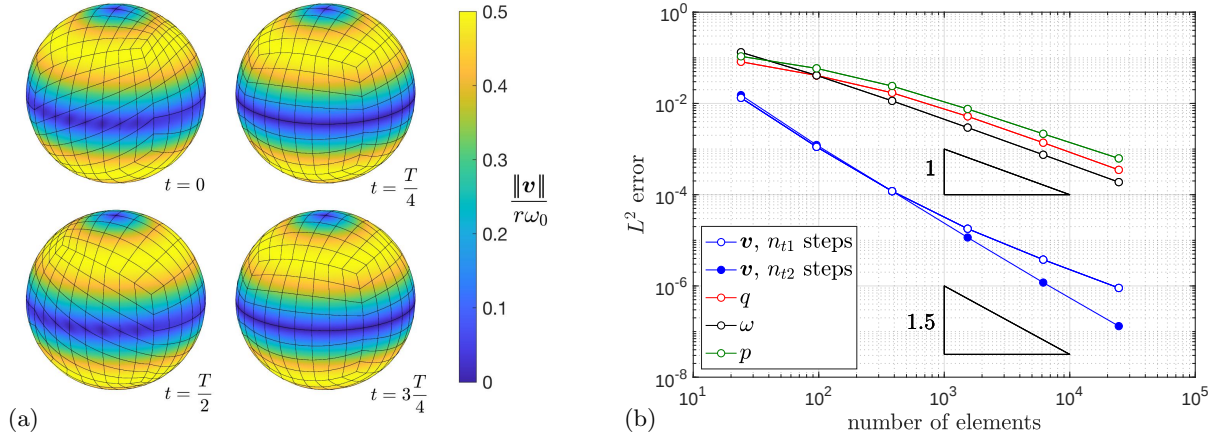


Figure 6: Simple shear flow on a rigid sphere: (a) Flow field $\mathbf{v}$ and (b) convergence behavior due to the prescribed transient mesh motion $\mathbf{x}_m(t)$ given in (101) & (103) (ALE case d, load case 1). Again, optimal convergence rates are obtained if sufficiently many time steps are used. Here, the L^2 error from Eq. (97) is averaged over $n_t/4$ time steps.

Appendix E.2), this mesh motion leads to the spatially and temporally varying mesh velocity

$$\mathbf{v}_m = \theta' r \mathbf{e}_\theta, \quad (104)$$

with

$$\theta'(t) = -\theta_0 \omega_m \sin(\omega_m t) \sin \phi \cos^2 \Theta, \quad (105)$$

that has no out-of-plane component. The maximum mesh velocity for $\theta_0 = 1/2$ is $v_{m0} = \omega_m r/2$. The convergence behavior of load case 1 with this superposed mesh motion is shown in Fig. 6b. The trapezoidal rule is used for time integration, which is second order accurate in time. If

the time step size Δt is refined proportionally to the element size, see n_{t1} in Table 1, the convergence rate in $\mathbf{v}$ is therefore only quadratic, i.e. $O(h^2) = O(n_{el}^{-1})$. In order to maintain $O(h^3)$ convergence, the time step needs to be proportional to $h^{1.5}$, i.e. proportional to $m^{-1.5}$. This is confirmed in Fig. 6b for the time step number n_{t2} given in Table 1.

Load case 2 behaves similarly, as Fig. 29d in Appendix C shows: Again optimal convergence rates are obtained.

Thus it is possible to maintain ideal convergence rates even for transient mesh motions. It is believed that this only works because in this example $\mathbf{v}'$ lies in the tangent plane: From Sauer (2025), Eq. (153), follows

$$\mathbf{v}' = \mathbf{v}_{,\alpha} v_m^\alpha = \theta' \mathbf{v}_{,2} = \theta' r \omega_0 (2 \cos^2 \theta - 1) \mathbf{e}_\phi \quad (106)$$

for the velocity in Eq. (98). This is different for the example of Sec. 5.3, where $\mathbf{v}'$ has out-of-plane components – even for a fixed surface – and where the convergence rate for $\mathbf{v}$ is only $O(h^2)$, see Fig. 12.

5.2.5 Unknown normal velocity v_n (Load cases 3 & 4)

The preceding results of Secs. 5.2.1–5.2.4 are for out-of-plane Dirichlet BC (load cases 1 & 2), where the out-of-plane flow velocity is prescribed through the given mesh velocity ($v_n = v_{mn}$).

Next we consider out-of-plane Neumann conditions, where the out-of-plane flow velocity v_n is unknown (load case 3 & 4, see Table 2). In this case the surface velocity is unconstrained and thus a full 3D vector that can have out-of-plane components in the iterative solution process, even for cases with solution $v_n = 0$, as is considered here. It is thus a test to check whether unknown v_n can be handled by the proposed numerical scheme.

The first surface pressure that leads to zero³ v_n is

$$\bar{p} = p_p + \rho r \omega_0^2 \left(\frac{3}{2} \sin^4 \theta - \sin^2 \theta - \frac{1}{2} \right), \quad p_p = \frac{2q_p}{r} \quad (107)$$

(Sauer, 2025), where $p_p = \bar{p}(\theta = \pm\pi/2)$ denotes the pole pressure. It is denoted load case 3. Here, $q_p = \rho r^2 \omega_0^2 / 2$ is chosen such that the pole pressure becomes $p_p = \rho r \omega_0^2$, while the equatorial pressure is $p_{eq} = \bar{p}(\theta = 0) = p_p / 2$. The pressure distribution of Eq. (107) is then equal to the one shown in Fig. 3c. The convergence rates for this load case are shown in Fig. 7 for the four ALE cases. The velocity rates are only $O(h^2)$ now, while the other rates are as before. It is believed this is due to the fact that the weak form for the unknown fluid velocity $\mathbf{v}$ depends on the surface derivatives $\mathbf{a}_\alpha = \mathbf{x}_{,\alpha}$, e.g. through $\mathbf{D}_e$ and $\mathbf{a}_s^{\alpha\beta}$ in $\mathbf{f}_{int}^e$, see Eqs. (74)–(75). Since the accuracy of surface $\mathbf{x}$ is $O(h^3)$, the surface derivatives $\mathbf{a}_\alpha$ are only $O(h^2)$, as differentiation reduces an order. (The $O(h^3)$ convergence rate of $\mathbf{x}$ can be seen in the example of Fig. 13.) For ALE case c, the time step Δt is refined proportionally to h so that all convergence rates are optimal, i.e. $O(h^2)$ for the trapezoidal rule (and the surface order reduction noted above).

The second surface pressure that leads to zero³ v_n is

$$\bar{p} = p_p - \rho r \omega_0^2 \sin^2 \theta \cos^2 \theta, \quad p_p = \frac{2q_p}{r} \quad (108)$$

(Sauer, 2025). It has constant surface tension across the surface and is denoted load case 4, see Table 2. Picking $q_p = \rho r^2 \omega_0^2 / 2$ leads to the polar and equatorial pressures $p_p = p_{eq} = \rho r \omega_0^2$. The convergence rates for this load case are very similar to the previous load case, as Fig. 30 shows. Again, convergence rates are optimal and the surface pressure error is at machine precision.

³or $v_n = v_{mn}$ in ALE case b, see Table 2

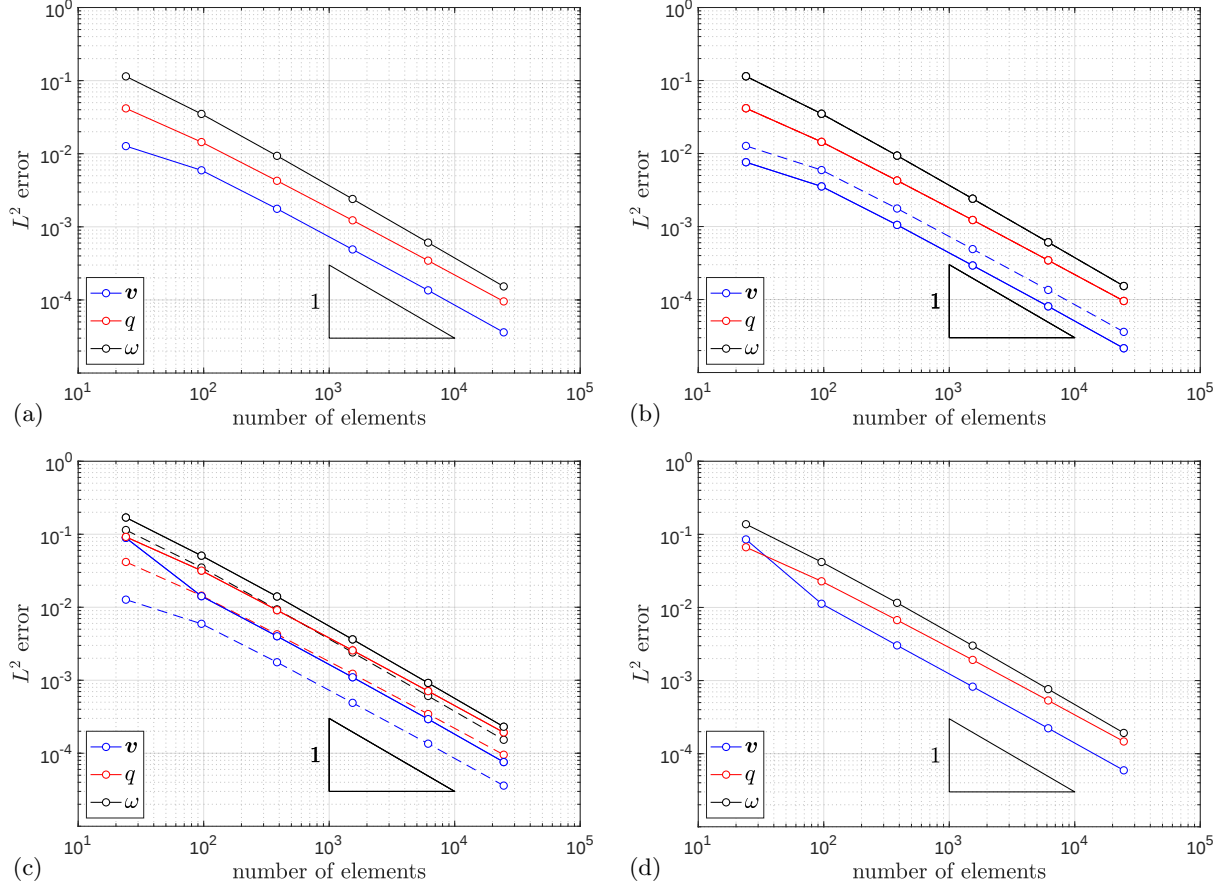


Figure 7: Simple shear flow on a rigid sphere for load case 3: Convergence of velocity, surface tension and vorticity for the four ALE cases (a-d) in the L^2 error norm. All convergence rates are $O(h^2) = O(n_{\text{el}}^{-1})$. The surface pressure error is at machine precision and hence not shown. The dashed lines in (b) & (c) are the results from (a).

5.3 Octahedral vortex flow

The second example in Sec. 5 considers surface flow defined by the stream function

$$\psi = v_0 r \sin 2\phi \sin \theta \cos^2 \theta. \quad (109)$$

The resulting velocity field $\mathbf{v}$ and required body force field are derived analytically in Sauer (2025) assuming constant surface tension q . Contrary to the preceding example, $\mathbf{v}$ is now non-axisymmetric and has components along both ϕ and θ , and those vary along both ϕ and θ . The same parametrization and meshes as before are used, see Table 1. Combining different boundary conditions and ALE cases leads to the seven cases shown in Table 3. For out-of-plane NBC (load case 4), v_n is an unknown that needs to be solved for, even if the applied surface pressure $\bar{p}$ is the one that gives $v_n = 0$. According to Sauer (2025), that pressure follows from the velocity field $\mathbf{v}$ as

$$\bar{p} = p_p - \frac{\rho}{r} \mathbf{v} \cdot \mathbf{v}, \quad (110)$$

where p_p is the pole pressure related to the constant surface tension q by the Young-Laplace equation $p_p = 2q/r$. The cases of Table 3 are examined in the following sub-sub-sections using a unit value for q .

	load case	2	4
ALE case		$q = \text{const.}$ DBC $v_n = 0$	$q = \text{const.}$, NBC $p = \bar{p}$
a	$\mathbf{v}_m = \mathbf{0}, \quad \theta_0 = 0$	Fig. 8	Fig. 8
b	$\mathbf{v}_m = \mathbf{c}_0, \quad \theta_0 = 0$	Fig. 10	Fig. 10
c	$\mathbf{v}_m = \mathbf{0}, \quad \theta_0 = 0.5$	Fig. 11	Fig. 11
d	$\mathbf{v}_m = \mathbf{v}_m(t), \quad \theta_0 = 0.5$	–	Fig. 12

Table 3: Octahedral vortex flow on a rigid sphere: Considered ALE mesh motion cases and surface load cases. The flow field $(\mathbf{v}, q)$ is the same for both load cases, but only in load case 4, v_n is treated unknown. It is nonzero in ALE case b. The ALE cases are the same as used in the previous example (see Sec. 5.2). DBC = Dirichlet boundary condition, NBC = Neumann boundary condition.

5.3.1 Zero mesh velocity (ALE case a)

The first ALE case considers no mesh motion and no mesh distortion. Fig. 8a shows the velocity field for this case. It is caused by an elaborate manufactured surface force (Sauer, 2025). Fig. 9

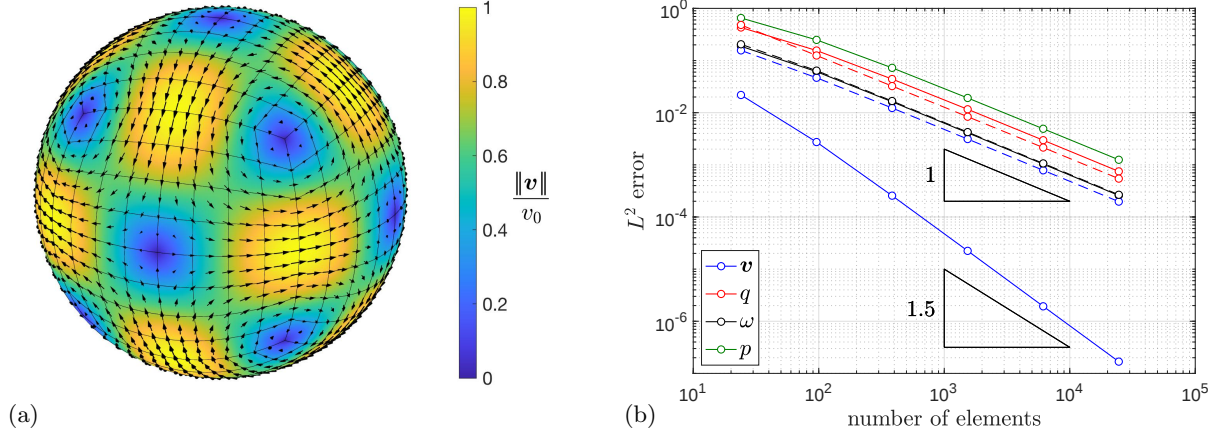


Figure 8: Octahedral vortex flow on a rigid sphere: (a) Flow field $\mathbf{v}$ and (b) convergence behavior for ALE case a. Solid lines are for load case 2; dashed lines for load case 4. (p is prescribed in the latter case and hence does not show up in the plot.) All convergence rates are optimal: At least $O(h^3) = O(n_{\text{el}}^{-1.5})$ for $\mathbf{v}$ in load case 2, and $O(h^2) = O(n_{\text{el}}^{-1})$ for all other fields. The convergence rate in $\mathbf{v}$ reduces to $O(h^2)$ for load case 4 due to its surface gradient dependency.

shows the vorticity ω and the outward surface pressure p required to hold the surface in place (p is the lateral boundary reaction for case 2, and the prescribed pressure for case 4; it is equal to $\bar{p}$ in (110)). A much finer mesh is needed for ω and p to be of comparable accuracy as $\mathbf{v}$, especially for load case 2. This can be seen from the convergence plots of Fig. 8b. The convergence rates are as in the previous example (see Sec. 5.2): For the Dirichlet case (load case 2, marked in solid lines) $\mathbf{v}$ converges at least cubically, while for the Neumann case (load case 4, marked dashed) $\mathbf{v}$ converges only quadratically due to the surface gradient dependency noted before. The direct comparison between the two load cases, shows that q is slightly more accurate for the Neumann case.

5.3.2 Constant mesh velocity (ALE case b)

The second ALE case (defined by the constant mesh velocity of Eq. (100)) has an out-of-plane velocity component. Fig. 10 shows the resulting flow field and its convergence rates. Again, only load case 2 attains at least cubic velocity convergence. Compared to ALE case a, the

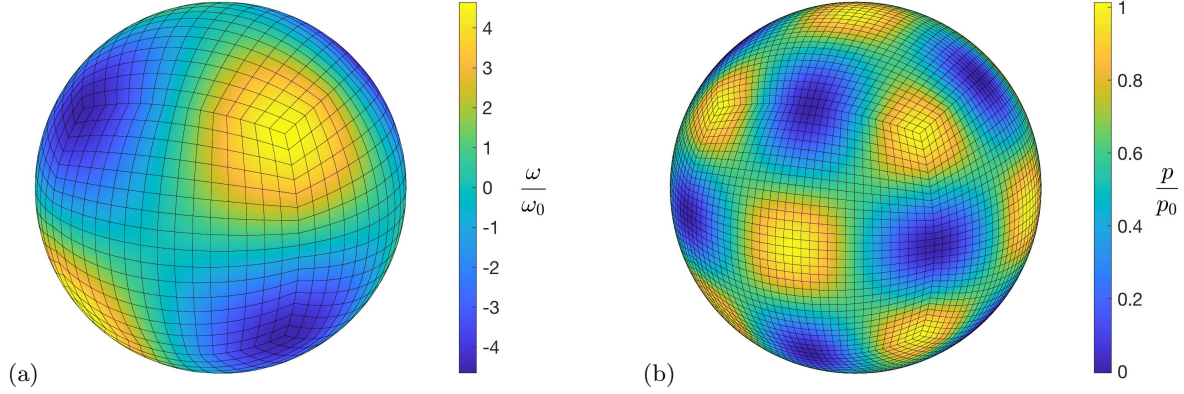


Figure 9: Octahedral vortex flow on a rigid sphere: (a) Vorticity ω and (b) lateral surface pressure p . A much finer mesh is required for ω and p to be of comparable accuracy as $\mathbf{v}$. Here $\omega_0 = v_0/r$ and $p_0 = p_p$.

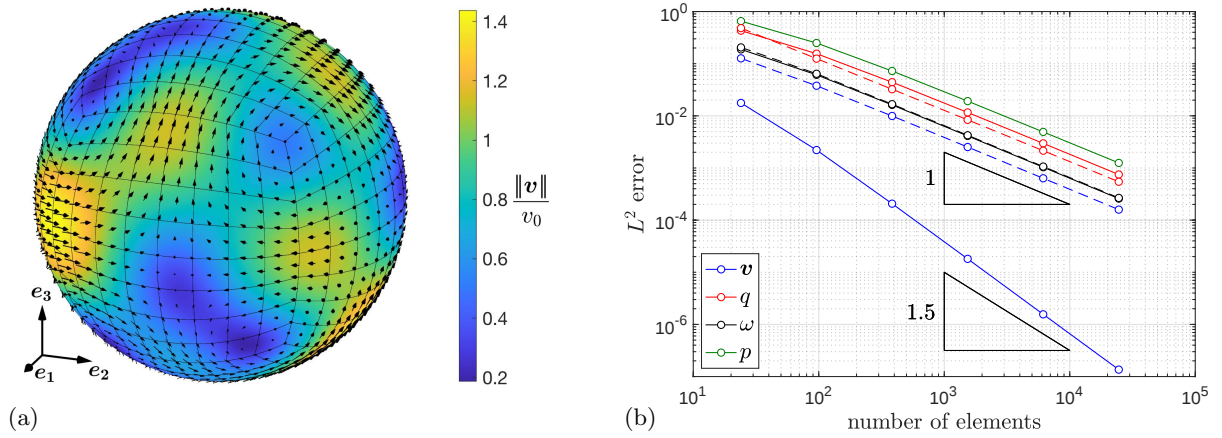


Figure 10: Octahedral vortex flow on a rigid sphere: (a) Flow field $\mathbf{v}$ and (b) convergence behavior due to the prescribed constant mesh velocity $\mathbf{v}_m = \mathbf{c}_0$ given in (100) (ALE case b). In this case, $\mathbf{v}_m$, and hence also $\mathbf{v}$, have out-of-plane components. Solid lines are for load case 2; dashed lines for load case 4. All convergence rates are optimal or better, as was also seen in Fig. 8.

error in fields q , ω and p is equal, while $\mathbf{v}$ has a slightly lower error now. The added constant translation thus slightly improves the relative accuracy. The same was observed in the example of Sec. 5.2.

5.3.3 Fixed mesh distortion (ALE case c)

The third ALE case considers a fixed mesh distortion (defined by Eqs. (101) & (102)). Fig. 11 shows the flow field and its convergence rates. Again, only load case 2 attains cubic velocity convergence. Compared to ALE case a, the errors in all fields is slightly larger (i.e. they are slightly offset).

5.3.4 Varying mesh velocity (ALE case d)

The last ALE case considers transient mesh motion (defined by $\mathbf{x}_m(t)$ given in (101) & (103)). Fig. 12 shows the flow field and its convergence rates. As in the previous example the transient part is $\mathbf{v}' = \theta' \mathbf{v}_2$. But now $\mathbf{v}_2$, and hence $\mathbf{v}'$, has an out-of-plane component (Sauer, 2025). As a result, the problem cannot be solved accurately anymore by projection (load case 2). Thus only load case 4 gives accurate results. Also attributed to the out-of-plane component of $\mathbf{v}'$

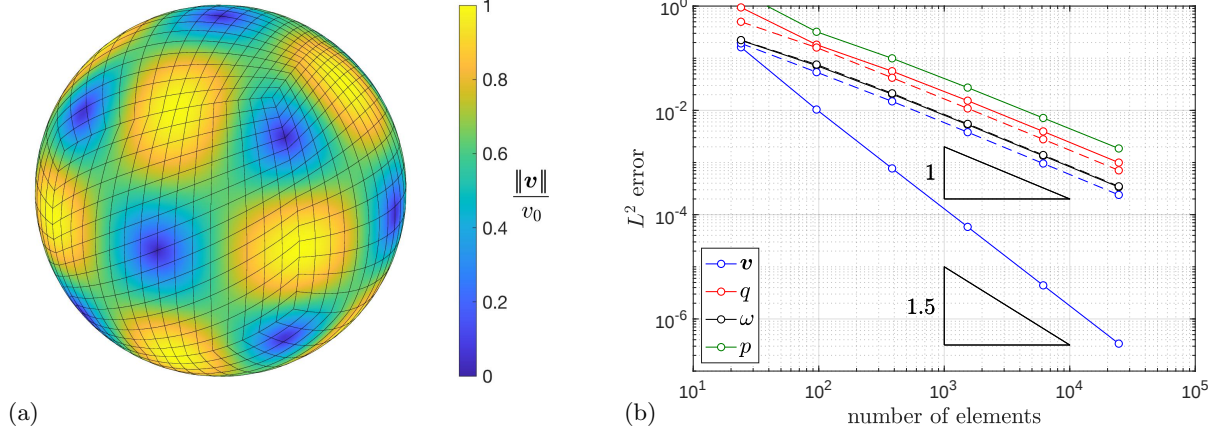


Figure 11: Octahedral vortex flow on a rigid sphere: (a) Flow field $\mathbf{v}$ and (b) convergence behavior due to the prescribed fixed mesh distortion $\mathbf{x}_m$ given in (101) & (102) (ALE case c). Solid lines are for load case 2; dashed lines for load case 4. All convergence rates are optimal or better, as seen in Fig. 8 before.

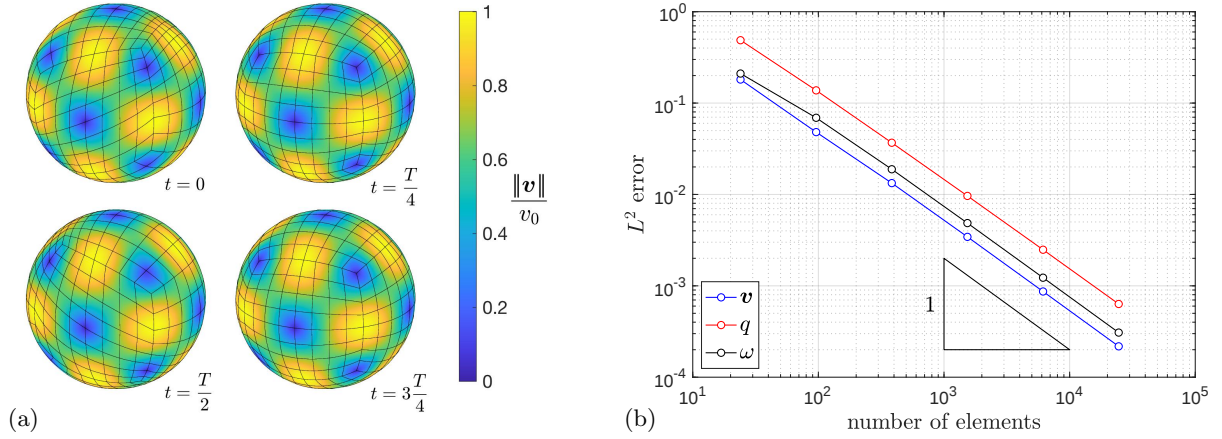


Figure 12: Octahedral vortex flow on a rigid sphere: (a) Flow field $\mathbf{v}$ and (b) convergence behavior due to the prescribed mesh motion $\mathbf{x}_m(t)$ given in (101) & (103) (ALE case d). This mesh motion causes out-of-plane components in $\mathbf{v}'$ and hence can only be solved accurately with Neumann BC (load case 4). All convergence rates are $O(h^2) = O(n_{el}^{-1})$. Here, the L^2 error from (97) is averaged over $n_{t1}/4$ steps.

is that fluid velocity $\mathbf{v}$ converges only quadratically in h , even for time step n_{t2} . This was different in the previous example (see Fig. 6), where $O(h^3)$ -convergence was obtained for n_{t2} . The convergence rates for q and ω are not affected, and the same as before.

6 Numerical examples for deforming surface flows

This section now considers the case when the surface deforms due to the flow. Thus there are seven primary unknowns now: The components of $\mathbf{v}$, q and $\mathbf{v}_m$. In Sec. 5 there were never more than four ($\mathbf{v}$, q). First, the two examples from the previous section are revisited in Secs. 6.1 and 6.2. Due to area-incompressibility and a closed surface, the total surface area does not change during deformation. This is different in the final example (Sec. 6.3) due to an inflow boundary. Since $\mathbf{v}_m$ is an unknown now, the mesh equations from Sec. 4.2.3 need to be included. For this, the examples in Secs. 6.1 and 6.2 use Eulerian mesh motion, while the example of Sec. 6.3 uses both Eulerian and elastic mesh motion. The surface (mesh) $\mathbf{x}$ then follows from integrating $\mathbf{v}_m$ using the update formula (92.2), here with $\gamma = 1/2$. Thus time stepping is still needed, even when considering steady flows (with negligible inertia $\rho \mathbf{v}'$), which are the first two sub-cases

considered in both Sec. 6.1 and 6.2.

6.1 Simple shear flow on a deformable sphere

First the axisymmetric shear flow example from Sec. 5.2 is examined using the problem setup from Sec. 5.1 (with the meshes from Table 1 and unit values for r , ω_0 , η and ρ), unless otherwise noted. Three sub-cases are studied in the following.

6.1.1 Fixed surface solution

The first sub-case uses $\mathbf{f} = \mathbf{f}(\mathbf{X})$ from (99) and (107), which is the fixed sphere solution for which the surface is not supposed to change. This is obtained correctly as Fig. 13 shows: The

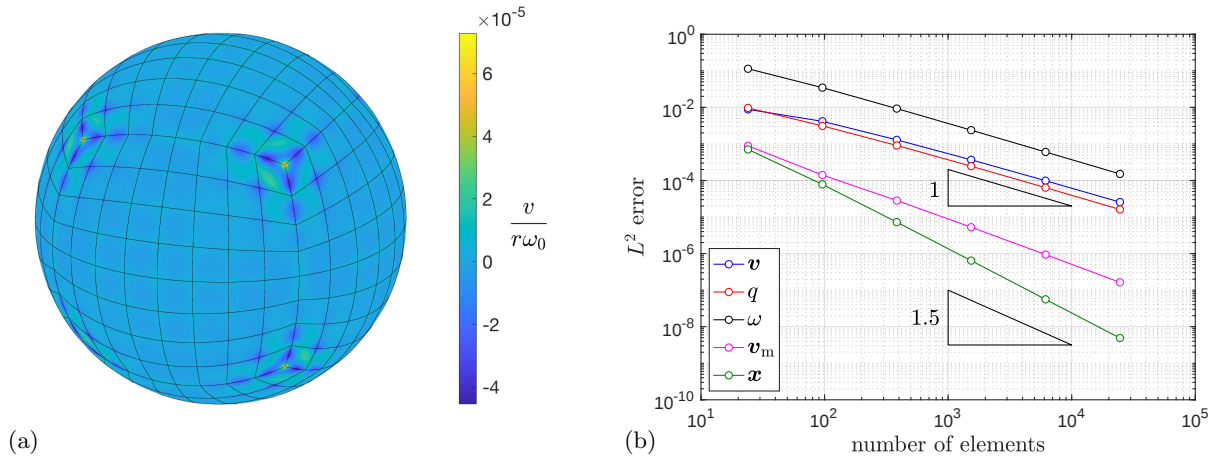


Figure 13: Simple shear flow on a deformable sphere with balancing pressure applied: (a) normal surface velocity $v = \mathbf{v} \cdot \mathbf{n}$ and (b) convergence behavior. All convergence rates are optimal: The primary unknowns $\mathbf{v}$, q , $\mathbf{v}_m$ converge with $O(h^2) = O(n_{\text{el}}^{-1})$, while the integrated variable $\mathbf{x}$ converges at least with $O(h^3) = O(n_{\text{el}}^{-1.5})$.

problem converges to the correct solution at ideal rates. In particular, the normal velocity decays to zero with mesh refinement. As Fig. 13a shows, for $m = 4$ it is already four magnitudes below the tangential velocity (seen in Fig. 2a). In the simulations here, the time step $\Delta t = 1/m/\omega_0$ is used, while the polar pressure is taken as $p_p = 4\rho r\omega_0^2$. The pressure is chosen larger than before since it has a stabilizing effect on the surface shape. Conversely, too low surface pressures (esp. negative ones) do not admit stable solution, which is seen in the following sub-case.

6.1.2 Deformable surface solution for $\rho v' = 0$

The second sub-case uses the prescribed uniform surface tractions $f_2 = 0$ and $p = \bar{p} = \rho r\omega_0^2$ within Eq. (99), where r still denotes the initial, undeformed sphere radius. This internal pressure p is larger at the equator than before, and so a flattening of the sphere can be expected. This is confirmed by Fig. 14. Here the equatorial radius has increased by about 2.4 %, while the pole distance has decreased by about 9.4 %. The evolution of these deformations is shown in Fig. 15a. Here the prescribed shear traction f_1 has been ramped-up using

$$\bar{\omega}_0(t) = \frac{\omega_0}{2} \begin{cases} 1 - \cos(\pi t/t_1) & \text{for } 0 \leq t < t_1 \\ 2 & \text{for } t \geq t_1 \end{cases}, \quad (111)$$

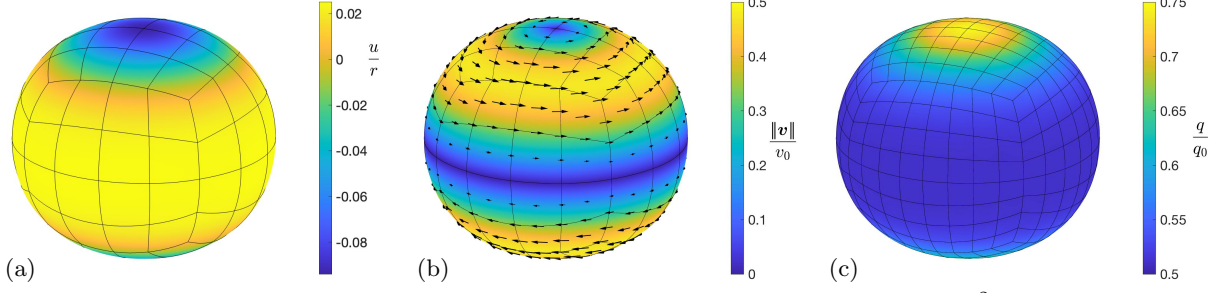


Figure 14: Simple shear flow on a deforming sphere with constant pressure $\bar{p} = \rho r \omega_0^2$ applied: (a) radial surface displacement $u = \mathbf{u} \cdot \mathbf{e}_r$, (b) velocity field $\mathbf{v}$ and (c) surface tension q plotted on the deformed sphere at time $t = 40/\omega_0$. The equatorial and polar distances change by 2.44% and 9.41%, respectively. The velocity and surface tension range between $\|\mathbf{v}\| \in [0, 0.476] v_0$ and $q \in [0.512, 0.747] q_0$, where $v_0 := r\omega_0$ and $q_0 := r\bar{p}$.

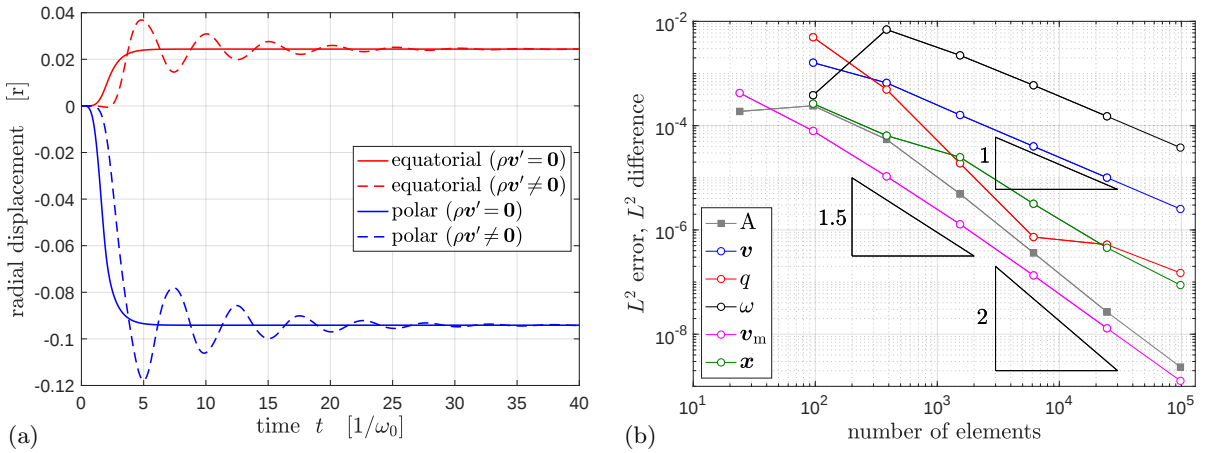


Figure 15: Simple shear flow on a deformable sphere with constant pressure $\bar{p} = \rho r \omega_0^2$: (a) shape change over time and (b) convergence behavior at $t = 40/\omega_0$ for $\rho \mathbf{v}' = \mathbf{0}$. All convergence rates are at least optimal: The primary unknowns $\mathbf{v}$ and q converge with $O(h^2) = O(n_{\text{el}}^{-1})$, and $\mathbf{v}_m$ even with $O(h^3) = O(n_{\text{el}}^{-1.5})$. The integrated variables $\mathbf{x}$ and A (the surface area) converge with $O(h^3) = O(n_{\text{el}}^{-1.5})$ and $O(h^4) = O(n_{\text{el}}^{-2})$, respectively. Here, the plots for A and $\mathbf{v}_m$ are L^2 errors (compared to the exact values), while the remaining plots are L^2 differences (between subsequent meshes).

for ω_0 in (99). The ramp-up time is chosen as $t_1 = 2/\omega_0$, while the model parameters r , ω_0 , ρ and η_n are again taken as unity apart from $\eta = 1/2$. The time step $\Delta t = 2/m/\omega_0$ is used. It is noted that the sphere volume changes but the surface area is conserved due to the considered area-incompressibility. The computations run stable as long as sufficient out-of-plane pressure $\bar{p}$ and out-of-plane viscosity η_n are applied, as they stabilize the surface motion. The latter is applied according to (77) with η_n taken as unity in this example. Without such stabilization, the flow may cause surface instabilities such as out-of-plane surface buckling.

Fig. 15b shows the L^2 errors of the total surface area $A = 4\pi r^2$ and the mesh velocity $\mathbf{v}_m$ (which vanishes at steady state), as well as the L^2 differences of $\mathbf{v}$, q , ω and $\mathbf{x}$ (for which no exact solution exists). As seen, the convergence rate (w.r.t. element size h) is quartic for A , cubic for $\mathbf{v}_m$ & $\mathbf{x}$, and at least linear for the other fields. The exceptional behavior of q is due to a zero-crossing between the third and fourth point. It is noted, that under Stokes flow conditions (achieved by setting $\rho = 0$ to eliminate the convective term), no surface deformations occur in this example.

6.1.3 Deformable surface solution for $\rho\mathbf{v}' \neq \mathbf{0}$

The results discussed so far are the steady solution (where $\rho\mathbf{v}'$ is neglected). Now the transient case is examined, where $\rho\mathbf{v}' \neq \mathbf{0}$. It is also shown in Fig. 15a. As can be expected, the surface inertia $\rho\mathbf{v}'$ leads to oscillations in the deformation. They decay due to viscosity, such that eventually the steady state from before is reached. Here now the viscosity parameters are reduced to $\eta = 1/4$ and $\eta_n = 0$ compared to before. This works well, since inertia additionally stabilizes the surface motion.

6.2 Octahedral vortex flow on a deformable sphere

Second, the non-axisymmetric octahedral surface flow from Sec. 5.3 is examined using again the problem setup from Sec. 5.1. The same three sub-cases as in Sec 6.1 are investigated here.

6.2.1 Fixed surface solution

The first sub-case considers the fixed sphere solution that appears if the varying out-of-plane pressure $\bar{p}$ from Eq. (110) is prescribed with $p_p = 4p_0$, $p_0 := \rho v_0^2/r$. As in Sec. 6.1.1 before, the problem correctly converges to the analytical solution at ideal rates, as Fig. 16 shows. The convergence rate for the surface shape $\mathbf{x}$ is even close to $O(n_{\text{el}}^{-2})$.

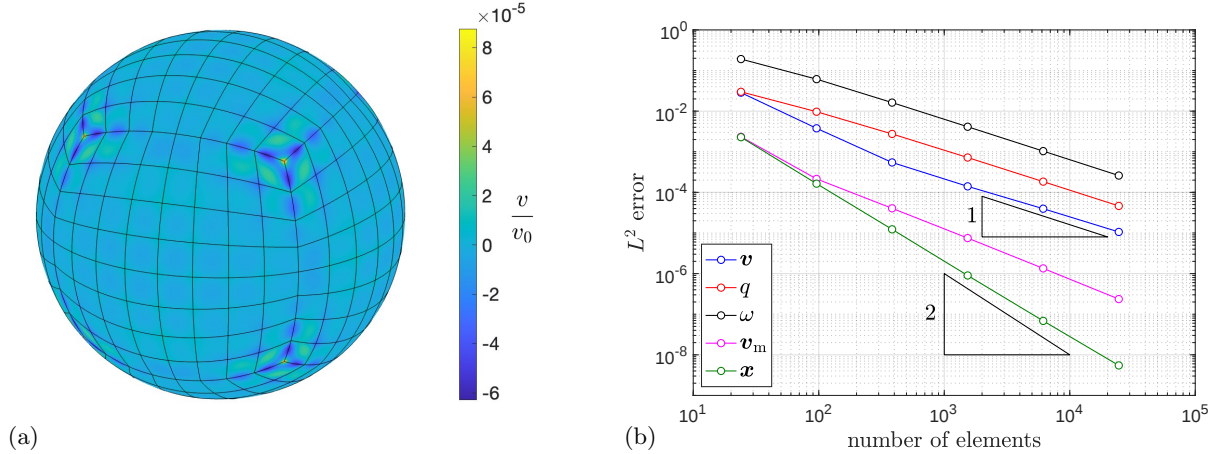


Figure 16: Octahedral vortex flow on a deformable sphere with balancing surface pressure applied: (a) normal surface velocity $v = \mathbf{v} \cdot \mathbf{n}$ and (b) convergence behavior. All convergence rates are optimal: At least $O(n_{\text{el}}^{-1})$ for primary unknowns $\mathbf{v}$, q , $\mathbf{v}_m$, and at least $O(n_{\text{el}}^{-1.5})$ for the integrated variable $\mathbf{x}$.

6.2.2 Deformable surface solution

If a different surface pressure is used for $\bar{p}$, an out-of-plane motion occurs. This is shown in Fig. 17 for $\bar{p} = 2.5p_0$. As seen the surface deformation aligns with the flow field. The evolution of these deformations is shown in Fig. 18a both for the steady state ($\rho\mathbf{v}' = \mathbf{0}$) and transient case ($\rho\mathbf{v}' \neq \mathbf{0}$). Here the flow has been ramped-up using Eq. (111) with $t_1 = 2/\omega_0$ and $\Delta t = 1/m/\omega_0$. The convergence behavior seen in Fig. 18b is similar to the one of Fig. 15.

6.3 Inflation of a soap bubble

The final example considers the inflation of a soap bubble. It is an advanced flow example involving transient flow on deforming surfaces and introducing two new challenges: Large

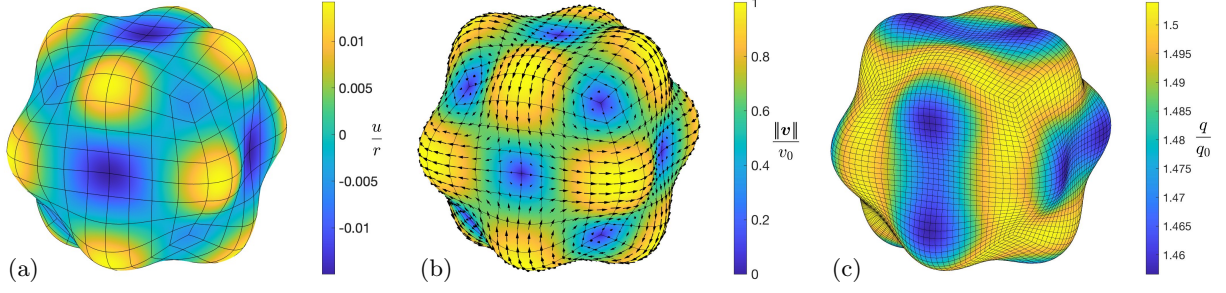


Figure 17: Octahedral vortex flow on a deformable sphere with constant pressure $\bar{p} = 2.5p_0$ applied: (a) radial surface displacement $u = \mathbf{u} \cdot \mathbf{e}_r$, (b) velocity field $\mathbf{v}$ and (c) surface tension q plotted on the deformed sphere. The relative displacement ranges between -1.48% and 1.42%. It is scaled-up by a factor of 10 in the three plots to increase its visibility. The velocity and surface tension range between $\|\mathbf{v}\| \in [0, 1.014]v_0$ and $q \in [1.457, 1.504]q_0$, where $q_0 := rp_0$.

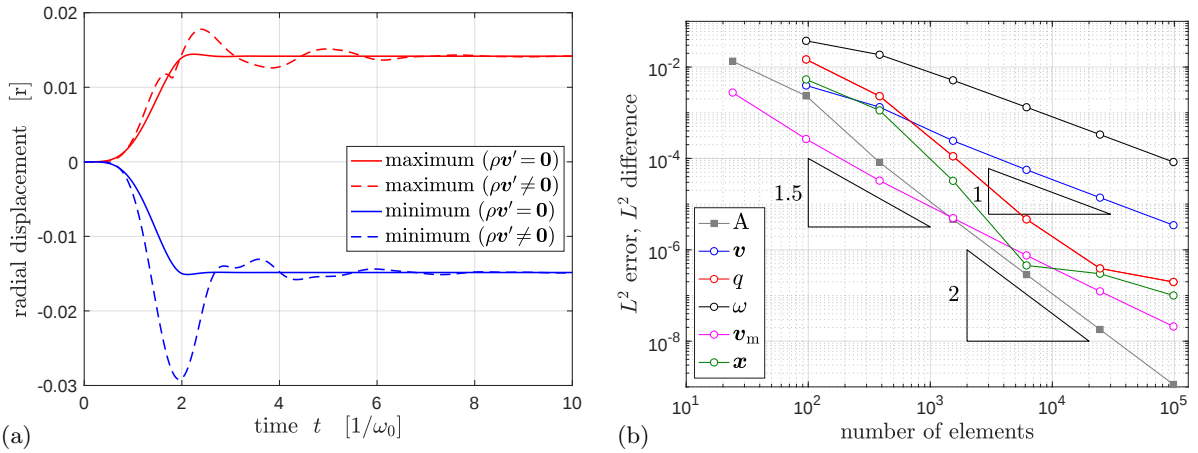


Figure 18: Octahedral vortex flow on a deformable sphere with constant pressure $\bar{p} = 2.5p_0$: (a) shape change over time and (b) convergence behavior at $t = 10/\omega_0$ for $\rho v' = \mathbf{0}$. Area A and mesh velocity $\mathbf{v}_m$ are L^2 errors converging at optimal rates. The remaining fields are L^2 differences decreasing at least linearly. The results for q and $\mathbf{x}$ have zero-crossings between the 4th and 5th point.

shape changes and deformation-dependent boundary conditions. It is also used to examine C^1 -continuous surface discretization and the influence of surface stretches on Eqs. (79)-(84). The considered parameters are given in Table 4. The following computational setup is used, see Fig. 19: An initially hemispherical bubble, pinned at the left boundary, is inflated by increasing

symbol	material parameter	value	unit
R	initial bubble radius	10	mm
p_0	initial surface pressure	5	$\mu\text{N}/\text{mm}^2 = \text{Pa}$
q_0	initial surface tension	25	$\mu\text{N}/\text{mm}$
η	surface viscosity	0.04	$\mu\text{N ms}/\text{mm}$
ρ	surface density	0.2	$\mu\text{N ms}^2/\text{mm}^3$
μ_m	ALE mesh stiffness	$10^{-5} \dots 10^{-9}$	$\mu\text{N}/\text{mm}$

Table 4: Soap bubble example: Considered geometry and material parameters. Here, q_0 is estimated from Sane et al. (2017) for a water solution with 4% soap, η and ρ are based on the bulk viscosity $\tilde{\eta} = 0.2 \text{ Pa}\cdot\text{s}$ and density $\tilde{\rho} = 1 \text{ g}/\text{cm}^3$ multiplied by the soap film thickness $h = 0.2 \mu\text{m}$. From this follows $p_0 = 2q_0/R$. Numerical parameter μ_m is determined from computational tests. $\eta_n = 0$ unless noted otherwise.

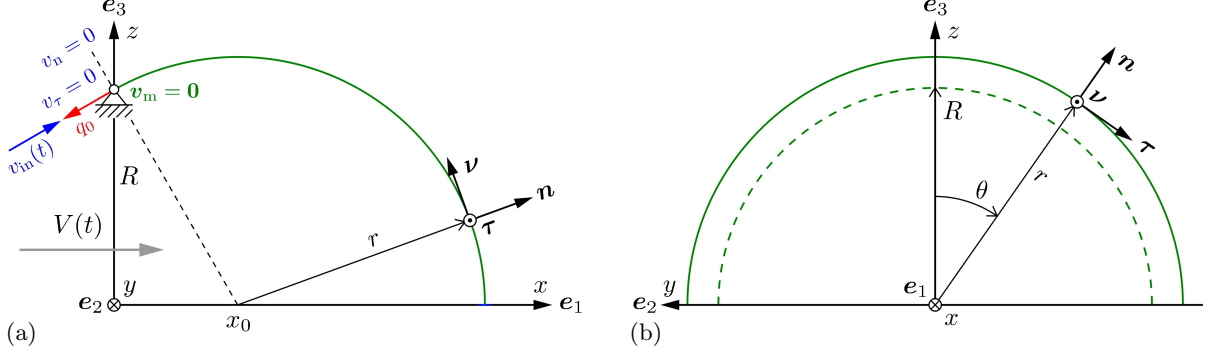


Figure 19: Soap bubble inflation: Problem setup, geometry and boundary conditions. (a) side view; (b) back view. At the inflow, the Dirichlet boundary conditions $v_n = 0$, $v_\tau = 0$, $q = q_0$, $\mathbf{v}_m = \mathbf{0}$ and the Neumann boundary condition $T_\nu = q$ are prescribed.

the enclosed bubble volume according to

$$V(t) = V_0 + \Delta V(1 - \cos(\pi t/t_1)), \quad V_0 := \frac{2\pi}{3}R^3. \quad (112)$$

This causes an inflow of new material at the left boundary. The corresponding boundary conditions on the flow and mesh motion are illustrated in Fig. 19a. Explicitly these are $v_n = 0$, $v_\tau = 0$ and $T_\nu = q$ for the fluid velocity, $q = q_0$ for the surface tension, and $\mathbf{v}_m = \mathbf{0}$ for the mesh motion. For $\mathbf{v}$, this is both a mixed BC – Neumann along in-flow direction $\boldsymbol{\nu}$ and Dirichlet along normal and circumferential directions $\mathbf{n}$ and $\boldsymbol{\tau}$ – and a follower BC, as directions $\boldsymbol{\nu}$ and $\mathbf{n}$ change during inflation. Details on its FE application and linearization are given in Appendix A.2.3. Volume constraint (112) is added to the system of equations by the Lagrange multiplier method, see Sauer et al. (2014) for details. The corresponding Lagrange multiplier – the internal bubble pressure p – becomes a fourth unknown next to $\mathbf{v}$, q and $\mathbf{v}_m$. The resulting transient system is solved monolithically with the implicit trapezoidal time integration scheme described in Sec. 4.3. Table 5 lists the FE meshes used for this.

m	n_{sel}	n_{bel}	n_{no} (Q2)	n_{dof} (Q2)	n_{no} (N2)	n_{dof} (N2)	n_t
4	96	16	417	2,920	160	1,121	50
8	384	32	1,601	11,208	504	3,529	100
16	1,536	64	6,273	43,912	1,768	12,377	200
32	6,144	128	24,833	173,832	6,600	46,201	400
64	24,576	256	98,817	691,720	25,480	178,361	800
128	98,304	512	394,247	2,759,730	100,104	700,729	1,600

Table 5: Soap bubble inflation: Considered FE meshes based on bi-quadratic Lagrange (Q2) and bi-quadratic NURBS (N2) elements and corresponding dofs for unknown $\mathbf{v}$, q , $\mathbf{v}_m$ (seven unknown components). Here, $n_{\text{sel}} = 6m^2$ is the number of surface FE, $n_{\text{bel}} = 4m$ the number of boundary FE, n_{no} the number of nodes, and $n_{\text{dof}} = 7n_{\text{no}} + 1$ the number of dofs; n_t is the number of time steps used in the first example of Sec. 6.3.1.

Two cases are considered in the following sections: uniform and non-uniform inflow. They illustrate the inaccuracies resulting from an in-plane Eulerian surface description. It is shown that these inaccuracies reduce substantially with the proposed in-plane mesh elasticity formulation of Eqs. (89) & (33)

6.3.1 Uniform inflow

The first case considers a (spatially) uniform inflow velocity, such that the flow remains fully laminar. The 2D analytical study of Sauer (2025) showed that a purely Eulerian surface de-

scription leads to large mesh deformations at the tip of the bubble, while an ALE surface description allows for a uniform mesh deformation across the surface. Therefore it is expected that the proposed ALE description based on mesh elasticity will yield better numerical results. The inflation parameters in (112) are taken as $\Delta V = 9V_0$ and $t_1 = 2\text{s}$ with time step size $\Delta t = 80\text{ms}/m$.

Fig. 20 shows the resulting shape change of the bubble during inflation. As seen, for these

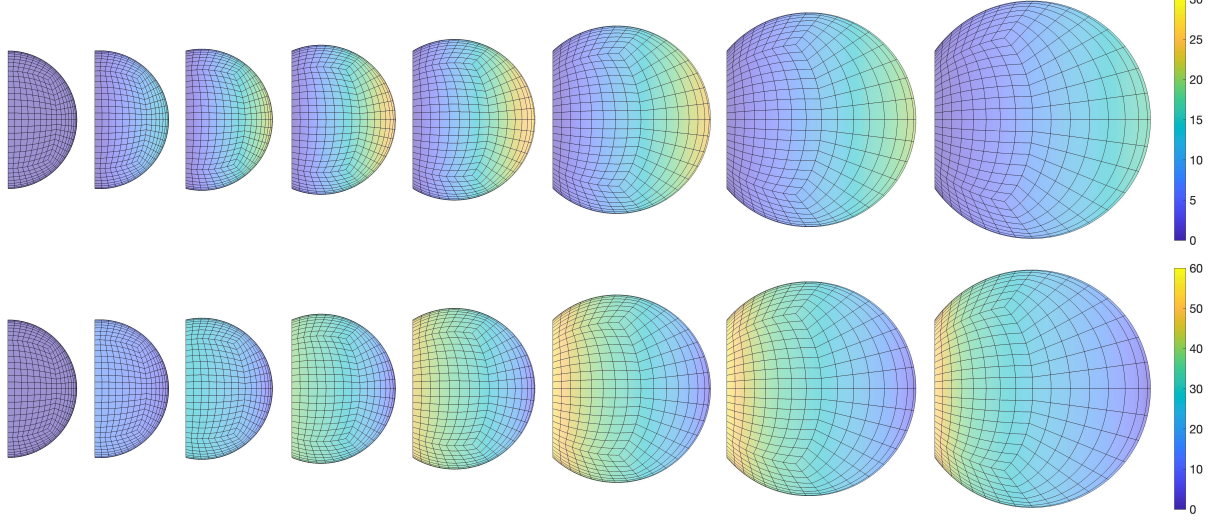


Figure 20: Laminar inflation of a soap bubble: Bubble shape at $t = \{0, 0.1, 0.2, 0.3, 0.4, 0.6, 0.8, 1\}\text{s}$, i.e. $V \approx \{1, 1.1, 1.4, 2.0, 2.7, 4.7, 7.2, 10\}V_0$ (left to right). Top row: normal fluid velocity v_n ; bottom row: tangential fluid velocity $\|v_t\|$. The results are from elastic mesh motion with $\mu_m = 10^{-7}\mu\text{N}/\text{mm}$.

parameters, the bubble remains spherical during inflation. This is due to the small bubble radius and high surface tension. The comparison between the in-plane Eulerian and in-plane elastic mesh motions is shown in Fig. 21.

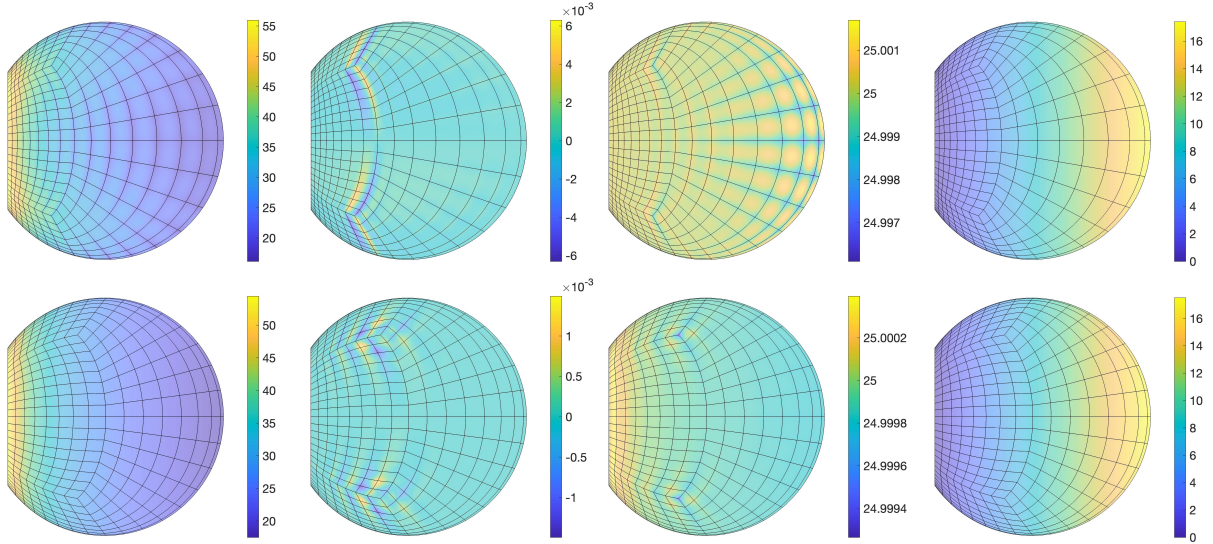


Figure 21: Laminar inflation of a soap bubble: Left to right: fluid velocity $\mathbf{v}$ [mm/s], vorticity ω [1/s], surface tension q [$\mu\text{N}/\text{mm}$] and mesh velocity $\mathbf{v}_m$ [mm/s] for $m = 8$, $V = 10V_0$ and $\mu_m = 10^{-7}\mu\text{N}/\text{mm}$. Top row: Eulerian mesh motion; Bottom row: elastic mesh motion. The latter is more accurate, especially for $\mathbf{v}$, ω and q . It also shows much smaller mesh deformations towards the bubble tip.

In the Eulerian case large errors are observed at the tip of the bubble. In particular, q and ω are quite off. On the other hand, much more accurate results are obtained with elastic mesh ALE,

as long as μ_m is sufficiently small. This is also seen in Fig. 22. Here, the left side shows the fluid

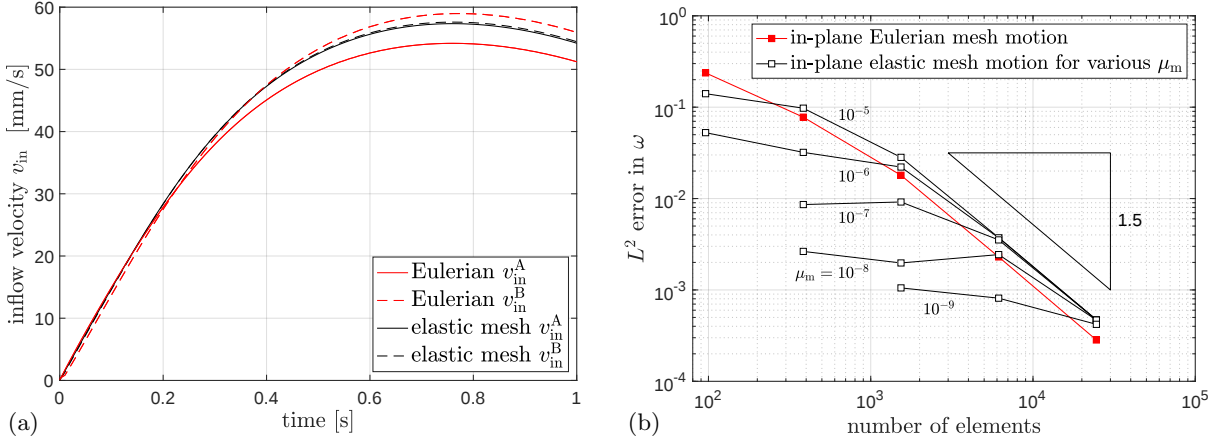


Figure 22: Laminar inflation of a soap bubble: Accuracy comparison between the Eulerian and elastic mesh description. (a) Inflow velocity $v_{in}(t)$ at inflow points A and B (at $\theta = 0$ and $\theta = \pi/4$) for $m = 8$ and $\mu_m = 10^{-7} \mu\text{N}/\text{mm}$. (b) L^2 error of the surface vorticity (from Eq. (113)) vs. mesh refinement. The elastic mesh description is much more accurate than the Eulerian mesh description, especially for coarse meshes and small μ_m .

velocities at two different inflow points, which should be identical. As seen, the elastic mesh result is much better. The right side shows the normalized L^2 error of the surface vorticity,

$$e_\omega := T \|\omega\|_{L^2}, \quad \|\omega\|_{L^2} := \sqrt{\int_S \omega^2 da}, \quad (113)$$

normalized by the final time $T = 1\text{s}$. In theory, e_ω should be exactly zero, as there is no surface vorticity in this example. Again, the elastic mesh results tend to be much better than the Eulerian ones, especially for small μ_m .⁴ As in the examples of Sec. 5, ideal convergence rates are obtained. Interestingly, for each fixed μ_m , the error of the elastic mesh results starts at a very low level, then stagnates, and finally decreases together with the Eulerian case – but falling slightly behind it. The reason for this behavior is suspected to lie in the surface continuity: The mesh elasticity formulation, contrary to the Eulerian formulation, relies on an accurate decomposition into in-plane and out-of-plane parts, as Eq. (89) shows. Such a decomposition, however, is not represented very accurately with quadratic Lagrange elements, since they are only C^0 -continuous at element boundaries. C^1 -continuous finite element discretizations can be expected to remedy this issue and yield further accuracy gains. This is confirmed in the following example. The superiority of C^1 -continuous finite element discretizations was already seen for the quasi-static droplet simulations in Sauer (2014), which also used formulation (89) to stabilize the mesh.

As the analytical example in Sauer (2025) shows, the Eulerian mesh quality degrades with inflation size. Therefore, if the bubble is inflated even further, the elastic mesh results should become even better than those in Fig. 22. This is confirmed by the results in Fig. 23, which consider inflation up to $V = 50V_0$ (using $\Delta V = 49V_0$ and $t_1 = 4\text{s}$ within Eq. (112) and taking $\Delta t = 64\text{ms}/m$). Now the elastic mesh results are consistently more accurate than the Eulerian ones.

The picture changes slightly when the stabilization terms of Sec. 4.2.2 are integrated over the current instead of the initial mesh. This is shown in Fig. 24: Comparing Fig. 24a with 23b,

⁴The smaller μ_m , the more accurate the results, especially for coarse meshes. But too small μ_m cause ill-conditioning. The value $\mu_m = 10^{-6} \mu\text{N}/\text{mm}$ seems to be a good compromise for the considered Lagrange meshes.

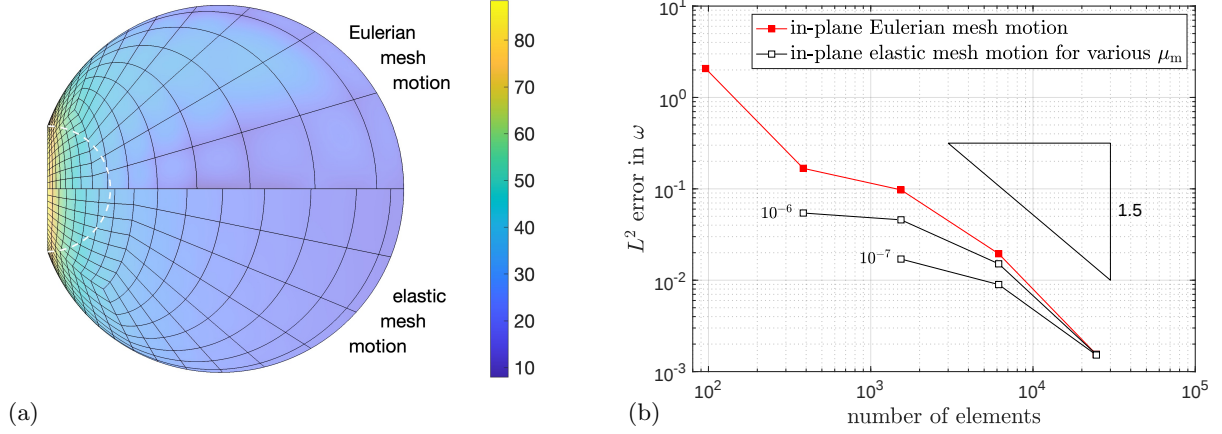


Figure 23: Laminar inflation of a soap bubble ($V = 50V_0$): Accuracy comparison between Eulerian and elastic mesh motion for stabilization term integration over the initial surface. (a) Comparison of velocity field $\mathbf{v}$ [mm/s] for $m = 8$ and $\mu_m = 10^{-6} \mu\text{N/mm}$. (b) L^2 error of surface vorticity vs. mesh refinement. The accuracy gains of the elastic mesh formulation are bigger than in Fig. 22 as they increase with bubble size. The dashed white line shows the initial bubble shape.

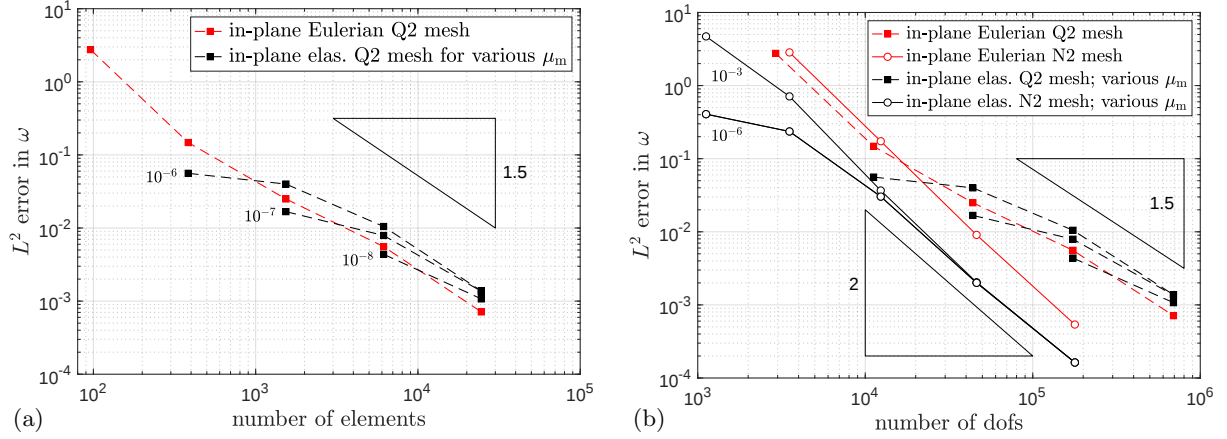


Figure 24: Laminar inflation of a soap bubble ($V = 50V_0$): Accuracy comparison between Eulerian and elastic mesh motion for stabilization term integration over the current surface. L^2 error of surface vorticity vs. mesh refinement for (a) quadratic Lagrange (Q2) discretization and (b) quadratic NURBS (N2) discretization. The Q2 results of (a) are included in (b), but now plotted vs. the number of dofs. The current surface integration improves the accuracy, especially for the more distorted Eulerian mesh. Thus, the latter becomes as accurate as the elastic mesh for Q2 elements. But for N2 elements, the elastic mesh is still much more accurate. Further, N2 achieves a better convergence rate than Q2.

shows that the current surface integration improves accuracy, especially for the more distorted Eulerian mesh. Now Eulerian and elastic mesh descriptions are equally accurate for quadratic Lagrange elements. For quadratic NURBS (non-uniform rational B-spline) elements, however, the elastic mesh description is still much more accurate than the Eulerian one, as Fig. 24b shows. The figure also shows that the convergence rate of NURBS elements is higher than that of Lagrange elements. The reason for the superior behavior of NURBS elements lies in their more accurate representation of surface curvatures, which appear explicitly in the elastic mesh integral of Eq. (89.2). For NURBS, mesh parameter μ_m can also be chosen much larger, as Fig. 24b shows, and hence NURBS meshes are much less likely to become ill-conditioned.

The results of Fig. 24 illustrate that the current surface integration of the stabilization terms can yield major accuracy gains. This is because in that example the volume change and hence surface stretches are very large – up to $J_m = 250$ for Eulerian mesh motion, which is why

accuracy gains are particularly high there. In the previous (and remaining) examples, surface stretches are much smaller, and hence accuracy gains are much smaller. Thus, integration over dA becomes more convenient, as no extra linearization terms, that can contribute to ill-conditioning, are needed.

It is finally noted that in the examples of Sec. 6.3.1, the bubble shape remains nearly spherical, which is why the surface tension (seen in Fig. 21) is nearly constant. This can change for non-uniform inflow.

6.3.2 Non-uniform inflow

The last case considers the situation when inflow is only allowed partially – on the boundary between $|\theta| < \pi/8$ (see Fig. 19b). On this boundary the boundary conditions are the same as before, while on the remaining boundary all velocities components are set to zero. This partial inflow case causes two vortices to appear on the evolving surface, as Fig. 25 shows. The

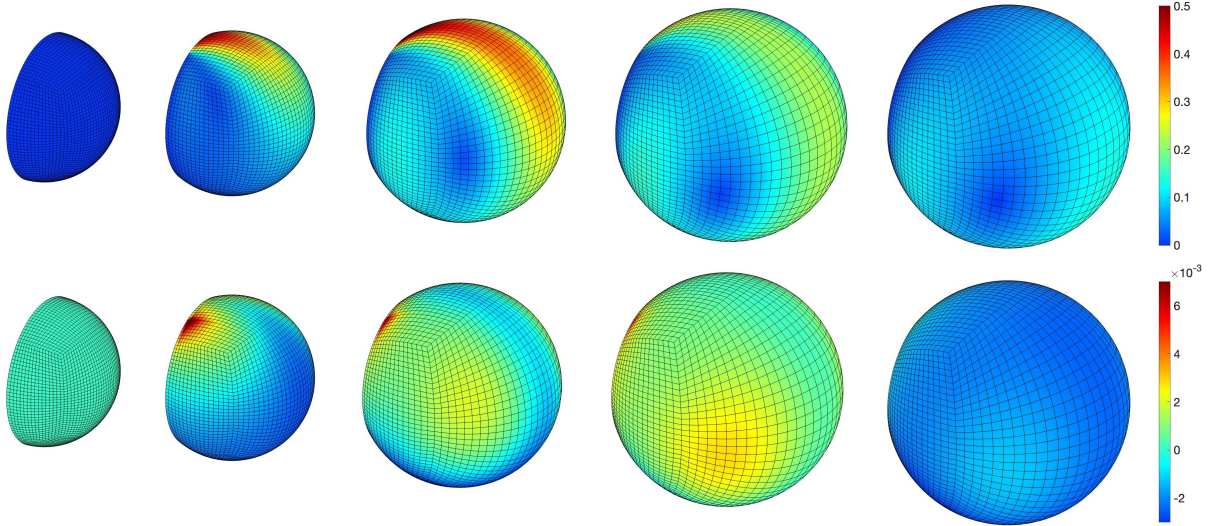


Figure 25: Non-uniform inflation of a soap bubble with high surface tension: Left to right: Bubble shape at $t = \{0, 0.25, 0.5, 0.75, 1\}$ s. Top row: Velocity field $\mathbf{v}$ [m/s]. Bottom row: Surface tension change Δq [$\mu\text{N}/\text{mm}$]. The results are from elastic mesh motion with $m = 16$, $\Delta t = 2.5\text{ms}$, $\mu_m = 10^{-4}\mu\text{N}/\text{mm}$, $\alpha_{\text{DB}} = 1000$ and R to ρ from Table 4. For these parameters, the bubble shape remains near spherical.

corresponding surface velocity evolution is shown in Fig. 26 for various meshes and α_{DB} values. Considered are the inflow parameters $V = 8V_0$ and $t_1 = 1\text{s}$ in Eq. (112). The resulting peak value of v_{max} is about 0.61m/s leading to a Reynolds number of 23.95 (taking the inlet diameter $\pi R/4$ as characteristic length). The example shows that the proposed ALE formulation with elastic mesh motion can capture non-trivial nonlinear flow fields on evolving surfaces. It converges with mesh refinement as Fig. 26a shows. In this example, α_{DB} has a significant influence on the solution, as Fig. 26b shows. Larger α_{DB} are observed to alleviate inaccuracies in q , but at the price of less accurate enforcement of area-incompressibility ($\text{div}_s \mathbf{v} = 0$). The values used in Fig. 26a are chosen based on examining fields q and $\text{div}_s \mathbf{v}$ for different α_{DB} , and they constitute a good compromise, but more rigorous guidelines should be investigated in future work.

Fig. 25 also shows that the bubble shape remains near spherical during inflation. This is because the flow does not alter the high static surface tension q_0 significantly (Δq is more than 1000 times less than q_0 as Fig. 25 shows). This changes when q_0 is reduced drastically, as Fig. 27 shows. Here now $q_0 = 0.01\mu\text{N}/\text{mm}$ is used together with $R = 100\text{mm}$ (hence $p_0 = 0.2\text{mPa}$), $\alpha_{\text{DB}} = (m/32)^2 \cdot 10^6$ and $\mu_m = 10^{-4}\mu\text{N}/\text{mm}$. Parameters η and ρ are the same as before, see

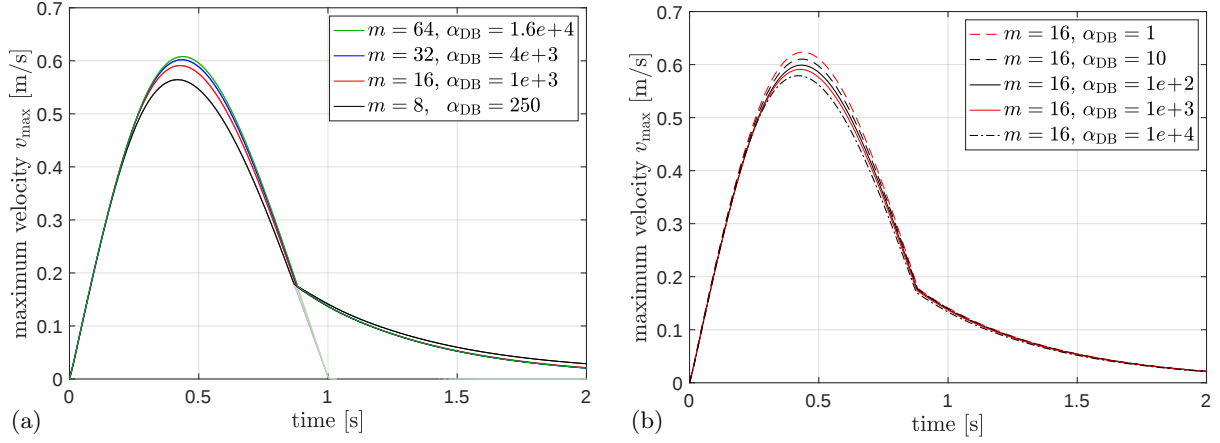


Figure 26: Non-uniform inflation of a soap bubble with high surface tension: (a) Mesh convergence and (b) α_{DB} -convergence of the maximum surface velocity. The light lines in (a) show the inflow velocity at $\theta = 0$, which is the maximum velocity until $t \approx 0.883$ s.

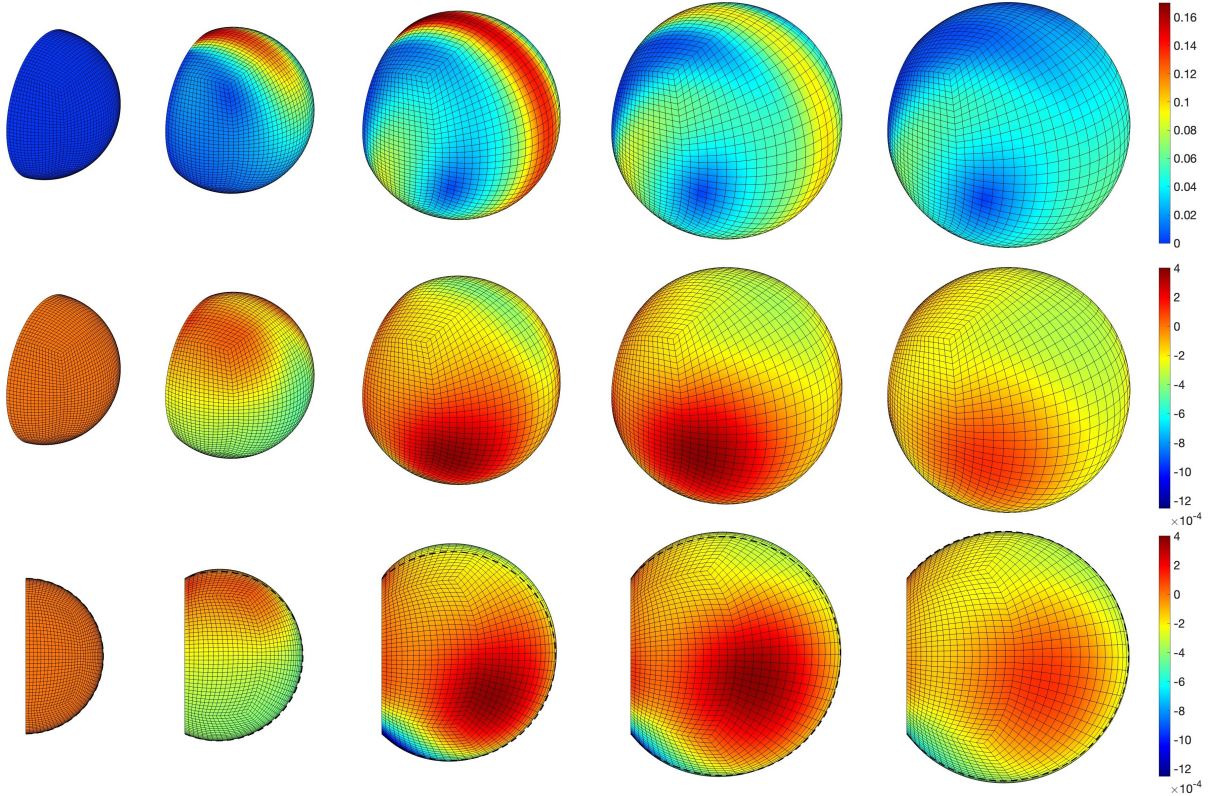


Figure 27: Non-uniform inflation of a soap bubble with low surface tension: Left to right: Bubble shape at $t = \{0, 8, 16, 24, 32\}$ s. Top row: Velocity field $\mathbf{v}$ [m/s]. Middle & bottom row: Surface tension change Δq [$\mu\text{N}/\text{mm}$]. The dashed line below marks the shape of a perfect sphere with equal volume. This shows that the bubble shape deviates significantly from a sphere, especially at the top, where the fluid velocity is high. The results are from elastic mesh motion with $m = 16$, $\Delta t = 16\text{ms}$, $R = 100\text{mm}$, $q_0 = 0.01\mu\text{N}/\text{mm}$, $p_0 = 0.2\text{mPa}$, $\mu_m = 10^{-4}\mu\text{N}/\text{mm}$ and $\alpha_{DB} = 10^4$ using η and ρ from Table 4.

Table 4. The smaller q_0 necessitates a larger α_{DB} value to capture the surface flow accurately. Additionally, small out-of-plane viscosity has been added now according to Remark 4.4, using $\eta_m = 0.001\mu\text{Nms}/\text{mm}^3$ in order to stabilize the out-of-plane bubble motion. This value has been found to successfully bound out-of-plane surface oscillations while having negligible influence on the fluid flow. The peak velocity now is about 0.22m/s leading to a Reynolds number of 86.4 .

As Fig. 27 shows, the bubble shape deviates significantly from a sphere during inflation. This deviation is related to the surface velocity. This was also seen in the example of Fig. 17, but in that case there was no tangential mesh motion and no inflow into the surface. The proposed formulation thus also works for non-spherical shapes. Although it has to be noted that not all parameter combinations work equally well. In particular, the numerical parameters α_{DB} and μ_{m} have important influences on the accuracy and stability of the simulations. This should be studied further in future work.

The example of Fig. 27 has the highest Reynolds number. Hence the local Péclet number of Eq. (43) will also be highest, and the question arises whether the computational meshes are sufficient to guarantee convective stability as noted in Remark 3.2. Therefore, Fig. 28 shows

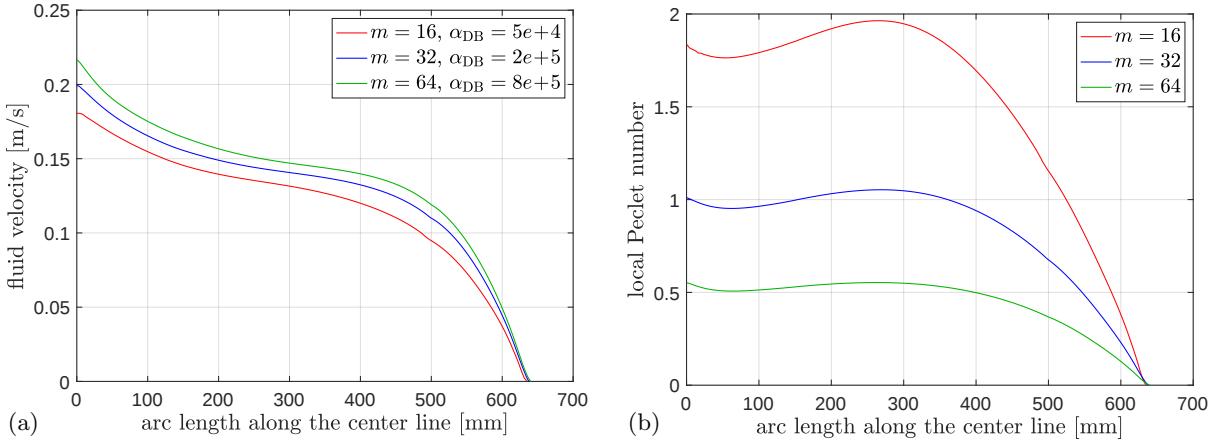


Figure 28: Non-uniform inflation of a soap bubble with low surface tension: (a) Fluid velocity $\|\mathbf{v}\|$ and (b) local Péclet number Pe along the symmetry line for time $t = 16\text{s}$.

the local Péclet number together with fluid velocity $\|\mathbf{v}\|$, which in this example is largest along the symmetry line and approximately equal to the convective velocity $\|\mathbf{v} - \mathbf{v}_{\text{m}}\|$ as $\mathbf{v}_{\text{m}}$ is quite small here. It is seen that for mesh $m = 64$, Pe is well below one and hence sufficiently fine to guarantee stability, while for meshes $m = 32$, Pe is around one, and for mesh $m = 16$ Pe exceeds one. Mesh $m = 16$ should thus be treated with care, even if no problems were detected in the present results. All other examples above have Péclet numbers below one even for the coarse meshes. On the other hand, the example in Fig. 7a of Sauer (2025) has Pe up to 6.4 for Eulerian mesh motion and shows severe velocity oscillations, but the proposed elastic mesh motion does not show any oscillations, even though Pe goes up to 8.2 for that example. Thus, mesh elasticity may have also a stabilizing influence on the system.

7 Conclusion

This work presented a general surface ALE implementation for area-incompressible, transient Navier-Stokes flow on self-evolving manifolds. The unknowns of the formulation are the material velocity, surface tension (negative in-plane pressure) and mesh motion. The latter is determined from in-plane membrane elasticity in such a way that the material flow is not affected. The formulation achieves optimal convergence rates, as is demonstrated through several examples.

Those include fixed surfaces, free surfaces that remain spherical through balancing surface pressures, and evolving surfaces with large surface changes. If surface stretches are large, it is more accurate to integrate the Dohrmann-Bochev stabilization terms over the current surface

instead of the initial surface. Further, the proposed mesh elasticity formulation tends to be much more accurate than an Eulerian formulation, especially for C^1 -continuous surface discretizations.

Future extensions of this work include the detailed study of C^1 -continuous surface discretizations, bending resistance, and residual-based stabilization, such as streamline upwind Petrov-Galerkin (SUPG) stabilization. Adapting the latter to evolving surface flows would allow to use coarser meshes and thus increase efficiency. Also the pressure stabilizing Petrov-Galerkin (PSPG) scheme could be explored as an alternative to the Dohrmann-Bochev scheme. These extensions would then allow for more detailed studies of evolving soap bubbles, rolling droplets (Sauer and Luginsland, 2018) and budding of lipid membranes (Sauer et al., 2017).

Acknowledgements

The author is grateful to Thang X. Duong for his comments.

A Finite element linearization

This section provides the linearization of finite element force vectors $\mathbf{f}$, $\bar{\mathbf{f}}$ and $\tilde{\mathbf{f}}$. The notation $\mathbf{k} := \partial \mathbf{f} / \partial \mathbf{x}$, $\mathbf{c} := \partial \mathbf{f} / \partial \mathbf{v}$, $\mathbf{m} := \partial \mathbf{f} / \partial \mathbf{v}'$, $\mathbf{d} := \partial \mathbf{f} / \partial \mathbf{q}$ (and likewise for $\bar{\mathbf{f}}$ and $\tilde{\mathbf{f}}$) is used to denote the tangent matrices corresponding to stiffness, viscosity, mass and the Lagrange multiplier. If the mesh motion is known (e.g. for fixed surfaces) the linearization of the mesh “forces” $\tilde{\mathbf{f}}$ is not needed. This is also the case for Lagrangian mesh motion, where $\mathbf{u} := \mathbf{v} - \mathbf{v}_m = \mathbf{0}$.

A.1 Linearization of kinematic variables

Since the shape functions do not change with deformation, the increments $\Delta \mathbf{x}^h$, $\Delta \mathbf{v}^h$, $\Delta \mathbf{a}_\alpha^h$, $\Delta \mathbf{v}_{,\alpha}^h$ and Δq have the same structure as $\mathbf{x}^h$, $\mathbf{v}^h$, $\mathbf{a}_\alpha^h$, $\mathbf{v}_{,\alpha}^h$ and q^h , i.e.

$$\begin{aligned}\Delta \mathbf{x}^h &= \mathbf{N}_e \Delta \mathbf{x}_e, \\ \Delta \mathbf{v}^h &= \mathbf{N}_e \Delta \mathbf{v}_e, \\ \Delta \mathbf{a}_\alpha^h &= \mathbf{N}_{e,\alpha} \Delta \mathbf{x}_e, \\ \Delta \mathbf{v}_{,\alpha}^h &= \mathbf{N}_{e,\alpha} \Delta \mathbf{v}_e, \\ \Delta q^h &= \mathbf{L}_e \Delta \mathbf{q}_e.\end{aligned}\tag{114}$$

The linearization of $\mathbf{a}^\alpha$ yields

$$\Delta \mathbf{a}^\alpha = \mathbf{a}_n^{\alpha\beta} \Delta \mathbf{a}_\beta,\tag{115}$$

with

$$\mathbf{a}_n^{\alpha\beta} := a^{\alpha\beta} \mathbf{n} \otimes \mathbf{n} - \mathbf{a}^\beta \otimes \mathbf{a}^\alpha\tag{116}$$

(Sauer and Duong, 2017). Note that Eqs. (26) and (116) lead to $\mathbf{a}_s^{\alpha\beta} + \mathbf{a}_n^{\alpha\beta} = \mathbf{1} a^{\alpha\beta}$, which offers an alternative way of calculating $\mathbf{a}_n^{\alpha\beta}$ – one that does not require $\mathbf{n}$. The linearization of J_m is needed for the integration (since $da = J_m dA$). The increment ΔJ_m follows analogous to J_m (Sauer, 2025) and hence is

$$\Delta J_m = J_m \mathbf{a}^\beta \cdot \Delta \mathbf{a}_\beta = J_m \mathbf{D}_e \Delta \mathbf{x}_e,\tag{117}$$

due to (114.3) and (64). Therefore

$$\Delta(J_m \mathbf{a}^\alpha) = J_m \mathbf{a}_j^{\alpha\beta} \Delta \mathbf{a}_\beta,\tag{118}$$

with

$$\mathbf{a}_j^{\alpha\beta} := \mathbf{a}_n^{\alpha\beta} + \mathbf{a}^\alpha \otimes \mathbf{a}^\beta.\tag{119}$$

A.2 Linearization of the equation of motion

The FE force vector $\mathbf{f}$ representing the equation of motion consists of the parts derived in Sec. 4.2.1. Its linearization results in the increments

$$\Delta \mathbf{f} = \Delta \mathbf{f}_{\text{in}} + \Delta \mathbf{f}_{\text{intq}} + \Delta \mathbf{f}_{\text{intv}} - \Delta \mathbf{f}_{\text{ext}}. \quad (120)$$

They are discussed in the following.

A.2.1 Linearization of $\mathbf{f}_{\text{in}}^e$

Linearizing the elemental inertia vector $\mathbf{f}_{\text{in}}^e$ specified by (71) and (72) leads to

$$\Delta \mathbf{f}_{\text{in}}^e = \mathbf{m}_e \Delta \mathbf{v}'_e + \mathbf{c}_{\text{in}}^e \Delta \mathbf{v}_e + \mathbf{c}_{\text{m}}^e \Delta \mathbf{x}'_e + \mathbf{k}_{\text{in}}^e \Delta \mathbf{x}_e, \quad (121)$$

with

$$\begin{aligned} \mathbf{m}_e &:= \int_{\Omega^e} \rho \mathbf{N}_e^T \mathbf{N}_e \, da, \\ \mathbf{c}_{\text{m}}^e &:= - \int_{\Omega^e} \rho \mathbf{N}_e^T \nabla_s \mathbf{v} \mathbf{N}_e \, da, \\ \mathbf{c}_{\text{in}}^e &:= \int_{\Omega^e} \rho \mathbf{N}_e^T u^\alpha \mathbf{N}_{e,\alpha} \, da - \mathbf{c}_{\text{m}}^e, \\ \mathbf{k}_{\text{in}}^e &:= \int_{\Omega^e} \rho \mathbf{N}_e^T \mathbf{u}_x^\beta \mathbf{N}_{e,\beta} \, da + \int_{\Omega^e} \rho \mathbf{N}_e^T \mathbf{v}' \otimes \mathbf{a}^\beta \mathbf{N}_{e,\beta} \, da, \end{aligned} \quad (122)$$

where $\nabla_s \mathbf{v}$ and $u^\alpha = \dot{\zeta}^\alpha$ follow from (22), (60) and (58), and

$$\mathbf{u}_x^\beta := \mathbf{v}_{,\alpha} \otimes (a^{\alpha\beta} u \mathbf{n} - \mathbf{a}^\alpha u^\beta + u^\alpha \mathbf{a}^\beta) \quad (123)$$

comes from linearizing $u^\alpha \, da$. The first term in $\mathbf{u}_x^\beta$ vanishes in theory, as $u = 0$. But this may not be the case in the discretized system, especially not during Newton-Raphson iteration. The last term in $\mathbf{u}_x^\beta$ comes from linearizing da . Similarly, the second term in $\mathbf{k}_{\text{in}}^e$ comes from linearizing da in mass matrix $\mathbf{m}_e$.

A.2.2 Linearization of $\mathbf{f}_{\text{int}}^e$

Applying (114) and (118) to (75) directly leads to

$$\begin{aligned} \Delta \mathbf{f}_{\text{intq}}^e &= \mathbf{d}_{\text{int}}^e \Delta \mathbf{q}_e + \mathbf{k}_{\text{intq}}^e \Delta \mathbf{x}_e, \\ \Delta \mathbf{f}_{\text{intv}}^e &= \mathbf{c}_{\text{int}}^e \Delta \mathbf{v}_e + \mathbf{k}_{\text{intv}}^e \Delta \mathbf{x}_e, \end{aligned} \quad (124)$$

with

$$\begin{aligned} \mathbf{d}_{\text{int}}^e &:= \int_{\Omega^e} \mathbf{D}_e^T \mathbf{L}_e \, da, \\ \mathbf{c}_{\text{int}}^e &:= \int_{\Omega^e} \eta \mathbf{N}_{e,\alpha}^T \mathbf{a}_s^{\alpha\beta} \mathbf{N}_{e,\beta} \, da, \\ \mathbf{k}_{\text{intq}}^e &:= \int_{\Omega^e} q \mathbf{N}_{e,\alpha}^T \mathbf{a}_j^{\alpha\beta} \mathbf{N}_{e,\beta} \, da, \\ \mathbf{k}_{\text{intv}}^e &:= \int_{\Omega^e} 2\eta \mathbf{N}_{e,\alpha}^T \mathbf{d}_x^{\alpha\beta} \mathbf{N}_{e,\beta} \, da, \end{aligned} \quad (125)$$

where $\mathbf{d}_x^{\alpha\beta}$ is defined by

$$\Delta(J_{\text{m}} \mathbf{a}_s^{\alpha\beta}) \mathbf{v}_{,\beta} = 2J_{\text{m}} \mathbf{d}_x^{\alpha\beta} \Delta \mathbf{a}_\beta \quad (126)$$

and results in

$$2\mathbf{d}_x^{\alpha\beta} = \mathbf{a}_s^{\alpha\beta} (\nabla_s \mathbf{v}^T (\mathbf{n} \otimes \mathbf{n}) - \nabla_s \mathbf{v}) + 2\mathbf{a}_n^{\alpha\gamma} d_\gamma^\beta + 2d^{\alpha\gamma} \mathbf{a}_\gamma \otimes \mathbf{a}^\beta. \quad (127)$$

The last part in $\mathbf{d}_x^{\alpha\beta}$ comes from the linearization of J_m .

A.2.3 Linearization of $\mathbf{f}_{\text{ext}}^e$

In this work, the external force vector $\mathbf{f}_{\text{ext}}^e$ contains the two solution-dependent contributions $\mathbf{f}_p^e$ and $\mathbf{f}_T^e$, i.e. $\mathbf{f}_{\text{ext}}^e = \mathbf{f}_p^e + \mathbf{f}_T^e$. The linearization of $\mathbf{f}_{\text{ext}}^e$ produces the three tangent contributions $\mathbf{k}_{\text{ext}}^e = \mathbf{k}_p^e + \mathbf{k}_T^e$, $\mathbf{c}_{\text{ext}}^e = \mathbf{c}_p^e$ and $\mathbf{d}_{\text{ext}}^e = \mathbf{d}_T^e$ derived in the following.

A.2.3.1 Surface pressure

The first solution-dependent external force that is present in all examples is surface pressure. It is captured by the FE force vector

$$\mathbf{f}_p^e := \int_{\Omega^e} \mathbf{N}_e^T p \mathbf{n} \, da. \quad (128)$$

Here, $p = \bar{p} + p_{\text{visc}}$ contains the prescribed constant pressure $\bar{p}$ and the viscous pressure from Eq. (77). The linearization of the two parts is given in Sauer et al. (2014) and Sauer (2016). Together, they give

$$\Delta \mathbf{f}_p^e = \mathbf{c}_p^e \Delta \mathbf{v}_e + \mathbf{k}_p^e \Delta \mathbf{x}_e, \quad (129)$$

with the tangent matrices

$$\begin{aligned} \mathbf{c}_p^e &:= - \int_{\Omega^e} \eta_n \mathbf{N}_e^T (\mathbf{n} \otimes \mathbf{n}) \mathbf{N}_e \, da, \\ \mathbf{k}_p^e &:= \int_{\Omega^e} \mathbf{N}_e^T \left[p (\mathbf{n} \otimes \mathbf{a}^\alpha - \mathbf{a}^\alpha \otimes \mathbf{n}) + \eta_n \mathbf{n} \otimes \mathbf{n} v^\alpha \right] \mathbf{N}_{e,\alpha} \, da, \end{aligned} \quad (130)$$

due to (117) and

$$\Delta \mathbf{n} = -(\mathbf{a}^\alpha \otimes \mathbf{n}) \Delta \mathbf{a}_\alpha \quad (131)$$

(Sauer et al., 2014). Note, that these $\mathbf{c}_p^e$ and $\mathbf{k}_p^e$ need to be subtracted from the other $\mathbf{c}$ and $\mathbf{k}$ due to the minus sign in (120).

A.2.3.2 Boundary forces in the soap bubble example

The soap bubble example of Sec. 6.3 contains a free boundary, where the two solution-dependent boundary conditions $v_n := \mathbf{v} \cdot \mathbf{n} = 0$ and $\mathbf{T} = \bar{\mathbf{T}} := q \boldsymbol{\nu}$ are prescribed

The first BC is enforced by the penalty method. Thus the surface pressure $p_\epsilon = -\epsilon \mathbf{v} \cdot \mathbf{n}$ is applied along the boundary. This is done as described in Eqs. (128)-(130), replacing pressure p by p_ϵ and integration domain Ω^e by Γ^e , as the boundary condition is only enforced along the line elements along the boundary. However, the whole surface element Ω^e needs to be sampled in order to evaluate the normal vector $\mathbf{n}$.

The second BC leads to the external FE force vector

$$\mathbf{f}_T^e := \int_{\Gamma^e} \mathbf{N}_e^T q \boldsymbol{\nu} \, ds \quad (132)$$

according to (76). Here $q = \mathbf{L}_e \mathbf{q}_e$ according to Eq. (67). Since the bubble is pinned, Γ^e and ds are fixed. Thus the linearization of (132) leads to

$$\Delta \mathbf{f}_T^e = \mathbf{d}_T^e \Delta \mathbf{q}_e + \mathbf{k}_T^e \Delta \mathbf{x}_e, \quad (133)$$

with

$$\begin{aligned}\mathbf{d}_T^e &:= \int_{\Gamma^e} \mathbf{N}_e^T \boldsymbol{\nu} \mathbf{L}_e \, ds, \\ \mathbf{k}_T^e &:= - \int_{\Gamma^e} \mathbf{N}_e^T q (\boldsymbol{\tau} \times \mathbf{a}^\alpha) \mathbf{n} \cdot \mathbf{N}_{e,\alpha} \, ds.\end{aligned}\tag{134}$$

The latter expression follows from

$$\Delta \boldsymbol{\nu} = -(\boldsymbol{\tau} \times \mathbf{a}^\alpha)(\mathbf{n} \cdot \Delta \mathbf{a}_\alpha),\tag{135}$$

which in turn follows from $\boldsymbol{\nu} = \boldsymbol{\tau} \times \mathbf{n}$ and (131) for fixed $\boldsymbol{\tau}$. It is noted that along the boundary one of the $\mathbf{a}^\alpha$ ($\alpha = 1, 2$) is typically parallel to $\boldsymbol{\tau}$, such that only the other $\mathbf{a}^\alpha$ appears within $\mathbf{k}_T^e$.

A.3 Linearization of $\bar{\mathbf{f}}^e$

According to (78),

$$\Delta \bar{\mathbf{f}}^e = \Delta \bar{\mathbf{f}}_{\text{div}}^e - \Delta \bar{\mathbf{f}}_{\text{DB}}^e.\tag{136}$$

From (63), (64) and (118) follows

$$\Delta (J_m \operatorname{div}_s^h \mathbf{v}^h) = J_m \mathbf{D}_e \Delta \mathbf{v}_e + J_m \mathbf{v}_{,\alpha} \cdot \mathbf{a}_J^{\alpha\beta} \mathbf{N}_{e,\beta} \Delta \mathbf{x}_e.\tag{137}$$

Thus (79.1) gives

$$\Delta \bar{\mathbf{f}}_{\text{div}}^e = \bar{\mathbf{c}}_e \Delta \mathbf{v}_e + \bar{\mathbf{k}}_e \Delta \mathbf{x}_e,\tag{138}$$

with

$$\begin{aligned}\bar{\mathbf{c}}_e &:= \int_{\Omega^e} \mathbf{L}_e^T \mathbf{D}_e \, da, \\ \bar{\mathbf{k}}_e &:= \int_{\Omega^e} \mathbf{L}_e^T \mathbf{v}_{,\alpha} \cdot \mathbf{a}_J^{\alpha\beta} \mathbf{N}_{e,\beta} \, da,\end{aligned}\tag{139}$$

confirming the usual symmetry $\bar{\mathbf{c}}_e^T = \mathbf{d}_{\text{int}}^e$. Also $\bar{\mathbf{k}}_e$ and $\mathbf{k}_{\text{int}q}^e$ exhibit a certain symmetry. Using (117) and $\Delta(\mathbf{h}_e^{-1}) = -\mathbf{h}_e^{-1} \Delta \mathbf{h}_e \mathbf{h}_e^{-1}$, the linearization of (83) leads to

$$\Delta \bar{\mathbf{f}}_{\text{DB}}^e = \bar{\mathbf{d}}_e \Delta \mathbf{q}_e + \bar{\mathbf{k}}_{\text{DB}}^e \Delta \mathbf{x}_e,\tag{140}$$

with

$$\begin{aligned}\bar{\mathbf{k}}_{\text{DB}}^e &:= \frac{\alpha_{\text{DB}}}{\eta} \left[\int_{\Omega^e} q \mathbf{L}_e^T \mathbf{D}_e \, da - \mathbf{g}_e^T \mathbf{h}_e^{-1} \int_{\Omega^e} q \check{\mathbf{L}}_e^T \mathbf{D}_e \, da \right. \\ &\quad \left. - \sum_{I=1}^{\tilde{m}_e} \left(\int_{\Omega^e} \mathbf{L}_e^T \mathbf{D}_e \check{L}_I \, da - \mathbf{g}_e^T \mathbf{h}_e^{-1} \int_{\Omega^e} \check{\mathbf{L}}_e^T \mathbf{D}_e \check{L}_I \, da \right) \check{q}_I \right],\end{aligned}\tag{141}$$

where $\check{L}_I$ and $\check{q}_I$ are the I -th components of vectors $\check{\mathbf{L}}_e$ and $\check{\mathbf{q}}_e = \mathbf{h}_e^{-1} \mathbf{g}_e \mathbf{q}_e$.

A.4 Linearization of $\tilde{\mathbf{f}}^e$

A.4.1 Eulerian mesh motion

From (85) and (131) follows

$$\Delta \tilde{\mathbf{f}}^e := \tilde{\mathbf{m}}_e \Delta \mathbf{x}'_e - \tilde{\mathbf{n}}_e \Delta \mathbf{v}_e + \tilde{\mathbf{n}}_e^v \Delta \mathbf{x}_e,\tag{142}$$

with

$$\tilde{\mathbf{n}}_e^v := \int_{\Omega_0^e} \mathbf{N}_e^T (v \mathbf{a}^\alpha \otimes \mathbf{n} + v^\alpha \mathbf{n} \otimes \mathbf{n}) \mathbf{N}_{e,\alpha} \, dA.\tag{143}$$

A.4.2 Elastic mesh motion

From (88) and (89) follows

$$\Delta \tilde{\mathbf{f}}^e := \Delta \tilde{\mathbf{f}}_i^e + \Delta \tilde{\mathbf{f}}_o^e, \quad (144)$$

with

$$\begin{aligned} \Delta \tilde{\mathbf{f}}_i^e &:= \tilde{\mathbf{k}}_i^e \Delta \mathbf{x}_e \\ \Delta \tilde{\mathbf{f}}_o^e &:= \tilde{\mathbf{n}}_e (\Delta \mathbf{x}'_e - \Delta \mathbf{v}_e) + \tilde{\mathbf{n}}_e^u \Delta \mathbf{x}_e. \end{aligned} \quad (145)$$

Here,

$$\begin{aligned} \tilde{\mathbf{k}}_i^e &:= \int_{\Omega_0^e} \mathbf{N}_{e,\alpha}^T (1 \tau_m^{\alpha\beta} + \mu_m \mathbf{a}_s^{\alpha\beta}) \mathbf{N}_{e,\beta} dA + \int_{\Omega_0^e} \mathbf{N}_e^T (\mathbf{n} \otimes \mathbf{n}) \tau_m^{\alpha\beta} \mathbf{N}_{e,\alpha\beta} dA \\ &+ \int_{\Omega_0^e} \mathbf{N}_e^T (2\mu_m b_\alpha^\gamma \mathbf{n} \otimes \mathbf{a}^\alpha - \tau_m^{\alpha\beta} b_{\alpha\beta} \mathbf{a}^\gamma \otimes \mathbf{n}) \mathbf{N}_{e,\gamma} dA \end{aligned} \quad (146)$$

follows from (131),

$$\Delta b_{\alpha\beta} = \mathbf{n} \cdot \mathbf{N}_{;\alpha\beta} \Delta \mathbf{x}_e, \quad \mathbf{N}_{;\alpha\beta} := \mathbf{N}_{,\alpha\beta} - \Gamma_{\alpha\beta}^\gamma \mathbf{N}_{e,\gamma}, \quad (147)$$

(Duong et al., 2017) and

$$\Delta \tau_m^{\alpha\beta} = c_m^{\alpha\beta\gamma\delta} \mathbf{a}_\gamma \cdot \Delta \mathbf{a}_\delta, \quad c_m^{\alpha\beta\gamma\delta} := \mu_m (a^{\alpha\gamma} a^{\beta\delta} + a^{\alpha\delta} a^{\beta\gamma}), \quad (148)$$

(Sauer and Duong, 2017), while $\tilde{\mathbf{n}}_e^u$ is analogous to $\tilde{\mathbf{n}}_e^v$ from (143) – simply $\mathbf{v}$ is replaced by $\mathbf{u} := \mathbf{v} - \mathbf{v}_m$ in (143), i.e.

$$\tilde{\mathbf{n}}_e^u := \int_{\Omega_0^e} \mathbf{N}_e^T (u \mathbf{a}^\alpha \otimes \mathbf{n} + u^\alpha \mathbf{n} \otimes \mathbf{n}) \mathbf{N}_{e,\alpha} dA. \quad (149)$$

As noted above, $u^\alpha = \dot{\zeta}^\alpha$ due to (15), but even though $u = 0$ in theory, u can become non-zero in the discrete system. It is noted that $\tilde{\mathbf{k}}_i^e$ is equal to the more lengthy tangent expression given in Appendix A.1 of Sauer (2014).

B Tangent matrices of the time discretized system

This section provides the FE tangent matrices for the monolithic solution of the coupled system in Sec. 4.4. This is done in the framework of the generalized- α method (Chung and Hulbert, 1993; Jansen et al., 2000), even though all numerical examples use $\alpha_f = \alpha_m = 1$. The generalized- α method simply evaluates the current residual $\mathbf{r}_{n+1}$ at the asynchronous arguments

$$\begin{aligned} \mathbf{x}_{n+\alpha_f} &= \alpha_f \mathbf{x}_{n+1} + (1 - \alpha_f) \mathbf{x}_n, \\ \mathbf{v}_{n+\alpha_f} &= \alpha_f \mathbf{v}_{n+1} + (1 - \alpha_f) \mathbf{v}_n, \\ \mathbf{a}_{n+\alpha_m} &= \alpha_m \mathbf{a}_{n+1} + (1 - \alpha_m) \mathbf{a}_n, \end{aligned} \quad (150)$$

and so simply an extra factor of α is picked up in the corresponding linearization.

The linearization of (92) leads to (reducing n by 1)

$$\Delta \mathbf{a}_n = \Delta \mathbf{v}'_n = \frac{1}{\gamma \Delta t_n} \Delta \mathbf{v}_n, \quad \Delta \mathbf{v}_m^n = \frac{1}{\gamma \Delta t_n} \Delta \mathbf{x}_n. \quad (151)$$

With this and Appendix A, one can obtain the elemental blocks of $\mathbf{K}$ in (96). It is noted that indices e and n are placed wherever convenient. The first three blocks of $\mathbf{K}$ in (96) follow from Eqs. (121), (124), (129) and (151), as

$$\mathbf{k}_{\text{vv}}^{en} := \frac{\partial \mathbf{f}_n^e}{\partial \mathbf{v}_n} = \frac{\alpha_m}{\gamma \Delta t_n} \mathbf{m}_e^n + \alpha_f (\mathbf{c}_{\text{in}}^{en} + \mathbf{c}_{\text{int}}^{en} - \mathbf{c}_{\text{ext}}^{en}), \quad (152)$$

$$\mathbf{k}_{\text{vx}}^{en} := \frac{\partial \mathbf{f}_n^e}{\partial \mathbf{x}_n} = \frac{\alpha_f}{\gamma \Delta t_n} \mathbf{c}_m^{en} + \alpha_f (\mathbf{k}_{\text{in}}^{en} + \mathbf{k}_{\text{intq}}^{en} + \mathbf{k}_{\text{intv}}^{en} - \mathbf{k}_{\text{ext}}^{en}), \quad (153)$$

and

$$\mathbf{k}_{\text{vq}}^{en} := \frac{\partial \mathbf{f}_n^e}{\partial \mathbf{q}_n} = \mathbf{d}_{\text{int}}^{en} - \mathbf{d}_{\text{ext}}^{en}. \quad (154)$$

The next three blocks follow from Eqs. (142) and (144) as

$$\mathbf{k}_{\text{xv}}^{en} := \frac{\partial \tilde{\mathbf{f}}_n^e}{\partial \mathbf{v}_n} = -\alpha_f \tilde{\mathbf{n}}_e^n, \quad (155)$$

$$\mathbf{k}_{\text{xx}}^{en} := \frac{\partial \tilde{\mathbf{f}}_n^e}{\partial \mathbf{x}_n} = \frac{\alpha_f}{\gamma \Delta t_n} \begin{Bmatrix} \tilde{\mathbf{m}}_e \\ \tilde{\mathbf{n}}_e \end{Bmatrix}_n + \alpha_f \begin{Bmatrix} \tilde{\mathbf{n}}_e^v \\ \tilde{\mathbf{k}}_i^e + \tilde{\mathbf{n}}_e^u \end{Bmatrix}_n, \quad (156)$$

and

$$\mathbf{k}_{\text{xq}}^{en} := \frac{\partial \tilde{\mathbf{f}}_n^e}{\partial \mathbf{q}_n} = \mathbf{0}. \quad (157)$$

Here the top entries in the curly brackets are for Eulerian mesh motion (option I), while the lower entries are for elastic mesh motion (option II). The last three blocks follow from Eq. (136) as

$$\mathbf{k}_{\text{qv}}^{en} := \frac{\partial \bar{\mathbf{f}}_n^e}{\partial \mathbf{v}_n} = \alpha_f \bar{\mathbf{c}}_e^n = \mathbf{k}_{\text{vq}}^{enT}, \quad (158)$$

$$\mathbf{k}_{\text{qx}}^{en} := \frac{\partial \bar{\mathbf{f}}_n^e}{\partial \mathbf{x}_n} = \alpha_f (\bar{\mathbf{k}}_e^n - \bar{\mathbf{k}}_{\text{DB}}^{en}), \quad (159)$$

and

$$\mathbf{k}_{\text{qq}}^{en} := \frac{\partial \bar{\mathbf{f}}_n^e}{\partial \mathbf{q}_n} = -\bar{\mathbf{d}}_e^n. \quad (160)$$

Remark B.1: If the mesh motion is known, e.g. by prescribing either $\mathbf{x}$ or $\mathbf{v}_m$, the corresponding blocks in $\mathbf{K}$ are simply eliminated in the global system.

Remark B.2: Using (151), the unknowns can be easily exchanged. E.g. replacing $\mathbf{x}$ by $\mathbf{v}_m$ simply changes

$$\begin{bmatrix} \mathbf{K}_{\text{vv}} & \mathbf{K}_{\text{vx}} & \mathbf{K}_{\text{vq}} \\ \mathbf{K}_{\text{xv}} & \mathbf{K}_{\text{xx}} & \mathbf{K}_{\text{xq}} \\ \mathbf{K}_{\text{qv}} & \mathbf{K}_{\text{qx}} & \mathbf{K}_{\text{qq}} \end{bmatrix}_n \begin{bmatrix} \Delta \mathbf{v} \\ \Delta \mathbf{x} \\ \Delta \mathbf{q} \end{bmatrix}_{n+1} = - \begin{bmatrix} \mathbf{f} \\ \tilde{\mathbf{f}} \\ \bar{\mathbf{f}} \end{bmatrix}_n \quad (161)$$

into

$$\begin{bmatrix} \mathbf{K}_{\text{vv}} & \gamma \Delta t_n \mathbf{K}_{\text{vx}} & \mathbf{K}_{\text{vq}} \\ \mathbf{K}_{\text{xv}} & \gamma \Delta t_n \mathbf{K}_{\text{xx}} & \mathbf{K}_{\text{xq}} \\ \mathbf{K}_{\text{qv}} & \gamma \Delta t_n \mathbf{K}_{\text{qx}} & \mathbf{K}_{\text{qq}} \end{bmatrix}_n \begin{bmatrix} \Delta \mathbf{v} \\ \Delta \mathbf{v}_m \\ \Delta \mathbf{q} \end{bmatrix}_{n+1} = - \begin{bmatrix} \mathbf{f} \\ \tilde{\mathbf{f}} \\ \bar{\mathbf{f}} \end{bmatrix}_n. \quad (162)$$

C Additional numerical results for fixed surfaces

This appendix shows the results for load cases 2 and 4 from the study of Sec. 5.2, see Table 2. Load case 2, shown in Fig. 29, is the case where the normal surface velocity v_n is prescribed by a Dirichlet boundary condition. Load case 4, shown in Fig. 30, is the case where the normal surface velocity v_n adjusts to the Neuman boundary condition on the surface pressure p . The convergence rate for q does not look quite so good in this load case, which could be an effect of too low α_{DB} .

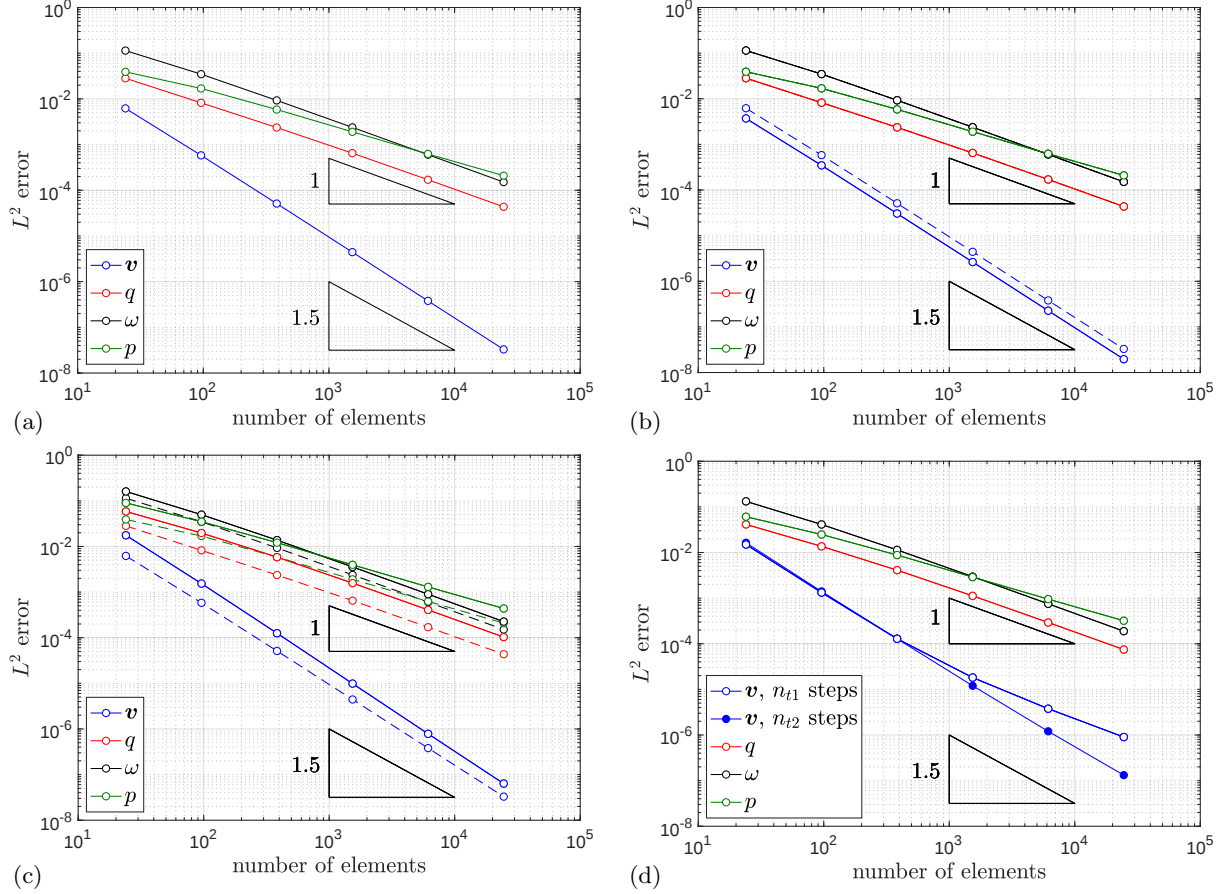


Figure 29: Simple shear flow on a rigid sphere for load case 2: Convergence of velocity, surface tension and vorticity for the four ALE cases (a-d) in the L^2 error norm. Optimal convergence rates are obtained for all fields. The dashed lines in (b) & (c) are the results from (a).

References

- Al-Izzi, S. C. and Morris, R. G. (2023). Morphodynamics of active nematic fluid surfaces. *J. Fluid Mech.*, **957**:A4.
- Arroyo, M. and DeSimone, A. (2009). Relaxation dynamics of fluid membranes. *Phys. Rev. E*, **79**:031915.
- Babuška, I. (1973). The finite element method with Lagrangian multipliers. *Num. Math.*, **20**:179–192.
- Bachini, E., Krause, V., Nitschke, I., and Voigt, A. (2023). Derivation and simulation of a two-phase fluid deformable surface model. *J. Fluid Mech.*, **977**:A41.
- Barrett, J. W., Garcke, H., and Nürnberg, R. (2008a). On the parametric finite element approximation of evolving hypersurfaces in $\mathbb{R}^3$. *J. Comput. Phys.*, **227**(9):4281–4307.
- Barrett, J. W., Garcke, H., and Nürnberg, R. (2008b). Parametric approximation of Willmore flow and related geometric evolution equations. *SIAM J. Sci. Comput.*, **31**(1):225–253.
- Barrett, J. W., Garcke, H., and Nürnberg, R. (2015). Numerical computations of the dynamics of fluidic membranes and vesicles. *Phys. Rev. E*, **92**:052704.
- Barrett, J. W., Garcke, H., and Nürnberg, R. (2016). A stable numerical method for the dynamics of fluidic membranes. *Numer. Math.*, **134**:783–822.

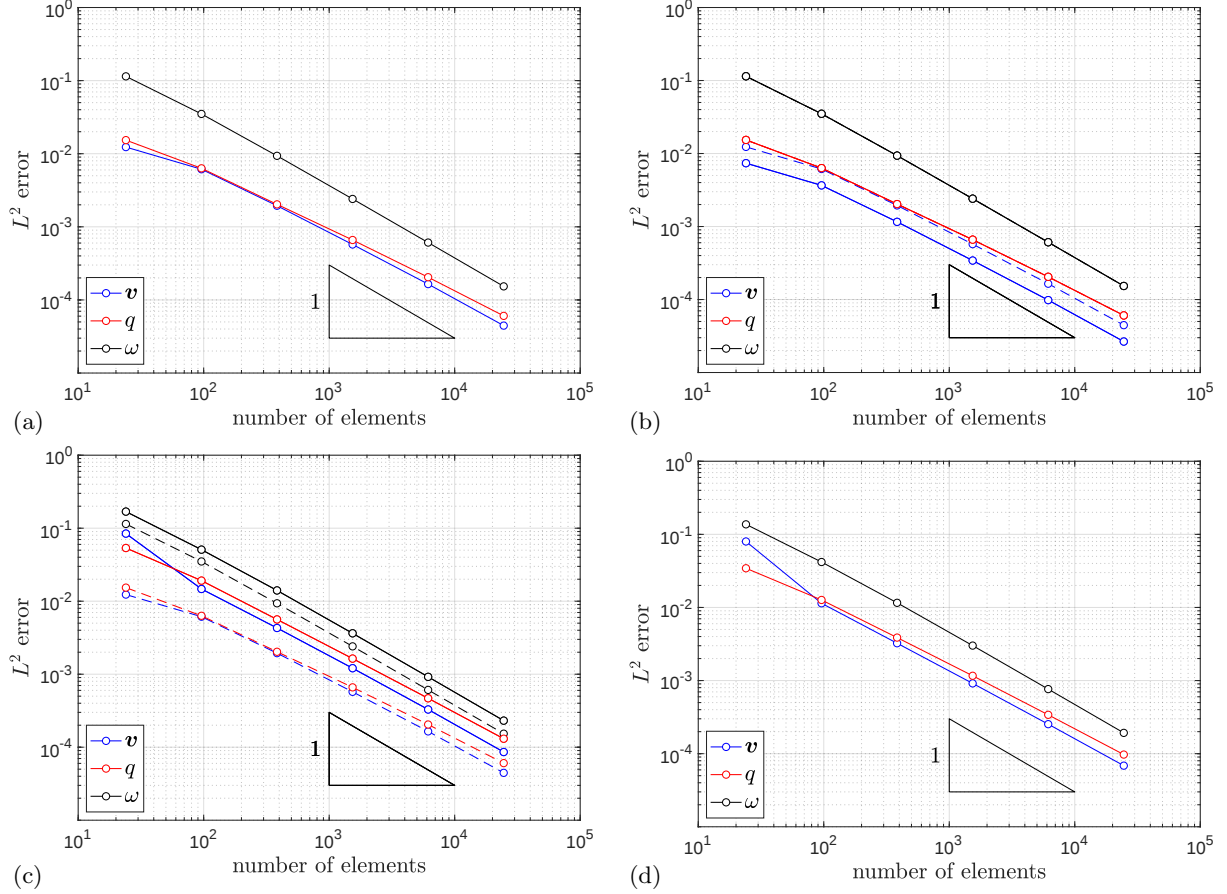


Figure 30: Simple shear flow on a rigid sphere for load case 4: Convergence of velocity, surface tension and vorticity for the four ALE cases (a-d) in the L^2 error norm. Nearly optimal convergence rates are obtained for all fields.

Bharadwaj, A. S., Kuhnert, J., Bordas, S. P., and Suchde, P. (2022). A discrete droplet method for modelling thin film flows. *Appl. Math. Model.*, **112**:486–504.

Bonito, A., Demlow, A., and Licht, M. (2020). A divergence-conforming finite element method for the Stokes equation. *SIAM J. Numer. Anal.*, **58**(5):2764–2798.

Brakke, K. A. (1992). The surface evolver. *Experimental Mathematics*, **1**(2):141–165.

Brezzi, F. (1974). On the existence, uniqueness and approximation of saddle-point problems arising from Lagrangian multipliers. *Publications des s minaires de math matiques et informatique de Rennes*, **S4**:1–26.

Brooks, A. N. and Hughes, T. J. R. (1982). Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. *Adv. Appl. Mech.*, **32**:199–259.

Brown, R. A., Orr, F. M., and Scriven, L. E. (1980). Static drop on an inclined plate: Analysis by the finite element method. *J. Colloid Interface Sci.*, **73**(1):76–87.

Chung, J. H. and Hulbert, G. M. (1993). A time integration algorithm for structural dynamics with improved numerical dissipation: the generalized- α method. *J. Appl. Mech.*, **60**:371–375.

de Kinkelder, E., Sagis, L., and Aland, S. (2021). A numerical method for the simulation of viscoelastic fluid surfaces. *J. Comput. Phys.*, **440**:110413.

- Dharmavaram, S. (2021). A gauge-fixing procedure for spherical fluid membranes and application to computations. *Comput. Methods Appl. Mech. Eng.*, **381**:113849.
- Dharmavaram, S., Wan, X., and Perotti, L. E. (2022). A Lagrangian thin-shell finite element method for interacting particles on fluid membranes. *Membranes*, **12**:960.
- Dohrmann, C. R. and Bochev, P. B. (2004). A stabilized finite element method for the Stokes problem based on polynomial pressure projections. *Int. J. Numer. Methods Fluids*, **46**:183–201.
- Donea, J. and Huerta, A. (2003). *Finite Element Methods for Flow Problems*. Wiley, Hoboken.
- Donea, J., Huerta, A., Ponthot, J.-P., and Rodríguez-Ferran, A. (2004). Arbitrary Lagrangian-Eulerian methods. In Stein, E., de Borst, R., and Hughes, T. J. R., editors, *Encyclopedia of Computational Mechanics. Vol. 1: Fundamentals. Chapter 14*. Wiley.
- Duong, T. X., Roohbakhshan, F., and Sauer, R. A. (2017). A new rotation-free isogeometric thin shell formulation and a corresponding continuity constraint for patch boundaries. *Comput. Methods Appl. Mech. Eng.*, **316**:43–83.
- Dziuk, G. (1991). An algorithm for evolutionary surfaces. *Numer. Math.*, **58**:603–611.
- Dziuk, G. (2008). Computational parametric Willmore flow. *Numer. Math.*, **111**:55–80.
- Dziuk, G. and Elliott, C. M. (2013). Finite element methods for surface PDEs. *Acta Numerica*, **22**:289–396.
- Elliott, C. M. and Mavrakakis, A. (2024). SFEM for the Lagrangian formulation of the surface Stokes problem. *arXiv*, 2410.19470.
- Elliott, C. M. and Stinner, B. (2013). Computation of two-phase biomembranes with phase dependent material parameters using surface finite elements. *Comm. Comp. Phys.*, **13**(2):325–360.
- Feng, F. and Klug, W. S. (2006). Finite element modeling of lipid bilayer membranes. *J. Comput. Phys.*, **220**:394–408.
- Fries, T.-P. (2018). Higher-order surface FEM for incompressible Navier-Stokes flows on manifolds. *Int. J. Numer. Methods Fluids*, **88**:55–78.
- Gross, B. J. and Atzberger, P. J. (2018). Hydrodynamic flows on curved surfaces: Spectral numerical methods for radial manifold shapes. *J. Comput. Phys.*, **371**:663–689.
- Gross, B. J., Trask, N., Kuberry, P., and Atzberger, P. J. (2020). Meshfree methods on manifolds for hydrodynamic flows on curved surfaces: A Generalized Moving Least-Squares (GMLS) approach. *J. Comput. Phys.*, **409**:109340.
- Hardering, H. and Praetorius, S. (2024). Parametric finite-element discretization of the surface stokes equations: inf-sup stability and discretization error analysis. *IMA J. Numer. Anal.*, drae080.
- Hughes, T. J. R., Franca, L. P., and Balestra, M. (1986). 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. *Comput. Methods Appl. Mech. Eng.*, **59**(1):85–99.

- Jankuhn, T., Olshanskii, M. A., Reusken, A., and Zhiliakov, A. (2021). Error analysis of higher order trace finite element methods for the surface Stokes equations. *J. Numer. Math.*, **29**(3):245–267.
- Jansen, K. E., Whiting, C. H., and Hulbert, G. M. (2000). A generalized- α method for integrating the filtered Navier-Stokes equations with a stabilized finite element method. *Comput. Methods Appl. Mech. Eng.*, **190**:305–319.
- Krause, V. and Voigt, A. (2023). A numerical approach for fluid deformable surfaces with conserved enclosed volume. *J. Comput. Phys.*, **486**:112097.
- Krause, V. and Voigt, A. (2024). Wrinkling of fluid deformable surfaces. *J. R. Soc. Interface*, **21**(216):20240056.
- Ladyzhenskaya, O. A. (1969). *The Mathematical Theory of Viscous Incompressible Flow*, volume 12. Gordon & Breach, New York.
- Lederer, P. L., Lehrenfeld, C., and Schöberl, J. (2020). Divergence-free tangential finite element methods for incompressible flows on surfaces. *Int. J. Numer. Meth. Eng.*, **121**(11):2503–2533.
- Ma, L. and Klug, W. S. (2008). Viscous regularization and r-adaptive meshing for finite element analysis of lipid membrane mechanics. *J. Comput. Phys.*, **227**:5816–5835.
- Nitschke, I., Reuther, S., and Voigt, A. (2017). Discrete exterior calculus (DEC) for the surface Navier-Stokes equation. In Bothe, D. and Reusken, A., editors, *Transport Processes at Fluidic Interfaces*, pages 177–197. Birkhäuser, Cham.
- Nitschke, I., Voigt, A., and Wensch, J. (2012). A finite element approach to incompressible two-phase flow on manifolds. *J. Fluid Mech.*, **708**:418–438.
- Oden, J. T. and Sato, T. (1967). Finite strains and displacements of elastic membranes by the finite element method. *Int. J. Solids Struct.*, **3**(4):471–488.
- Olshanskii, M. A., Quaini, A., Reusken, A., and Yushutin, V. (2018). A finite element method for the surface Stokes problem. *SIAM J. Sci. Comput.*, **40**:A2429–A2518.
- Olshanskii, M. A. and Reusken, A. (2018). Trace finite element methods for PDEs on surfaces. In Bordas, S., Burman, E., Larson, M., and Olshanskii, M., editors, *Geometrically Unfitted Finite Element Methods and Applications*, pages 211–258. Springer, Cham.
- Olshanskii, M. A., Reusken, A., and Schwing, P. (2024). An Eulerian finite element method for tangential Navier-Stokes equations on evolving surfaces. *Math. Comp.*, **93**:2031–2065.
- Rahimi, M. and Arroyo, M. (2012). Shape dynamics, lipid hydrodynamics, and the complex viscoelasticity of bilayer membranes. *Phys. Rev. E*, **86**:011932.
- Rangamani, P., Agrawal, A., Mandadapu, K. K., Oster, G., and Steigmann, D. J. (2013). Interaction between surface shape and intra-surface viscous flow on lipid membranes. *Biomech. Model. Mechanobiol.*, **12**(4):833–845.
- Rangarajan, R. and Gao, H. (2015). A finite element method to compute three-dimensional equilibrium configurations of fluid membranes: Optimal parameterization, variational formulation and applications. *J. Comput. Phys.*, **297**:266–294.
- Rank, M. and Voigt, A. (2021). Active flows on curved surfaces. *Phys. Fluids*, **33**:072110.

- Reusken, A. (2025). Analysis of the Taylor-Hood surface finite element method for the surface Stokes equations. *Math. Comput.*, **94**:1701–1719.
- Reuther, S., Nitschke, I., and Voigt, A. (2020). A numerical approach for fluid deformable surfaces. *J. Fluid Mech.*, **900**:R8.
- Reuther, S. and Voigt, A. (2015). The interplay of curvature and vortices in flow on curved surfaces. *Multiscale Model. Simul.*, **13**(2):632–643.
- Reuther, S. and Voigt, A. (2018). Solving the incompressible surface Navier-Stokes equation by surface finite elements. *Phys. Fluids.*, **30**:012107.
- Rower, D. A., Padidar, M., and J.Atzberger, P. (2022). Surface fluctuating hydrodynamics methods for the drift-diffusion dynamics of particles and microstructures within curved fluid interfaces. *J. Comput. Phys.*, **455**:110994.
- Sahu, A. (2024). Arbitrary Lagrangian-Eulerian finite element method for lipid membranes. *arXiv*, 2412.07596.
- Sahu, A., Omar, Y. A. D., Sauer, R. A., and Mandadapu, K. K. (2020). Arbitrary Lagrangian-Eulerian finite element method for curved and deforming surfaces: I. General theory and application to fluid interfaces. *J. Comput. Phys.*, **407**:109253.
- Sane, A., Mandre, S., and Kim, I. (2017). Surface tension of flowing soap films. *J. Fluid Mech.*, **841**:R2.
- Sauer, R. A. (2014). Stabilized finite element formulations for liquid membranes and their application to droplet contact. *Int. J. Numer. Methods Fluids*, **75**(7):519–545.
- Sauer, R. A. (2016). A frictional sliding algorithm for liquid droplets. *Comput. Mech.*, **58**(6):937–956.
- Sauer, R. A. (2025). A curvilinear surface ALE formulation for self-evolving Navier-Stokes manifolds – General theory and analytical solutions. *J. Fluid Mech.*, **1016**:A34.
- Sauer, R. A. and Duong, T. X. (2017). On the theoretical foundations of solid and liquid shells. *Math. Mech. Solids*, **22**(3):343–371.
- Sauer, R. A., Duong, T. X., and Corbett, C. J. (2014). A computational formulation for constrained solid and liquid membranes considering isogeometric finite elements. *Comput. Methods Appl. Mech. Eng.*, **271**:48–68.
- Sauer, R. A., Duong, T. X., Mandadapu, K. K., and Steigmann, D. J. (2017). A stabilized finite element formulation for liquid shells and its application to lipid bilayers. *J. Comput. Phys.*, **330**:436–466.
- Sauer, R. A. and Luginsland, T. (2018). A monolithic fluid-structure interaction formulation for solid and liquid membranes including free-surface contact. *Comput. Methods Appl. Mech. Eng.*, **341**:1–31.
- Scriven, L. E. (1960). Dynamics of a fluid interface – Equations of motion for Newtonian surface fluids. *Chem. Eng. Sci.*, **12**:98–108.
- Sharma, B. L., Perotti, L. E., and Dharmavaram, S. (2023). Computational modeling of coupled interactions of fluid membranes with embedded filaments. *Comput. Methods Appl. Mech. Eng.*, **417**:116441.

- Simo, J. C., Fox, D. D., and Rifai, M. S. (1990). On a stress resultant geometrically exact shell model. Part III: Computational aspects of the nonlinear theory. *Comput. Methods Appl. Mech. Eng.*, **79**:21–70.
- Suchde, P. (2021). A meshfree Lagrangian method for flow on manifolds. *Int. J. Numer. Methods Fluids*, **93**(6):1871–1894.
- Sun, M., Xiao, X., Feng, X., and Wang, K. (2022). Modeling and numerical simulation of surfactant systems with incompressible fluid flows on surfaces. *Comput. Methods Appl. Mech. Eng.*, **390**:114450.
- Torres-Sánchez, A., Millán, D., and Arroyo, M. (2019). Modelling fluid deformable surfaces with an emphasis on biological interfaces. *J. Fluid Mech.*, **872**:218–271.
- Toshniwal, D. and Hughes, T. J. R. (2021). Isogeometric discrete differential forms: Non-uniform degrees, Bézier extraction, polar splines and flows on surfaces. *Comput. Methods Appl. Mech. Eng.*, **376**:113576.
- Valizadeh, N. and Rabczuk, T. (2019). Isogeometric analysis for phase-field models of geometric PDEs and high-order PDEs on stationary and evolving surfaces. *Comput. Methods Appl. Mech. Eng.*, **351**:599–642.
- Zimmermann, C., Toshniwal, D., Landis, C. M., Hughes, T. J. R., Mandadapu, K. K., and Sauer, R. A. (2019). An isogeometric finite element formulation for phase fields on deforming surfaces. *Comput. Methods Appl. Mech. Eng.*, **351**:441–477.